IBM Business Process Manager
Version 7 Release 5

IBM Business Process Manager Standard Installation Guide

IBM

PDF books and the information center

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

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

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

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

Contents

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

The IBM® Business Process Manager installer and configuration programs accommodate a variety of development and production scenarios. This roadmap guides you through the most common and widely used paths for installing and configuring IBM Business Process Manager Standard in a stand-alone or network deployment environment.

The roadmap includes a section for each common installation and configuration scenario. Each section includes a table listing the activities to perform, with links to the instructions and notes about the result of completing each activity.

The sections in this roadmap describe the following scenarios:

- Typical installation.

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

  The Typical installation option is the simplest and quickest method for installing and configuring the software. It is also the ideal method for getting full Business Process Manager features and functionality (for Express, Standard, or Advanced configurations) on a single machine. Typical installation is only available from an electronic installation image (not a DVD).

  See Installing IBM Business Process Manager Standard using the **Typical installation** option.

- Custom installation for a stand-alone environment.

  The **Custom installation** option installs the software on to your system and then runs the Profile Management Tool, giving you more control over how to configure your software. If you are creating a stand-alone Process Center or Process Server environment that requires configuration features that are not available through the Typical installation path, create the stand-alone Process Center or Process Server profiles after the installation of the product binaries by using either the Profile Management Tool or the `manageprofiles` command-line utility.

  See Installing IBM Business Process Manager Standard using the **Custom installation** option and configuring one or more stand-alone profiles.

- Custom installation and configuration for a network deployment environment:

  This installation and configuration scenario provides the capacity, scalability, and robustness that is generally required of a large-scale production environment. A network deployment configuration is made up of a group of servers (known as clusters) that collaborate to provide workload balancing and failover. The servers are managed centrally, using a single administrative console known as the deployment manager.

  After custom installation, you have several configuration options.

  - Use the `configureNode` command-line utility for a simple network deployment configuration:

    The installer installs the software on to your system, and then you use the `configureNode` command to create and configure a deployment manager and one or more custom (managed node) profiles and create a pattern-based network deployment configuration. See Installing IBM Business Process Manager Standard and configuring a network deployment environment using the configureNode command.

  - Use profile configuration and the deployment environment wizard:

    The installer installs the software on to your system, and then you use the profile configuration programs (Profile Management Tool or `manageprofiles` command-line utility) to create and configure one or more deployment manager and custom (managed node) profiles and the deployment environment wizard to create a pattern-based network deployment configuration. See

Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the deployment environment wizard.

– Use profile configuration and the administrative console or wsadmin commands:

The installer installs the software on to your system, and then you use the profile configuration programs (Profile Management Tool or **manageprofiles** command-line utility) to create and configure one or more deployment manager and custom (managed node) profiles and the administrative console or wsadmin commands to create a customized network deployment configuration. See Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the administrative console or wsadmin commands.

For conceptual information about network deployment, see Choosing a stand-alone or network deployment environment and Planning your deployment environment.

For information about topology patterns packaged with the software and supported by the deployment environment wizard, see Overview: Deployment environment topologies and patterns.

Installing IBM Business Process Manager Standard using the Typical installation option

If you are installing from an electronic image, you can use the Typical installation option in the installer to install the software and configure a stand-alone profile automatically. In a Typical installation there is no need to run the Profile Management Tool or run **manageprofiles** after the installation.

The Typical option is the quickest way to install and configure the software. This installation option will default the majority of settings but does allow you to specify the installation path and database type.

For a Typical installation, you install Process Center and Process Server separately. You first run the Typical installation for Process Center to completion and then run the Typical installation for Process Server to completion.

For information about installing IBM Business Process Manager Standard using the **Typical** option, see Typical installation: IBM Business Process Manager Standard.

Installing IBM Business Process Manager Standard using the Custom installation option and configuring one or more stand-alone profiles

You can use the Custom installation option in the installer to install the software and configure one or more stand-alone profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Table 1. Installing IBM Business Process Manager Standard using the Custom installation option and configuring one or more stand-alone profiles

Task	Where to find information	Result after completing the task
Install the software using the Custom installation option.	For information about installing IBM Business Process Manager Standard using the Custom installation option, see Custom installation: IBM Business Process Manager Standard	The custom installation installs the product binaries for Process Center, IBM Process Designer, and Process Server on to your system. The custom installation does not create profiles during the installation.

Table 1. Installing IBM Business Process Manager Standard using the Custom installation option and configuring one or more stand-alone profiles (continued)

Task	Where to find information	Result after completing the task
Launch the Profile Management Tool.	If you did not launch the Profile Management Tool from the installer, you can do so now. See Starting the Profile Management Tool **Restriction:** You cannot use the Profile Management Tool to create profiles on BPM installations on 64-bit architectures except on the Linux on zSeries platform. Instead, proceed to the next step and use the **manageprofiles** command-line utility.	The Profile Management Tool is started and ready to use. Because data from the Process Center profile is used as input to the Process Server profile creation process, you should create your Process Center stand-alone profile first (before creating the Process Server stand-alone profile).
Create one or more Process Center stand-alone profiles.	Depending on how you want to create your Process Center stand-alone profiles, see one of the following topics: • Creating Process Center stand-alone profiles using the Profile Management Tool • Creating Process Center stand-alone profiles using the manageprofiles utility	You have created a Process Center stand-alone profile. This profile defines your stand-alone server environment and it contains command files, configuration files, and log files. The profile creation process configures the database (common and component-specific) and generates the database tables required to support your stand-alone server environment. You can now create your Process Server stand-alone profiles, or you can install Process Designer.
Install Process Designer.	If you want to begin using the business process designing capabilities of the software, download and install Process Designer. See Installing IBM Process Designer	You have installed Process Designer. For information about how to use Process Designer, see the following topics: • Hiring Sample Tutorial for IBM Process Designer • Authoring environments • Process development with the Process Center
Create one or more Process Server stand-alone profiles.	If your objective is to fully configure the software, including the runtime environment, create the Process Server stand-alone profiles. Depending on how you want to create your Process Server stand-alone profiles, see one of the following topics: • Creating Process Server stand-alone profiles using the Profile Management Tool • Creating Process Server stand-alone profiles using the manageprofiles utility	The stand-alone server environment is now configured and ready to use.

Task	Where to find information	Result after completing the task

If you cleared the **Run database scripts to initialize the databases** check box during profile creation, run the database scripts manually and then run the bootstrap utility to load the database with system information.	See Loading the database with system information for information about how to run the bootstrap utility.
	
	`Windows`
	bootstrapProcessServerData.bat
	
	`Linux` `UNIX`
	bootstrapProcessServerData.sh
	
	You must run the bootstrap utility prior to the first startup of a cluster member.
		You have loaded the database with system information prior to successfully starting the Process Server / Process Center.

Installing IBM Business Process Manager Standard and configuring a network deployment environment using the configureNode command

You can use the Typical or Custom installation option in the installer to install the software and then use the **configureNode** command to configure a deployment manager and one or more custom (managed node) profiles and create a pattern-based network deployment environment. You can also use **configureNode** at a later time to add nodes to an existing environment.

The **configureNode** command creates a single cell environment using the Remote Messaging and Remote Support topology pattern, which includes the following three clusters:
- Messaging infrastructure cluster
- Supporting infrastructure cluster
- Application deployment target cluster

Table 2. Installing and configuring IBM Business Process Manager Standard using the configureNode command

Task	Where to find information	Result after completing the task
Install the software	For information about installing IBM Business Process Manager using the Typical installation option, see Typical installation: IBM Business Process Manager Standard For information about installing IBM Business Process Manager using the Custom installation option, see Custom installation: IBM Business Process Manager Standard	The installation for IBM Business Process Manager Standard installs the product binaries for Process Center and Process Designer on to your system. Process Server functionality is part of Process Center, but does not include Advanced Process Server functionality. For example, components for Business Process Choreographer, Business Rules, and Business Space are not part of IBM Business Process Manager Standard.
Use the configureNode command to create the deployment manager, one or more custom nodes, and the deployment environment.	See Creating the deployment environment using the configureNode command	You have created your Process Center or Process Server network deployment environment.

Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the deployment environment wizard

You can use the Custom installation option in the installer to install the software and use the Profile Management Tool or **manageprofiles** command-line utility to configure one or more deployment manager and custom (managed node) profiles. After profile creation, you can use the deployment environment wizard to generate a pattern-based network deployment configuration.

You can choose which pattern to select from the deployment environment wizard based on the planned usage of your environment (proof of concept, testing, or production) and the processing capabilities (such as high-availability and failover processing) required by your enterprise.

This installation and configuration scenario assumes that your business needs and requirements can be met by one of the topology patterns included with the software. If you have requirements that cannot be addressed by the patterns included with the software, you can create a customized network deployment environment using the administrative console. See "Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the administrative console or wsadmin commands" on page 7.

Table 3. Installing and configuring IBM Business Process Manager Standard using the deployment environment wizard

Task	Where to find information	Result after completing the task
Install the software using the Custom installation option.	For information about installing IBM Business Process Manager using the Custom installation option, see Custom installation: IBM Business Process Manager Standard.	The custom installation installs the product binaries for Process Center, Process Designer and WebSphere ESB on to your system. Process Server functionality is part of Process Center, but does not include Advanced Process Server functionality. For example, components for Business Process Choreographer, Business Rules, and Business Space are not part of IBM Business Process Manager Standard. You can now begin the profile creation process. The Profile Management Tool for IBM Business Process Manager Standard does not support importing a database design document.
Launch the Profile Management Tool.	If you did not launch the Profile Management Tool from the installer, you can do so now. See Starting the Profile Management Tool **Restriction:** You cannot use the Profile Management Tool to create profiles on BPM installations on 64-bit architectures except on the Linux on zSeries platform. Instead, proceed to the next step and use the **manageprofiles** command-line utility.	The Profile Management Tool is started and ready to use. Because data from the Process Center profile is used as input to the Process Server profile creation process, you should create your Process Center stand-alone profile first (before creating the Process Server stand-alone profile).

Table 3. Installing and configuring IBM Business Process Manager Standard using the deployment environment wizard (continued)

Task	Where to find information	Result after completing the task
Create a Process Center deployment manager profile.	Depending on how you want to create your profiles, see one of the following topics: • Creating Process Center deployment manager profiles using the Profile Management Tool • Creating Process Center deployment manager and custom profiles using manageprofiles	You have a deployment manager profile. This profile defines your environment and it contains command files, configuration files, and log files. A deployment manager is a server that manages operations for a logical group, or cell, of other servers. The deployment manager is the central location for administering the servers and clusters.
Start the deployment manager and verify that the start operation was successful.	For information about how to start the deployment manager, see Starting deployment managers	The deployment manager server is started.
Create the Process Center custom (managed node) profile or profiles.	This task assumes that you **are not federating nodes** as part of the profile creation process. It assumes that you will set parameters in the Profile Management Tool or **manageprofiles** command for *federating nodes later* (after creating the custom (managed node) profiles.) Depending on how you want to create your profiles, see one of the following topics: • Creating Process Center custom profiles (managed nodes) using the Profile Management Tool • Creating Process Center deployment manager and custom profiles using manageprofiles Repeat this task for each managed node.	You have your custom (managed node) profiles. These nodes are managed by and administered from the deployment manager. You can now federate the node into the Process Center deployment manager cell. The managed node contains a node agent and can contain managed servers. In a managed node, you can configure and run managed servers.
Federate the custom (managed) node to the deployment manager.	See Federating custom nodes to a deployment manager for instructions about how to use the **addNode** command to federate the Process Center custom node or nodes into the Process Center deployment manager cell.	The custom profile is federated into the deployment manager.
Using the deployment environment wizard, create the network deployment configuration.	For information about how to use the deployment environment wizard, see Creating the Process Center deployment environment using a pattern	As part of this task you need to address any deferred configuration items and then generate the Process Center deployment environment. You have created your Process Center network deployment environment.

Task	Where to find information	Result after completing the task

Create a cell for Process Server.	To run Process Server, you must create a separate cell.
	
	This involves the same sequence of steps, except that you create a Process Server deployment manager and Process Server custom (managed nodes) and then use the deployment environment wizard to create the Process Server deployment environment, as described in the following topics:
	
	• Creating Process Server deployment manager profiles using the Profile Management Tool
	• Creating Process Server custom profiles (managed nodes) using the Profile Management Tool
	• Creating Process Server deployment manager and custom profiles using manageprofiles
	  – Federating custom nodes to a deployment manager
	• Creating the Process Server deployment environment using a pattern
	  As part of this task, you must address any deferred configuration items and then generate the Process Server deployment environment.
		You have created your Process Server network deployment environment.

Perform manual database configuration steps.	If you have deferred configuration steps, you must complete additional database configuration before starting the clusters in your deployment environment. See Completing the database configuration for your network deployment environment	You have completed the required manual database configuration activities for the network deployment environment.
Load the database with system information.	See Loading the database with system information for information about how to run the bootstrap utility. **Windows** bootstrapProcessServerData.bat **Linux** **UNIX** bootstrapProcessServerData.sh You must run the bootstrap utility prior to the first startup of a cluster member.	You have loaded the database with system information prior to successfully starting the Process Server / Process Center.

Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the administrative console or wsadmin commands

The topology patterns packaged with the software and implemented using the deployment environment wizard are intended to address a broad spectrum of business processing requirements. However, if you have a scenario that the topology patterns do not address sufficiently, you can use the administrative console to create a customized network deployment environment.

Note: This scenario is intended for users who have an advanced understanding of how to configure product components and functionality using the administrative console. Before you embark on the installation and configuration scenario described in this section, consider using the deployment environment wizard to create your network deployment environment.

Command assistance is available for a subset of administrative console actions. When available, command assistance displays the wsadmin scripting command for the last console action you performed. You can then use this data to create wsadmin scripts that automate certain administrative tasks. For more information on command assistance, see Administrative console actions with command assistance.

Table 4. Installing and configuring IBM Business Process Manager Standard using the administrative console

Task	Where to find information	Result after completing the task
Install the software and configure the required profiles.	Complete the tasks listed in "Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the deployment environment wizard" on page 5 up to the task *Using the deployment environment wizard, create the network deployment configuration*. Instead of using the deployment environment wizard, you will create a customized deployment environment manually using the administrative console.	The custom installation for IBM Business Process Manager Standard installs the product binaries for Process Center and Process Designer on to your system. Process Server functionality is part of Process Center, but does not include Advanced Process Server functionality. For example, components for Business Process Choreographer, Business Rules, and Business Space are not part of IBM Business Process Manager Standard. You have created the deployment manager and custom managed nodes and have federated the nodes into the deployment manager cell. You can now use the administrative console to create servers, server clusters, and the components that will comprise your network deployment environment.
Create and configure servers and clusters using the administrative console.	See Creating and configuring servers and clusters using the administrative console	You have created the servers and server clusters for your environment.
Configure a server or cluster as a Process Server.	For information about using the administrative console to configure your server or server cluster as a Process Server, see Configuring a server or cluster as a Process Server You can also configure a server or cluster as a Process Server using wsadmin. See the configureProcessServer command	You have configured the servers or server clusters as Process Servers.
Configure the Performance Data Warehouse component on a server or cluster.	For information about using the administrative console to configure Performance Data Warehouse on a server or cluster, see Configuring the Performance Data Warehouse component on a server or cluster You can also configure Performance Data Warehouse on a server or cluster using wsadmin. See the configurePerfDW command	You have configured Performance Data Warehouse on the Process Server.

Table 4. Installing and configuring IBM Business Process Manager Standard using the administrative console (continued)

Task	Where to find information	Result after completing the task
Set up the Messaging Service.	For information about using the administrative console to configure the Messaging Service, see Setting up the messaging server environment	You have set up the messaging server environment.
Configure the JNDILookup web service.	For information about using the administrative console to configure the JNDILookup web service, see Configuring the JNDILookup Web Service	You have configured the JNDILookup Web Service.
Load the database with system information.	See Loading the database with system information for information about how to run the bootstrap utility. **Windows** bootstrapProcessServerData.bat **Linux** **UNIX** bootstrapProcessServerData.sh You must run the bootstrap utility prior to the first startup of a cluster member.	You have loaded the database with system information prior to successfully starting the Process Server / Process Center.

Chapter 2. Installation types and profiles

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

The following tables describe the relationships.

Table 5. Profile configuration as part of installation: IBM Business Process Manager Advanced

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

Table 6. Profile configuration as part of installation: IBM Business Process Manager Advanced: Process Server

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

Table 7. Profile configuration as part of installation: IBM Business Process Manager Standard

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

Table 8. Profile configuration as part of installation: IBM Business Process Manager Express

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

Chapter 3. Databases and installation and configuration scenarios

How databases are configured to work with the software is related to the software you are installing and the scenario you are trying to achieve.

In installation scenarios where you will be using the **Typical** installation option, you must pre-create your database, whether that database resides locally (on the same system on which you are installing IBM Business Process Manager or remotely (on a separate system). The only scenario where you are not required to pre-create your database is when you are installing the DB2 Express database that is embedded with IBM Business Process Manager. DB2 Express is embedded with IBM Business Process Manager on Windows and on Linux for Intel. If you do not already have DB2 Express installed, it is optionally installed when you run the IBM Business Process Manager installer, and the installer also creates the common and component-specific databases on DB2 Express. To install DB2 Express, you must have administrative privileges (Administrator or root user) and must not have an existing DB2 database server on the system.

Any scenario where you are not using DB2 Express for your **Typical** install is considered to be a **Typical** installation with an *existing database*, meaning you already have a database installed and you will use the *Existing database panel* of the installer to specify its configuration properties. See Configuring an existing database during a typical installation for details. To use this option, you must ensure that your database has already been created.

You and your database administrator should review the information on configuring databases in the information center prior to installing and configuring the software.

Chapter 4. Preparing to install and configure the software

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

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

Table 9. Preparing for installation and configuration

Task	Where to find information	Result after completing the task
Review hardware and software requirements	Depending on your IBM BPM configuration, visit: • IBM Business Process Manager Advanced system requirements • IBM Business Process Manager Standard system requirements • IBM Business Process Manager Express system requirements	You understand the system requirements necessary to support your IBM Business Process Manager installation.
Prepare your operating system	AIX Preparing AIX systems for installation Linux Preparing Linux systems for installation Solaris Preparing Solaris systems for installation Windows Preparing Windows systems for installation	You have prepared the operating system of each workstation to be used.

Table 9. Preparing for installation and configuration (continued)

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

Chapter 5. Planning for IBM Business Process Manager

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

Assessing your requirements

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

Several factors determine your software needs. These factors can be organized into the following categories.
- Product hardware and software requirements, your own system resource constraints, and the availability of resources to administer and maintain your system
- Applications to be deployed to the runtime environment, and the intended use of the configured environment
- Products, and the versions of these products, to install to meet your requirements

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

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

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

Process and process application considerations

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

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

Consider the following:
- Consider how process applications interact with existing services and back-end systems.

- Consider how process applications handle data and how data flows through your system to address a specific business need.

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

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

  - Process application invocation patterns

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

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

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

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

    Interruptible business processes, or macroflows, are long-running business processes that contain a set of activities, each of which is performed in its own transaction. Interruptible business processes can include activities that require human intervention or calls to remote systems or both. Asynchronous activities cause a business process to be interruptible because these activities might take minutes, hours, or even days to complete.

Resource considerations

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

Consider the following factors:

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

Development and deployment version levels

When you try to determine the version levels of IBM Business Process Manager that you need in your environment, your decision depends on the version levels that were used when your applications were

developed. Generally, applications deployed in a previous version of IBM Business Process Manager can run on the next available version of IBM Business Process Manager.

The following table describes compatibility between IBM Business Process Manager V7.5.1, including IBM Integration Designer V7.5.1 (previously WebSphere® Integration Developer) and IBM Process Designer V7.5.1, and prior releases.

Task	Supported?
Deployment from WebSphere Integration Developer version 6.1.0, 6.1.2, or 6.2.0 or 7.0.0 or IBM Integration Designer 7.5 to IBM Business Process Manager V7.5.1.	Yes. **Important:** For WebSphere Adapters V6.1.0, V6.1.2 and V6.2.0, you must install the interim fix titled *Mandatory adapter fix for running 6.1 and 6.2 Adapters on WPS v7.0*. If you do not plan to update the WebSphere Adapter to a V7.0 level, and you plan to continue to use the application with WebSphere Adapter V6.1.0, V6.1.2 or V6.2.0, you must apply this interim fix on the source environment. **Important:** Websphere Adapter for SAP V6.0.2, V6.1.0, V6.1.2 and V6.2.0 are not supported on IBM Business Process Manager V7.5.1. You must update Websphere Adapter for SAP to V7.0 before you can deploy any applications that use Websphere Adapter for SAP on IBM Business Process Manager V7.5.1.
Running IBM Business Process Manager V7.5.1 artifacts on WebSphere Process Server 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Business Process Manager 7.5.	No. Applications authored with IBM Integration Designer V7.5.1 cannot be published to or installed on WebSphere Process Server 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Business Process Manager 7.5 (any prior release) servers. Applications authored with WebSphere Integration Developer 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Integration Designer 7.5 and then generated in IBM Integration Designer V7.5.1 cannot be published to or installed on WebSphere Process Server 6.1.0, 6.1.2, 6.2.0, or 7.0 orIBM Business Process Manager 7.5 servers. Applications generated using `serviceDeploy` from IBM Business Process Manager V7.5.1 servers cannot be installed on WebSphere Process Server 6.1.0, 6.1.2, 6.2.0, or 7.0 or IBM Business Process Manager 7.5 servers.

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

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

Profile naming considerations

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

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

Double-byte characters are allowed.

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

Node, server, host, and cell naming considerations

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

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

Descriptions of fields on the Node and Hosts Names and Node, Host, and Cell Names pages: Table 10 describes the fields found on the Node and Host Names and Node, Host, and Cell Names pages of the Profile Management Tool, including the field names, default values, and constraints. Use this information as a guide when you are creating profiles.

Table 10. Naming guidelines for nodes, servers, hosts, and cells

Field name	Default value	Constraints	Description
Stand-alone server profiles			
Node name	**Linux** **UNIX** **Windows** *shortHostName* Node *NodeNumber* where: • *shortHost Name* is the short host name. • *NodeNumber* is a sequential number starting at 01.	Avoid using the reserved names.	Select any name you want. To help organize your installation, use a unique name if you plan to create more than one server on the system.
Server name	**Linux** **UNIX** **Windows** server1	Use a unique name for the server.	The logical name for the server.
Host name	**Linux** **UNIX** **Windows** The long form of the domain name server (DNS) name.	The host name must be addressable through your network. If you are planning to use Business Space, use a fully qualified host name.	Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table.

Table 10. Naming guidelines for nodes, servers, hosts, and cells (continued)

Field name	Default value	Constraints	Description
Cell name	Linux UNIX Windows *shortHostName* Node *NodeNumber* Cell where: • *shortHost Name* is the short host name. • *NodeNumber* is a sequential number starting at 01.	Use a unique name for the cell. A cell name must be unique in any circumstance in which the product is running on the same physical workstation or cluster of workstations, such as a Sysplex. Additionally, a cell name must be unique in any circumstance in which network connectivity between entities is required either between the cells or from a client that must communicate with each of the cells. Cell names also must be unique if their name spaces are going to be federated. Otherwise, you might encounter symptoms such as a javax.naming.Name NotFoundException exception, in which case, you need to create uniquely named cells.	All federated nodes become members of a deployment manager cell.
Deployment manager profiles			
Node name	Linux UNIX Windows *shortHostName* Cell Manager*Node Number* where: • *shortHost Name* is the short host name. • *NodeNumber* is a sequential number starting at 01.	Use a unique name for the deployment manager. Avoid using the reserved names.	The name is used for administration within the deployment manager cell.
Host name	Linux UNIX Windows The long form of the domain name server (DNS) name.	The host name must be addressable through your network. Avoid using the reserved names. If you are planning to use Business Space, use a fully qualified host name.	Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table.

Table 10. Naming guidelines for nodes, servers, hosts, and cells (continued)

Field name	Default value	Constraints	Description
Cell name	Linux UNIX Windows *shortHostName* Cell *CellNumber* where: • *shortHost Name* is the short host name. • *CellNumber* is a sequential number starting at 01.	Use a unique name for the deployment manager cell. A cell name must be unique in any circumstance in which the product is running on the same physical workstation or cluster of workstations, such as a Sysplex. Additionally, a cell name must be unique in any circumstance in which network connectivity between entities is required either between the cells or from a client that must communicate with each of the cells. Cell names also must be unique if their name spaces are going to be federated. Otherwise, you might encounter symptoms such as a javax.naming.Name NotFoundException exception, in which case, you need to create uniquely named cells.	All federated nodes become members of the deployment manager cell , which you name in the Node, Host, and Cell Names page of the Profile Management Tool.
Custom profiles			
Node name	Linux UNIX Windows *shortHostName* Node *NodeNumber* where: • *shortHost Name* is the short host name. • *NodeNumber* is a sequential number starting at 01.	Avoid using the reserved names. Use a unique name within the deployment manager cell.	The name is used for administration within the deployment manager cell to which the custom profile is added. Use a unique name within the deployment manager cell.
Host name	Linux UNIX Windows The long form of the domain name server (DNS) name.	The host name must be addressable through your network. If you are planning to use Business Space, use a fully qualified host name.	Use the actual DNS name or IP address of your workstation to enable communication with it. See additional information about the host name following this table.

Host name considerations:

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

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

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

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

The value that you specify for the host name is used as the value of the hostName property in configuration documents. Specify the host name value in one of the following formats:
* Fully qualified domain name servers (DNS) host name string, such as xmachine.manhattan.ibm.com
* The default short DNS host name string, such as xmachine
* Numeric IP address, such as 127.1.255.3

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

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

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

Another disadvantage of this format is that you cannot use the node if the host is disconnected from the network.

Preparing necessary security authorizations

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

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

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

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

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

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

Installation directories for the product and profiles

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

Variables used in the documentation

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

Here are the main variables used in the documentation:

Linux   UNIX   Windows *install_root*

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

profile_root
> Location of a IBM Business Process Manager profile.

How variable meanings can differ

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

 Linux   UNIX   Windows

Limitations of nonroot installers

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

Default directories for Typical Installation

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

Table 11 shows the default installation root directory into which the installation program installs both IBM Business Process Manager and WebSphere Application Server Network Deployment for both root (Administrator) and nonroot users.

Table 11. install_root default directory

Default *install_root* for root or Administrator users	Default *install_root* for nonroot users
AIX `/opt/IBM/BPM/v7.5`	**AIX** *user_home*`/IBM/BPM/v7.5`
Linux `/opt/ibm/BPM/v7.5`	**Linux** *user_home*`/ibm/BPM/v7.5C:\IBM\BPM\v7.5`
Windows `C:\IBM\BPM\v7.5`	**Windows** `C:\IBM\BPM\v7.5`

Table 12 shows the default installation directory for a profile named *profile_name* for both root (Administrator) and nonroot users.

Table 12. profile_root default directory

Default *profile_root* for root or Administrator users	Default *profile_root* for nonroot users
AIX `/opt/IBM/BPM/v7.5/profiles/`*profile_name*	**AIX** *user_home*`IBM/BPM/v7.5/profiles/`*profile_name*
Linux `/opt/ibm/BPM/v7.5/profiles/`*profile_name*	**Linux** *user_home*`/ibm/BPM/v7.5/profiles/`*profile_name*
Windows `C:\IBM\BPM\v7.5\profiles\`*profile_name*	**Windows** `C:\IBM\BPM\v7.5\profiles\`*profile_name*

Table 13 on page 26 shows the DB2 Express install location. DB2 Express is installed under the same directory as IBM Business Process Manager.

Table 13. DB2 Express default directory

Default DB2 Express binary location	Database Instance location
`Linux` /opt/ibm/BPM/v7.5/DB2	`Linux` Database instance is created under the `bpminst` user. For example: *user_home*/bpminst
`Windows` C:\IBM\BPM\v7.5\DB2	`Windows` The BPMINST database instance is created under the root (\) of the drive where IBM Business Process Manager is installed. For example, if IBM Business Process Manager is installed under C:\IBM\BPM\v7.5\ then you will see C:\BPMINST.

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

The following tables show the default installation locations of the product and its profiles. If you choose to install IBM Business Process Manager on top of an existing supported version of WebSphere Application Server or WebSphere Application Server Network Deployment, IBM Business Process Manager is installed into the same location. Table 14 shows the default installation root directory in such a case for both root (Administrator) and nonroot users.

Table 14. shows the default installation root directory into which the installation program installs both IBM Business Process Manager and WebSphere Application Server for both root (Administrator) and nonroot users.

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

Table 15. shows the default installation directory for a profile named profile_name for both root (Administrator) and nonroot users.

Default *profile_root* for root or Administrator users	Default *profile_root* for nonroot users
`AIX` /usr/IBM/WebSphere/AppServer/profiles/*profile_name*	`AIX` *user_home*/IBM/WebSphere/AppServer/profiles/*profile_name*
`Linux` `Solaris` /opt/IBM/WebSphere/AppServer/profiles/*profile_name*	`Linux` `Solaris` *user_home*/IBM/WebSphere/AppServer/profiles/*profile_name*
`Windows` C:\Program Files\IBM\WebSphere\AppServer\profiles*profile_name*	`Windows` *user_home*\IBM\WebSphere\AppServer\profiles*profile_name*

Default installation directories for Installation Manager

Table 16 on page 27 shows two default directories related to the Installation Manager tool.

The directories under **Installation directory** are the defaults (per platform) into which the launchpad application installs Installation Manager.

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

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

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

Table 16. Installation Manager default installation directories

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

Choosing a stand-alone or network deployment environment

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

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

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

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

Stand-alone environment
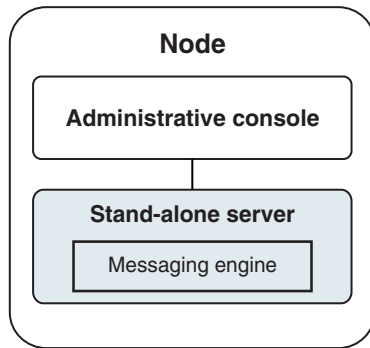
Figure 1. A stand-alone environment

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

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

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

Network deployment environment

A network deployment environment contains a collection of interconnected servers and clusters to run your service requester and provider enterprise applications and their mediation modules. The environment can also include application servers on WebSphere Application Server.
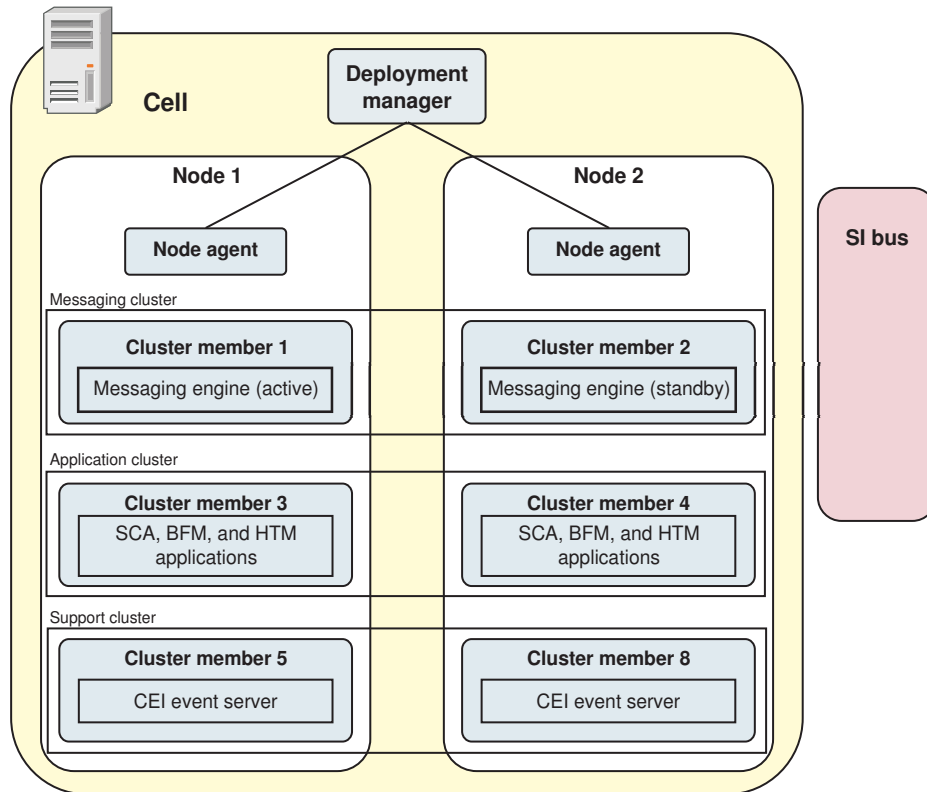
Figure 2. A network deployment environment

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

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

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

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

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

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

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

Table 17. Choice of stand-alone or network deployment cluster topology pattern for intended use of IBM Business Process Manager

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

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

Related concepts:

"Topologies of a network deployment environment" on page 37
A topology is the physical layout of the deployment environment. You can create the topology that best addresses your business needs by choosing one of the patterns provided by IBM or by creating your own customized pattern.

"Considerations for selecting a topology" on page 46
Selecting an appropriate topology for your deployment environment depends upon several factors.

Installation types and profiles

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

The following tables describe the relationships.

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

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

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

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

Table 19. Profile configuration as part of installation: IBM Business Process Manager Advanced: Process Server (continued)

Type of installation	Media	Creates stand-alone Profile?	Profile Management Tool optionally started?	First Steps Console optionally started?	Features available for selection from Installation Manager?
Custom	DVD or electronic image	Optionally A feature allows you to create Process Server and or WebSphere ESB profiles for Unit Test Environment (UTE) only	Yes	No	Yes

Table 20. Profile configuration as part of installation: IBM Business Process Manager Standard

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

Table 21. Profile configuration as part of installation: IBM Business Process Manager Express

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

Planning your network deployment environment

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

Before you plan your deployment environment complete the following tasks:
- Choose a database type
- Identify available resources
- Identify necessary security authorizations

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

1. Identify the functional requirements of the deployment environment.
   a. Identify the features or runtime capabilities of your deployment environment.

      Consider whether the deployment environment will support other IBM Business Process Manager products and non-BPM products in addition to IBM Business Process Manager.
   b. Identify the component types that you will deploy.

      Consider the component types and the interactions between components as part of the requirements.
   c. Identify the import and export implementation types and transports.

      Consider the resources needed for the databases or Java™ Message Service (JMS) resources and the need for business events and their transmission mechanism.
   d. Identify any functional requirements that are not related to applications.

      Consider security servers, routers, and any other hardware or software requirements to handle business events.
2. Identify the capacity and performance requirements for your environment.
3. Decide on the number of physical servers that you need for each function.
4. Design your deployment environment.

   Decide on the pattern. For IBM Business Process Manager, you can select one of four established topology patterns:
   • Single Cluster
   • Remote Messaging
   • Remote Messaging and Remote Support
   • Remote Messaging, Remote Support, and Web

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

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

   For more information about the patterns and the differences between them, see "Topologies of a network deployment environment" on page 37.
5. Understand the methods available to you for configuring your deployment environment.

   You can configure the following types of deployment environments for IBM Business Process Manager:
   • A standardized network deployment environment

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

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

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

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

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

Overview: Deployment environment topologies and patterns

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

What is a topology?

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

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

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

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

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

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

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

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

The purpose of deployment environment patterns

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

Because the deployment environment topology patterns represent recommended topologies with component configurations that work together, you can be sure that you are building a fully functional deployment environment. You can use the configuration rules of a deployment environment topology

pattern to generate a fast path configuration. This action is possible because many design decisions are implemented in the topology pattern; for example, which components to configure, and which default parameters and resources are needed.

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

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

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

Databases and deployment environments

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

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

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

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

For more information, see Planning your database configuration.

Functions of IBM-supplied deployment environment topology patterns

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

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

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

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

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

Application deployment target

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

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

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

Supporting infrastructure

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

- Business rules
- Selectors
- Human tasks
- Business processes

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

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

Messaging engine infrastructure

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

Web application infrastructure

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

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

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

```
┌─────────────────┐              ┌──────────────────────────────────────────────┐
│      DMGR        │              │ Databases                                    │
│  Administrative  │              │  ┌────────┐ ┌────────┐ ┌────────┐ ┌────────┐ │
│     Console      │              │  │ CMNDB  │ │  MEDB  │ │ PDWDB  │ │ BPMDB  │ │
└─────────────────┘              │  └────────┘ └────────┘ └────────┘ └────────┘ │
┌─────────────────────────────┐  │              ┌────────┐                      │
│ Application Target Cluster   │  │              │ BPEDB  │                      │
│    (AppTargetCluster)        │  │              └────────┘                      │
│                              │  └──────────────────────────────────────────────┘
│ ┌───────────┐ ┌───────────┐  │
│ │ATC_member1│ │ATC_member1│  │
│ └───────────┘ └───────────┘  │   Service Integration Buses
└─────────────────────────────┘
┌─────────────────────────────────────────────────────────┐
│ Messaging Engine Cluster                                 │
│    (MECluster)                                           │
│ ┌───────────────┐ ┌───────────────┐                      │
│ │ME_member1     │ │ME_member2     │                      │
│ │ (ME) (ME)     │ │               │                      │
│ │ (ME) (ME)     │ │               │                      │
│ └───────────────┘ └───────────────┘                      │
└─────────────────────────────────────────────────────────┘
                            CEI   BPC  SCA.SYSTEM  SCA.APP
┌─────────────────────────────┐
│ Support Cluster              │
│    (SupportCluster)          │
│ ┌───────────┐ ┌───────────┐  │
│ │Sup_member1│ │Sup_member2│  │
│ └───────────┘ └───────────┘  │
└─────────────────────────────┘
```
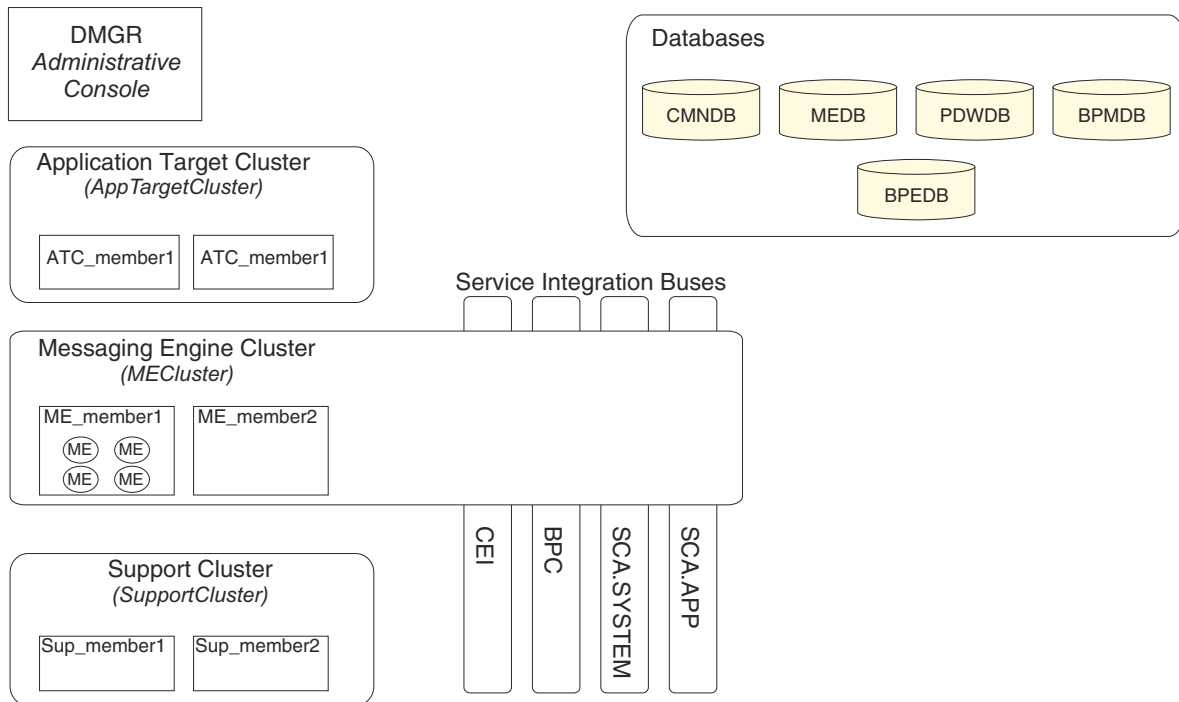
This diagram includes:

- **Deployment Manager** (DMGR): A server that manages operations for a logical group or cell of other servers.
- **Application Target Cluster**: The deployment target for customer applications. This cluster is configured to provide functionality for BPEL processes, human tasks, process applications, and SCA. In a Process Center deployment, this cluster also includes the Process Center console.
- **Databases**:

– Common database (CMNDB) - Required. Contains tables shared on a cell-wide basis for multiple IBM BPM Advanced Process Server and WebSphere ESB capabilities.

– Process Server database (BPMDB) - Required. Contains tables for IBM BPM Advanced Process Server capabilities.

– Performance Data Warehouse database (PDWDB) - Required. Contains tables for Performance Data Warehouse capabilities.

– Messaging Engine database (MEDB) - Each messaging engine requires a unique set of database tables.

– Business Process database (BPEDB) - Each deployment target that is configured for Business Process Choreographer requires a set of tables for Business Process Choreographer functions.

- **Service Integration Buses (SIBuses)**: At least one server or cluster is a member of each of the SIBuses. On multiplatforms, the messaging engine cluster is a member of each of the SIBuses. The bus member hosts a single messaging engine (ME) for each SIBus that it is associated with.

- **Messaging Engine (ME) Cluster**: Hosts the messaging engines. The ME Cluster is a member of each of the four service integration buses.

- **Support Cluster**: The Support Cluster hosts the applications that provide some utility for, but do not contribute to the workload of, the Application Target cluster. For example, the Support Cluster might host the Business Rules Manager.

Topologies of a network deployment environment

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

Related concepts:

"Considerations for selecting a topology" on page 46
Selecting an appropriate topology for your deployment environment depends upon several factors.

"Topology patterns and supported product features" on page 48
A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

"Choosing a stand-alone or network deployment environment" on page 27
Choose a stand-alone environment to evaluate the product or to support development of applications and services. Choose a network deployment environment when your production environment needs additional features such as capacity, availability, scalability, and failover support.

Single Cluster topology pattern

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

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

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

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

- Because asynchronous interactions (involving JMS and MQ/JMS bindings), human tasks, state machines, and long-running business processes can make extensive use of the messaging infrastructure, a single cluster environment is not ideal for applications with these components.

- Any messaging requirements must be kept to a minimum with this topology pattern (except for z/OS).
- Service Component Architecture (SCA) internal asynchronous invocations, the Java Message Service (JMS), and MQ messaging bindings do not support multiple messaging engines in the same cluster.

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

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

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

In a Single Cluster topology pattern, all deployment environment functions and components run on a single cluster:
- The applications:
  - Process Server
  - Process Portal
- The messaging bus members:
  - Common Event Interface (CEI) bus members
  - Performance Data Warehouse (PDW) bus members
  - Process Server bus members
  - Service Component Architecture (SCA) application bus members
  - SCA system bus members
- The support infrastructure applications:
  - CEI server application
  - Performance Data Warehouse
- The web applications:
  - Business Rules manager
  - REST API Services

See the following graphical representation of the Single Cluster topology pattern.
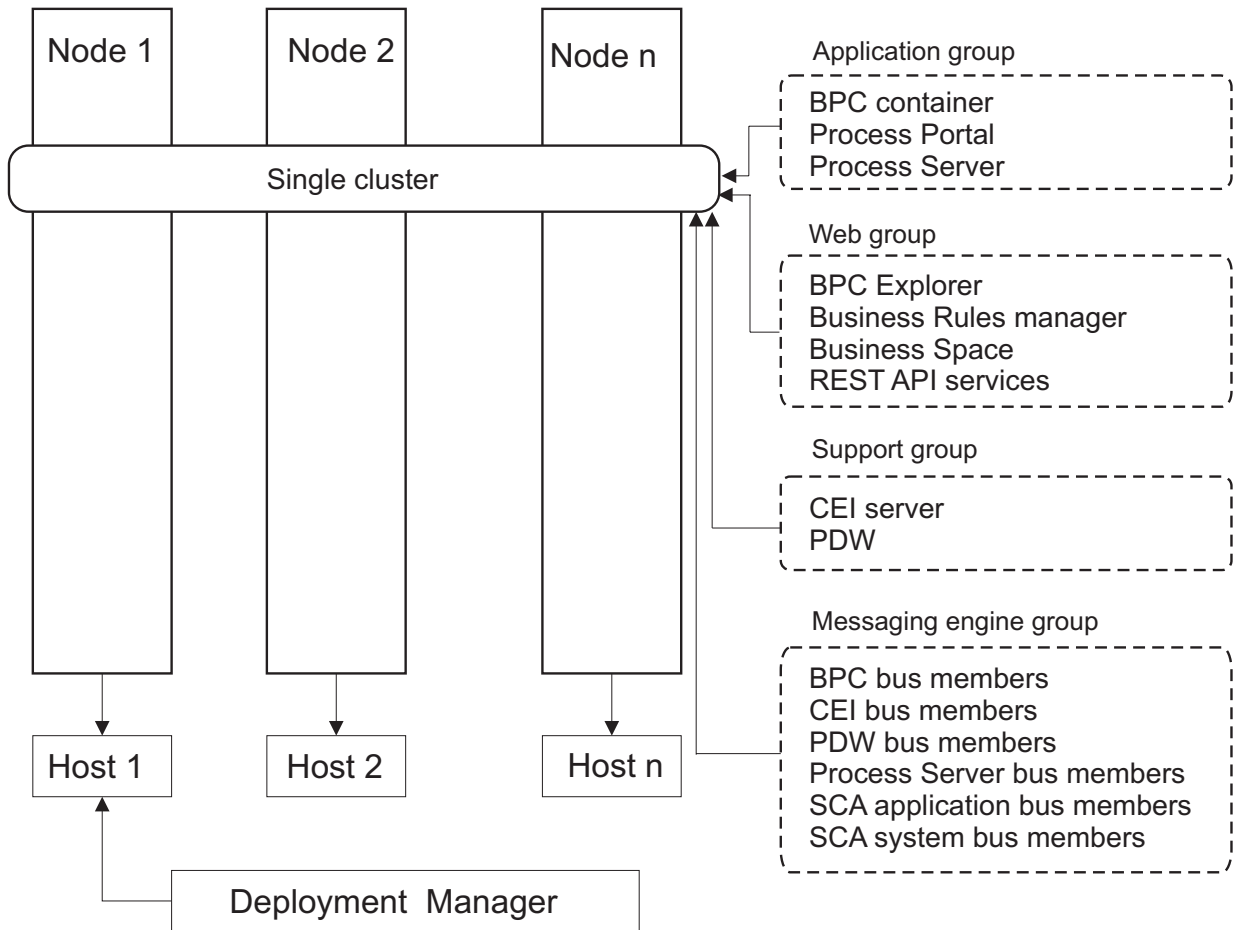
Figure 3. Single cluster topology pattern

Remote Messaging topology pattern

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

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

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

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

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

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

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

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

The application deployment target cluster hosts the following:
- The applications:
  - Process Server
  - Process Portal
- The support infrastructure applications:
  - CEI server application
  - Performance Data Warehouse
- The web applications:
  - Business Rules manager
  - REST API Services

The messaging infrastructure cluster hosts the following:
- Common Event Interface (CEI) bus members
- Performance Data Warehouse (PDW) bus members
- Process Server bus members
- Service Component Architecture (SCA) application bus members
- SCA system bus members
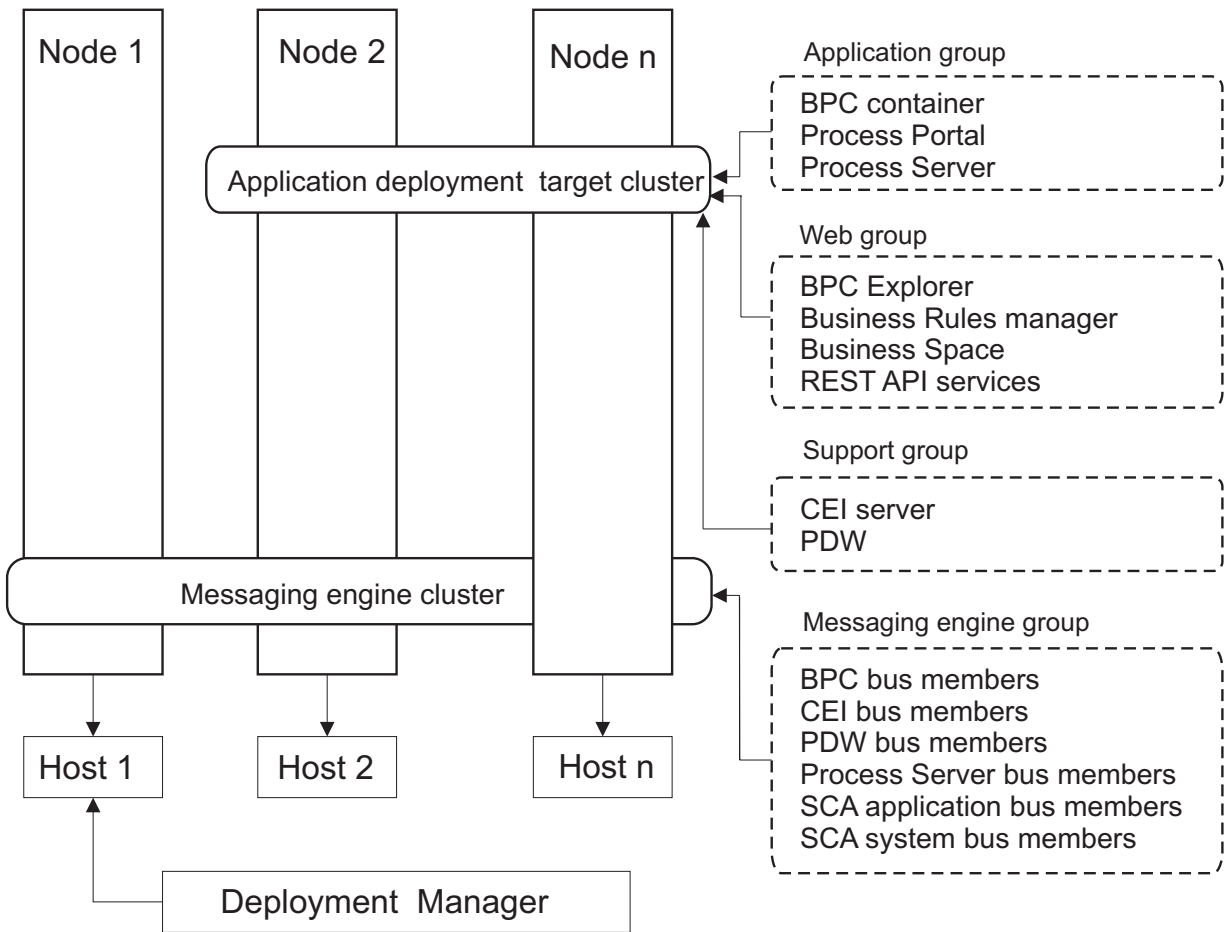
Figure 4. Remote Messaging topology pattern

Remote Messaging and Remote Support topology pattern

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

With this three-cluster topology pattern, resources are allocated to the cluster that handles the highest loads. This topology pattern is the most flexible and versatile, and is preferred by most users (except for z/OS). The components are divided among the three clusters.

For many customers with large computing infrastructures, the Remote Messaging and Remote Support topology pattern is the preferred environment. The hardware requirements for distributed platforms are more intensive. However, you have greater flexibility in adjusting and tuning memory usage for the Java virtual machines (JVMs) when you have three or more clusters with multiple members performing specific functions.

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

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

As with the Remote Messaging topology pattern, the Remote Messaging and Remote Support topology pattern provides an ideal environment for long-running business processes, state machines, human tasks, and asynchronous interactions (including JMS and MQ/JMS bindings).

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

The application deployment target cluster hosts the following:
- Process Server
- Process Portal

The messaging infrastructure cluster hosts the following:
- Common Event Interface (CEI) bus members
- Performance Data Warehouse (PDW) bus members
- Process Server bus members
- Service Component Architecture (SCA) application bus members
- SCA system bus members

The support infrastructure cluster hosts the following:
- The support applications:
  - CEI server application
  - Performance Data Warehouse
- The web applications:
  - Business Rules manager
  - REST API Services
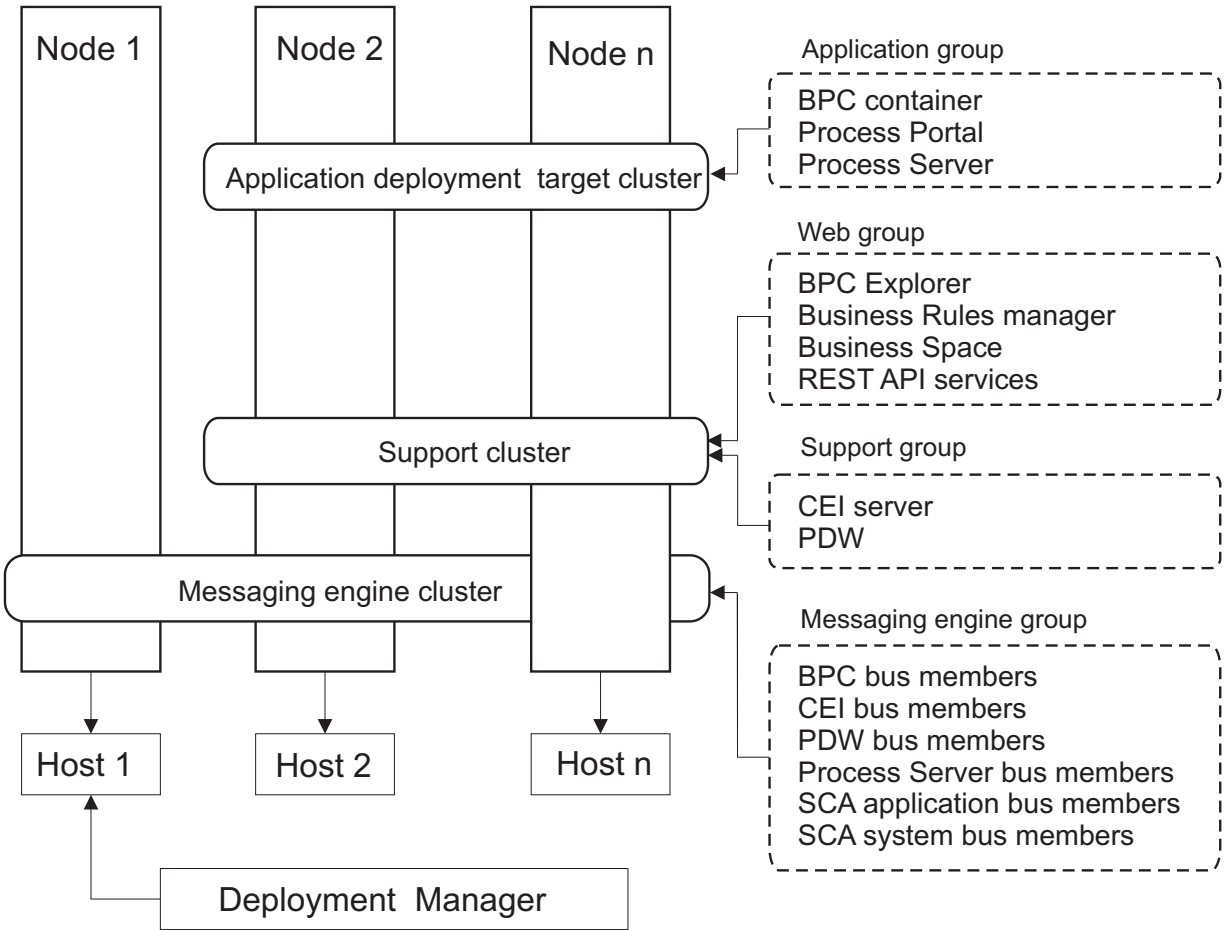
Figure 5. Remote Messaging and Remote Support topology pattern

Resource allocation example

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

Important: Load balancing is not available for the default configuration Remote Messaging and Remote Support topology pattern. That configuration uses a single messaging engine bus, while the load balancing feature requires at least two messaging engine buses.
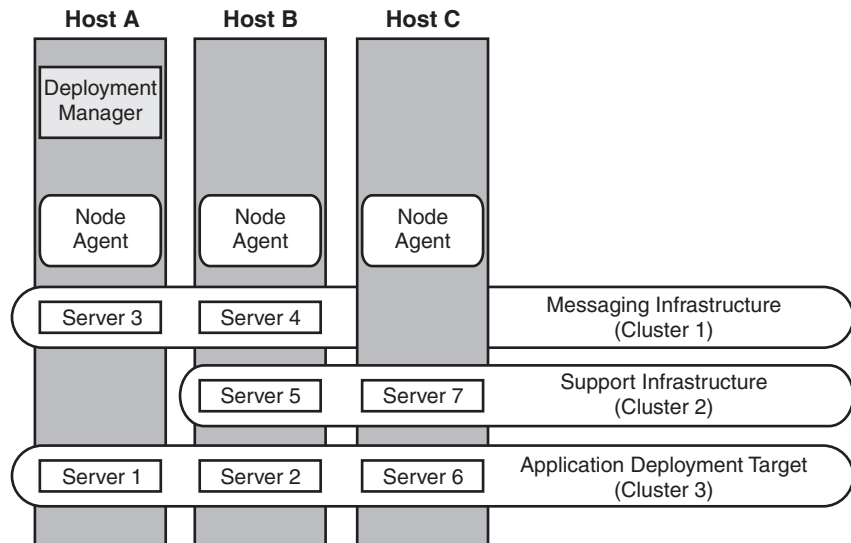
Figure 6. Resource allocation example

Remote Messaging, Remote Support, and Web topology pattern

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

Remote Messaging, Support and Web topology pattern is the recommended starting topology for IBM Business Process Manager Advanced.

Note: Do not use this pattern for a Standard configuration unless it includes IBM Business Monitor.

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

The application deployment target cluster hosts the following:
- Process Server
- Process Portal

The messaging infrastructure cluster hosts the following:
- Common Event Interface (CEI) bus members
- Performance Data Warehouse (PDW) bus members
- Process Server bus members
- Service Component Architecture (SCA) application bus members
- SCA system bus members

The support infrastructure cluster hosts the following:
- CEI server application
- Performance Data Warehouse

The web application cluster hosts the following:
- Business Rules manager

- REST API Services

In a Remote Messaging, Support and Web topology pattern, the deployment environment functions are divided among four separate clusters. One cluster is used for messaging functionality, one cluster for support functionality, one cluster for applications, and one cluster for web-based functions.

In addition to the ability to precisely control the individual components in your environment, the advantages of this topology pattern are similar to the advantages of the Remote Messaging and Remote Support topology pattern.

See the following graphical representation of a Remote Messaging, Support and Web topology.
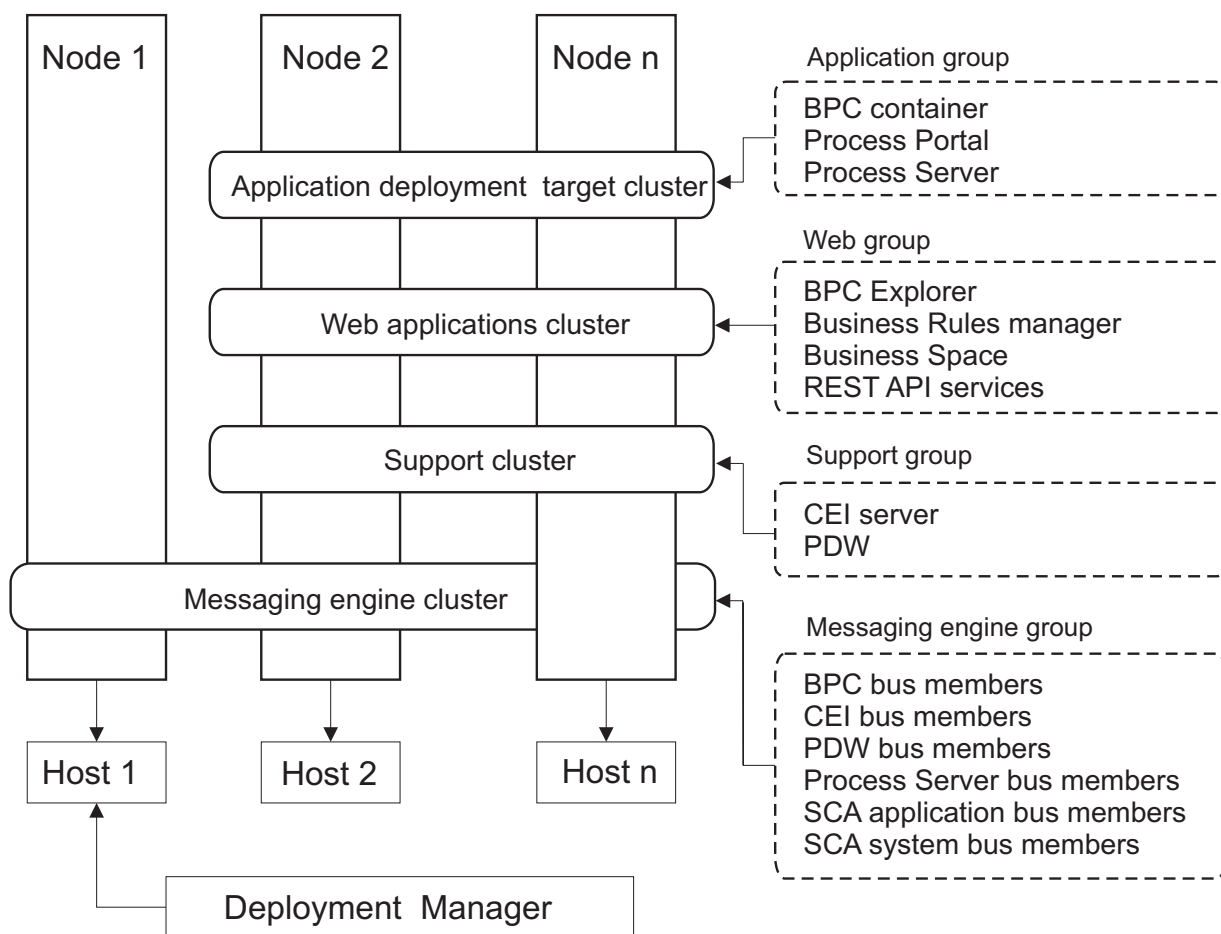
Figure 7. Remote Messaging, Support and Web topology pattern

Customized topology

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

If you need to define your own deployment environment topology, a customized topology is by far the most flexible. The IBM-supplied topologies (Single Cluster, Remote Messaging, Remote Messaging and Remote Support, and Remote Messaging, Remote Support, and Web), deploy all IBM Business Process Manager components to their default locations. You might or might not need the additional overhead associated with these components. For example, if your organization does not need Common Event

Interface (CEI), you could create a custom topology that removes CEI support from your environment. Similarly, if your organization has governance rules that prevent you from taking advantage of the Business Rules Manager, you could remove it from your deployment environment.

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

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

Considerations for selecting a topology

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

When you select a topology pattern, consider the following factors:
- Available hardware resources
- Application invocation patterns
- Types of business processes that you plan to implement (interruptible versus non-interruptible)
- How heavily you intend to use the Common Event Infrastructure (CEI)
- Individual scalability requirements
- Administrative effort involved

The IBM-supplied topologies can be applied to both Process Server and Process Center topologies. Therefore, your Process Center and Process Server network deployment environments can be organized in a similar way.

The procedures for creating environments for Process Server and Process Center based on IBM-supplied topologies are also similar. The only difference related to IBM-supplied patterns is the recommended patterns for a production environment, and the components configured on the clusters for those patterns.

For information on the components, features and functionality available in each of the IBM Business Process Manager configurations, see IBM Business Process Manager V7.5.1 configuration capabilities

Follow these tips regarding IBM-supplied topologies.
- For an IBM Process Server deployment environment, these topologies work best:
  - Remote Messaging, Remote Support, and Web - Four-cluster topology pattern

    Note: Do not use this pattern for a Standard configuration unless it includes IBM Business Monitor.
  - Remote Messaging and Remote Support - Three-cluster topology pattern
- For an IBM Process Center deployment environment, these topologies work best:
  - Single Cluster topology pattern
  - Remote Messaging - Two-cluster topology pattern

The pattern that you choose for your Process Server or Process Center network deployment environment must be based upon your unique requirements.

As you plan for your production environment, consider carefully the advantages and disadvantages of each of the common topology patterns.

Condensed topology pattern selection criteria

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

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

Table 22. Considerations for selecting a topology for your deployment environment

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

Related concepts:

A topology is the physical layout of the deployment environment. The product features and default usage

depends on your choice of topology pattern.

"Topologies of a network deployment environment" on page 37
A topology is the physical layout of the deployment environment. You can create the topology that best addresses your business needs by choosing one of the patterns provided by IBM or by creating your own customized pattern.

"Choosing a stand-alone or network deployment environment" on page 27
Choose a stand-alone environment to evaluate the product or to support development of applications and services. Choose a network deployment environment when your production environment needs additional features such as capacity, availability, scalability, and failover support.

Topology patterns and supported product features

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

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

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

Table 23 shows the relationship between the topology patterns and product features.

Table 23. Available supplied patterns and their relationship to product features

Topology pattern	Number of clusters	Description	Supported BPM products and features	Default status
Single Cluster	1	Messaging, application deployment target, and application support functions are contained in a single cluster. This topology pattern is useful for synchronous messaging, proof of concept, or application testing environments. A Single Cluster topology pattern is ideal for limited hardware. Because all of the components are installed in the same cluster, fewer physical machines are required.	Supported by the following products, or any combination of these products: • IBM Business Process Manager Advanced and Standard configurations • IBM Business Monitor	

Topology pattern	Number of clusters	Description	Supported BPM products and features	Default status
Remote Messaging	2	This topology pattern separates the messaging environment from the application deployment target and the application support functions. Use this topology pattern when message throughput is a critical requirement for your daily operation. This topology pattern is highly recommended for asynchronous messaging and transactional systems. In a Process Center network deployment environment, the Remote Messaging topology pattern is often enough to meet processing needs.	Supported by the following single product installations: • IBM Business Process Manager Advanced and Standard configurations	
Remote Messaging and Remote Support	3	This topology pattern separates messaging, Common Event Infrastructure (CEI), application deployment target, and application support functions into distinct clusters. Most businesses can use this topology pattern to support their deployment environments because it is designed for performance and isolation of transactional processing from messaging and other support functions.	Supported by the following single product installations: • IBM Business Process Manager Advanced and Standard configurations	This topology pattern is the default topology pattern for IBM Business Process Manager production environments. This topology pattern is the default topology pattern for the following installations: • IBM Business Process Manager Advanced (including deployment environments) • IBM Business Process Manager Standard
Remote Messaging, Remote Support, and Web	4	This topology pattern defines one cluster for application deployment, one remote cluster for the messaging infrastructure, one remote cluster for supporting applications, and one remote cluster for web application deployment (Business Process Choreographer Explorer, Business Space, and Business Rules Manager).	Supported by the following products, or any combination of these products: • IBM Business Process Manager Advanced and Standard configurations • IBM Business Monitor	This topology pattern is the default topology pattern for an IBM Business Monitor installation. For the Standard configuration, choose this pattern only if your topology includes IBM Business Monitor.

Configurable components for each configuration

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

Table 24 shows the components for each of the different configurations.

Table 24. IBM Business Process Manager components

Component	Advanced	WebSphere ESB	Standard	Express
Process Server	X		X	X
Process Server messaging engine (service integration bus)	X		X	X
Performance Data Warehouse	X		X	X
Performance Data Warehouse messaging engine (service integration bus)	X		X	X
Common database	X	X		
Common event infrastructure (CEI) messaging engine (service integration bus)	X	X		
Service Component Architecture (SCA)	X	X		
SCA application messaging engine (service integration bus)	X	X		
Business Space	X	X		
Business Process Choreographer	X			
Business Process Choreographer messaging engine (service integration bus)	X			
Business Process Choreographer Explorer	X			

Related concepts:

"Considerations for selecting a topology" on page 46
Selecting an appropriate topology for your deployment environment depends upon several factors.

"Topologies of a network deployment environment" on page 37
A topology is the physical layout of the deployment environment. You can create the topology that best addresses your business needs by choosing one of the patterns provided by IBM or by creating your own customized pattern.

Determining whether to create a standardized or customized network deployment environment

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

Reasons to create a standardized network deployment environment

If the IBM-supplied topology patterns (packaged as templates in IBM Business Process Manager) address all or most of your business processing needs, use the Deployment Environment wizard to create a standardized network deployment environment. The Deployment Environment wizard generates clusters and servers according to several predefined topology patterns, and configures multiple components across them all at the same time.

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

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

Reasons to create a customized network deployment environment

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

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

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

Load balancing and failover with IBM HTTP Server

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

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

1. Review the information in Selecting a front end for your WebSphere Application Server topology to learn more about using a proxy server front end.
2. Implement a web server plugin. Install and configure the plugin as part of the local deployment manager profile. These WebSphere Application Server topics provide more information:
   - Implementing a web server plug-in
   - Installing and configuring web server plug-ins (refer to Scenario 5 in this topic)
3. Configure Secure Sockets Layer (SSL) between the deployment manager for WebSphere Application Server and the IBM HTTP Server administration server. You must configure the Application Server to accept a self-signed certificate from IBM HTTP Server so SSL connections are accepted and transactions are completed. See Configure SSL between the IBM HTTP Server Administration Server and the deployment manager for more information.
4. Customize the Process Server or Process Center cluster so that the configuration file points to the web server, the Process Server is in online mode, and communication is enabled for HTTP over SSL or HTTP Secure (HTTPs). See "Customizing the Process Server/Process Center cluster to work with a web server" on page 643.

Planning your database configuration

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

Databases and your IBM Business Process Manager topology

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

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

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

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

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

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

You can incorporate configuration information for the database into the profile creation process by one of the following methods:
- Referencing a database design file
- Setting database configuration parameters with the Profile Management Tool or the `manageprofiles` command-line utility

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

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

For more information about the database design tool, see Creating database design files using the database design tool.

Process Center database and Process Server database

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

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

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

Database time zone and character set considerations

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

Time zone considerations

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

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

Character set considerations for Oracle databases

IBM BPM requires that the Oracle database be Unicode enabled. In Oracle, there are two character set parameters that influence how CHAR and NCHAR types are stored in the database:
1. For database character set (CHAR types), IBM BPM requires AL32UTF8.
2. For national character set (NCHAR types), IBM BPM requires UTF8.

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

For a more detailed description of the character set parameters, refer to the related Oracle information.
Related information:
Oracle Database Migration Assistant for Unicode

Choosing how and when to configure the common database

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

You can create the common database before, during, or after you create the IBM Business Process Manager profile.
- Before you configure IBM Business Process Manager:
  - Edit and run the default scripts that come with IBM Business Process Manager. You can use the default scripts to create only the common database and Business Process Choreographer tables.
  - Use the design file that was created using the database design tool (DDT). See Creating database design files by using the database design tool.
- After you configure IBM Business Process Manager:
  - Use the Profile Management Tool to configure IBM Business Process Manager to work with the tables in the database as you create the profile. You can create and configure the database tables

during profile creation, or delay creation and configuration until after the profile has been created. Use the Profile Management Tool to generate database scripts that you can use to create and configure the database tables. These generated scripts are ready to use. No editing is required.

– Use the design file that you created using the database design tool (DDT). See Creating database design files by using the database design tool.

Supported database types

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

See Table 25 for a list of the databases that are supported with IBM Business Process Manager.

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

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

The DB2 Express database is included in the IBM Business Process Manager software, and can be installed and configured automatically when you install IBM Business Process Manager. To install DB2 Express, you must have administrative privileges (Administrator or root user) and must not have an existing DB2 database server on the system.

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

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

- Linux Instance user name and password: bpminst and bpminst1

- Linux Fenced user name and password: bpmfenc and bpmfenc1

- Linux Administration server (DAS) user name and Password: bpmadmin and bpmadmin1

- Windows Administrative user name and Password: bpmadmin and bpmadmin1

Each database is represented by a parameter *dbType* which is a character string. The *dbType* parameter is used as a parameter in **manageprofiles** command-line utility. The values of *dbType* for the supported databases are shown in Table 25.

Table 25. Supported database types, their associated dbType values and restrictions

Supported database	dbType value	Restrictions and notes
DB2 Express	DB2_UNIVERSAL	Used as the default database type for a stand-alone profile.
DB2® Universal	DB2_UNIVERSAL	
DB2 Data Server	DB2_DATASERVER	Available for download from: 9.7 GA level Fixpacks

Table 25. Supported database types, their associated dbType values and restrictions (continued)

Supported database	dbType value	Restrictions and notes
DB2 for z/OS	DB2UDBOS390	If you are using DB2 for z/OS as your database management system, you must configure the database and database objects by using the **createDB.sh** script. The installation wizard is not able to create a database of this type. .
Microsoft SQL Server	Microsoft SQL Server JDBC 1.2 and 2.0 = MSSQLSERVER_MICROSOFT	Microsoft SQL Server JDBC 3.0 is also supported, though not listed as a separate database type on the Profile Management Tool. If you are using Microsoft SQL Server JDBC 3.0 as your database management system, selecting. a database type of Microsoft SQL Server JDBC 2.0 will support version 3.0 **Note:** The Process Server and Performance Data Warehouse databases cannot be shared in the CommonDB. **Note:** If a locale different from Latin must be specified, then the **createDatabase.sql** script can not be used. A different locale that is case-insensitive must be specified. **Important:** You must configure XA transactions after the database is installed and before you start the server. Failure to configure the XA transactions can result in an error during server start up. See "Configuring XA transactions" on page 560.
Oracle	ORACLE	The installation wizard is not able to create a database of this type for Oracle.

A second parameter used in file path and file naming conventions is *feature*, which indicates which of the various databases is under consideration. Table 26 lists the databases and the associated *feature* parameter.

Table 26. Databases and their associated feature name.

Database	Feature
Process Server	ProcessServer
Performance Data Warehouse	PerformanceDW
Common database	CommonDB
Enterprise service bus logger mediation database	EsbLoggerMediation

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

Where *feature* can be:
- CommonDB
- EsbLoggerMediation
- ProcessServer
- PerformanceDW

The creation of most JDBC providers is performed during profile creation, based upon the database parameters that you provide. However, the Business Process Choreographer JDBC providers may be

created later using scripts or the administrative console. If a suitable JDBC provider is located on the server or cluster, the existing JDBC provider is used and the creation of a new one is omitted.

Related concepts:

"JDBC drivers and locations" on page 58
The following tables list the supported JDBC drivers. The first table contains the names and locations of the JDBC drivers that are provided with the product. The second table contains the names of the JDBC drivers that are supported but not provided with the product.

Related information:

⏏ Using a DB2 database to hold custom user registry data

Database naming restrictions

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

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

* Process Server - BPMDB
* Performance Data Warehouse - PDWDB

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

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

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

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

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

* Installation Manager provides an option to create the profile, but assigns default values to the database names.
* You are using the Typical installation path and you choose to install DB2 Express, which is packaged with the installation images. Database names are assigned the default values.

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

IBM Business Process Manager Standard installation

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

* The Process Server and Performance Data Warehouse components must be in separate databases.
* The Process Server and Performance Data Warehouse components do not support case-sensitive databases. These databases must be case-insensitive.

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

Data sources for IBM Business Process Manager

Data sources provide a link between applications and relational databases. The data sources that you use are affected by whether you set up a stand-alone environment or a network deployment environment.

Applications use a data source to obtain connections to a relational database. A data source is analogous to the Java EE Connector Architecture (JCA) connection factory, which provides connectivity to other types of enterprise information systems (EIS).

A data source is associated with a Java Database Connectivity (JDBC) provider, which supplies the driver implementation classes that connect with a specific type of database. Application components interact directly with the data source to obtain connection instances to your database. The connection pool that corresponds to each data source provides connection management.

You can create multiple data sources with different settings, and associate them with the same JDBC provider. For example, you might use multiple data sources to access different databases within the same database application. In IBM Business Process Manager, JDBC providers must implement one or both of the following data source interfaces. Use these interfaces to run the application in a single-phase or two-phase transaction protocol.

ConnectionPoolDataSource
> A data source that supports application participation in local and global transactions, except two-phase commit transactions. When a connection pool data source is involved in a global transaction, the transaction manager does not provide transaction recovery. The application is responsible for providing the backup recovery process if multiple resource managers are involved.

The following tables provide examples of typical stand-alone environment setups and typical deployment environment setups:

Table 27. Typical stand-alone environment setup

Datasource	Component	Scope	JNDI Name
WBI DataSource	CommonDB	Node	jdbc/WPSDB
SCA Application Bus ME data source	SCA ME	Server	jdbc/com.ibm.ws.sib/nlNode01.server1-SCA.APPLICATION.localhostNode01Cell.Bus
event	CEI	Server	jdbc/cei
CEI ME data source	CEI ME	Server	jdbc/com.ibm.ws.sib/nlNode01.server1-CEI.*cellName*.BUS

Table 28. Typical deployment environment setup

Datasource	Component	Scope	JNDI Name
WBI DataSource	CommonDB	Cell	jdbc/WPSDB
SCA Application Bus ME data source	SCA ME	Cluster	jdbc/com.ibm.ws.sib/clusterone-SCA.APPLICATION.enduranceTestCell01.Bus
event	CEI	Cluster	jdbc/cei
CEI ME data source	CEI ME	Cluster	jdbc/com.ibm.ws.sib/clusterone-CEI.*cellName*.BUS

JDBC drivers and locations

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

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

Table 29. Supported JDBC drivers and locations that are provided with the product

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

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

Table 30. Supported JDBC drivers that are not provided with the product

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

Related concepts:

"Supported database types" on page 54
Choosing a database depends on your operating system and on the features that you will use with IBM Business Process Manager.

Related information:

 Detailed hardware and software requirements for IBM Business Process Manager Advanced

Identifying required database administrator tasks

If you want to perform some types of database creation and configuration tasks in IBM Business Process Manager, you must be a database administrator (DBA).

Database selection

Choosing how to configure your database

Database privileges and security considerations

* "Database privileges" on page 59
* Identifying necessary security authorizations

Profile creation

* Prerequisites for creating or augmenting profiles
* Creating a stand-alone environment

- Configuring the software after a Custom installation to create one or more Deployment manager and Custom (managed node) profiles

-

  Tip: If you use the deployment environment feature, you can use a database other than the default database server as your database product. The user ID that you provide for the **User name to authenticate with the database** field on the database configuration panels must have DBA privileges.

Database configuration
- Create the database and tables before profile creation or augmentation
  – Creating database design files by using the database design tool
- Create the database and tables after profile creation or augmentation
  – Creating the Common database and tables after profile creation or augmentation
  – Creating database design files by using the database design tool
- "Planning to configure the messaging engine database" on page 77

Relevant links
- Configuring Business Space

Nonadministrative user considerations

Remember the database details so that you can enter them during the installation.

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

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

To install as a nonadministrative user, you have the following choices:

Note: If you choose to use the DB2 Express database included (and optionally installed) with the product, you must meet the following criteria:
- Uninstall any other versions of DB2 from the system
- Install IBM Business Process Manager as a nonadministrative or nonroot user

Database privileges

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

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

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

Table 31. Database privileges

Header	Minimum privileges required to create objects in the database	Minimum privileges required to access objects in the database
DB2	The user ID needs CREATETAB authority on the database and CREATETS to create the table space. The user ID also needs CREATEIN and DROPIN privilege on the schema. The user ID needs system privileges CREATEDBA and CREATEDBC. The user ID also needs ALTER, DELETE, INDEX, INSERT, REFERENCES, SELECT, and UPDATE privileges on the created tables.	The user ID needs SELECT, INSERT, UPDATE, and DELETE privileges on the tables. The user ID also needs EXECUTE ON PROCEDURE on stored procedures.

Refer to Table 32 on page 61 for detailed DB2 database privileges for IBM Business Process Manager and WebSphere Enterprise Service Bus components. |
| DB2 for z/OS | The user ID needs CREATETAB authority on the database and CREATETS to create the table space. The user ID also needs CREATEIN and DROPIN privilege on the schema. To create storage groups for the database, the user ID needs CREATESG, CREATEDBA, and CREATEDBC system privileges. The user ID also needs ALTER, DELETE, INDEX, INSERT, REFERENCES, SELECT, and UPDATE privileges on the created tables. | The user ID needs SELECT, INSERT, UPDATE, and DELETE privileges on the tables. The user ID also needs EXECUTE ON PROCEDURE on stored procedures.

Refer to Table 33 on page 62 for detailed DB2 for z/OS database privileges for IBM Business Process Manager and WebSphere Enterprise Service Bus components. |
| Oracle | The user ID needs sufficient privilege to create relational tables and indexes in the data store schema. The database also needs a space quota in the default table space of the owner of that schema. | The user ID needs the SESSION privilege to connect to the database. If the same user ID owns both the data store schema, and the component that is connecting to the database, the user ID has sufficient privilege to manipulate the tables. Otherwise, the user ID needs SELECT, INSERT, UPDATE, ALTER, and DELETE object privileges on the tables that make up the data store, and the DROP ANY TABLE system privilege to enable the use of the TRUNCATE TABLE statement. The user ID also requires the CREATE INDEX privilege.

You must create the Oracle database using a UTF-8 character set, which supports the other customer character sets that are supported by IBM Business Process Manager.

See Table 34 on page 63 for detailed Oracle database privileges for IBM Business Process Manager and WebSphere Enterprise Service Bus components. |
| SQL Server | The user ID ideally requires DB OWNER privileges on the data stores used for IBM Business Process Manager. | Configure the SQL Server for SQL Server and Windows authentication so that authentication to be based on an SQL server login ID and password. The user ID must be the owner of the tables, or a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.

See Table 35 on page 65 for detailed SQL Server database privileges for IBM Business Process Manager and WebSphere Enterprise Service Bus components. |

Table 32 describes additional DB2 database privileges for IBM Business Process Manager components.

Table 32. Detailed DB2 database privileges

Component	Installation privileges	Runtime privileges
Common DB	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, USAGE ON SEQUENCE
Business Space	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE
Business Process Choreographer	CREATE TABLE, ALTER TABLE, CREATE VIEW, CREATE TABLESPACE, CREATE USER, CREATE PROCEDURE	SELECT, UPDATE, DELETE, INSERT
Business Process Choreographer Explorer reporting function	CREATE TABLE, ALTER TABLE, CREATE VIEW, CREATE TABLESPACE, CREATE FUNCTION	SELECT, UPDATE, DELETE, INSERT, and EXECUTE ON FUNCTION privileges on the functions listed in the `createFunctionsSql_Observer.sql` script
Messaging Engines	CREATE TABLE, CREATE INDEXTYPE	SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE **Note:** Messaging Engines use the TRUNCATE TABLE SQL statement, which may require the DROP ANY TABLE privilege. Refer to Database privileges.
Process Server Performance Data Warehouse	Required to create the database: • CREATEDBA • CREATEDBC Required to populate the database with our schemas and stored procedures: • CREATETAB • CREATEIN • DROPIN Additional required privileges on the created tables: • ALTER • DELETE • INDEX • INSERT • REFERENCES • SELECT • UPDATE	Required privileges on the tables in the Process Server and Performance Data Warehouse databases: • DELETE • INSERT • SELECT • UPDATE The runtime user must have EXECUTE ON PROCEDURE privileges on the six stored procedures in the `createProcedure_ProcessServer.sql` script. The runtime user requires all of the above listed privileges on the Performance Tracking Server database as well. In addition, the user also must be able to create new tables in the Performance Tracking database, requiring the CREATETAB privilege. To read the system metadata tables, the SELECT permission is required on **syscat.tables**, **syscat.views**, **syscat.columns**, **syscat.tabconst**, and **sysibm.sysdummy1**.

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

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

Table 33 describes additional DB2 for z/OS database privileges for IBM Business Process Manager components.

Table 33. Detailed DB2 for z/OS database privileges

Component	Installation privileges	Runtime privileges
Common DB	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, USAGE ON SEQUENCE
Business Space	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, USAGE ON SEQUENCE
Business Process Choreographer	CREATE TABLE, ALTER TABLE, CREATE VIEW, CREATE TABLESPACE, CREATE USER, CREATE PROCEDURE	SELECT, UPDATE, DELETE, INSERT
Business Process Choreographer Explorer reporting function	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, INSERT, CREATE TABLESPACE, CREATE FUNCTION	SELECT, UPDATE, DELETE, INSERT, and EXECUTE ON FUNCTION privileges on the functions listed in the `createFunctionsSql_Observer.sql` script.
Messaging Engines	CREATE TABLE, CREATE INDEXTYPE	SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE **Note:** Messaging Engines use the TRUNCATE TABLE SQL statement, which may require the DROP ANY TABLE privilege. Refer to Database privileges.

Table 33. Detailed DB2 for z/OS database privileges (continued)

Component	Installation privileges	Runtime privileges
Process Server Performance Data Warehouse	Required to create the database: • CREATESG • CREATEDBA • CREATEDBC Required to populate the database with our schemas and stored procedures: • CREATETS • CREATETAB • CREATEIN • DROPIN Additional required privileges on the created tables: • ALTER • DELETE • INDEX • INSERT • REFERENCES • SELECT • UPDATE	Required privileges on the tables in the Process Server and Performance Data Warehouse databases: • DELETE • INSERT • SELECT • UPDATE The runtime user must have EXECUTE ON PROCEDURE privileges on the six stored procedures in the `createProcedure_ProcessServer.sql` script. The runtime user requires all of the above listed privileges on the Performance Tracking Server database as well. In addition, the user also must be able to create new tables in the Performance Tracking database, requiring the CREATETS and CREATETAB privileges. To read the system metadata tables, the SELECT permission is required on **sysibm.systables**, **sysibm.sysviews**, **sysibm.syscolumns**, **sysibm.syschecks**, **sysibm.sysrels**, **sysibm.systabconst**, **sysibm.systablespace**, and, **sysibm.sysdummy1**.

Table 34 describes additional Oracle database privileges for IBM Business Process Manager components.

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

Table 34. Detailed Oracle database privileges

Component	Installation privileges	Runtime privileges
Common DB	CREATE TABLE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE
Business Space	CREATE TABLE, ALTER TABLE, INSERT, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE
Business Process Choreographer	CREATE TABLE, ALTER TABLE, CREATE VIEW, CREATE TABLESPACE, CREATE USER, CREATE PROCEDURE	SELECT, UPDATE, DELETE, INSERT
Business Process Choreographer Explorer reporting function	CREATE TABLE, ALTER TABLE, CREATE VIEW, CREATE TABLESPACE, CREATE FUNCTION	SELECT, UPDATE, DELETE, INSERT, and EXECUTE ON FUNCTION privileges on the functions listed in the `createFunctionsSql_Observer.sql` script.

Table 34. Detailed Oracle database privileges (continued)

Component	Installation privileges	Runtime privileges
Common Event Infrastructure (CEI)	CREATE TABLE, CREATE INDEXTYPE, ALTER TABLE, CREATE VIEW, ALTER SESSION, SELECT, UPDATE, DELETE, INSERT, CREATE TABLESPACE, CREATE PROFILE CREATE ROLE, CREATE PROCEDURE, CREATE TEMPORARY TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE PROCEDURE
Messaging Engines	CREATE TABLE	SELECT, UPDATE, DELETE, INSERT, DROP ANY TABLE **Note:** Messaging Engines use the TRUNCATE TABLE SQL statement, which may require the DROP ANY TABLE privilege. Refer to Database privileges.
Process Server	CREATE TABLE, SELECT, INSERT, UPDATE, ALTER LOCK TABLE, DELETE TABLE, DROP TABLE, CREATE INDEX, CREATE VIEW, DROP VIEW, CREATE PROCEDURE, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, DROP TABLE, DROP VIEW, CREATE SEQUENCE
Performance Data Warehouse	CREATE TABLE, SELECT, INSERT, UPDATE, ALTER LOCK TABLE, DELETE TABLE, DROP TABLE, CREATE INDEX, CREATE VIEW, DROP VIEW, CREATE PROCEDURE, CREATE SEQUENCE, CREATE USER, ALTER USER, CREATE TABLESPACE	SELECT, UPDATE, DELETE, INSERT, CREATE VIEW, CREATE PROCEDURE, DROP TABLE, DROP VIEW, CREATE SEQUENCE All schemas or users creating or migrating IBM BPM profiles must have access to the **DBMS_LOCK** package. Set the **Execute** permission on the **DBMS_LOCK** package for the Oracle user or schema using the performance database as shown in the following example: `GRANT execute ON DBMS_LOCK TO <schema_name>` In this example, *schema_name* is the user ID that is used for the performance database.

Syntax for assigning rights to an Oracle user:

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

Table 35 on page 65 describes additional SQL Server database privileges for IBM Business Process Manager components.

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

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

- db_ddladmin
- db_datawriter
- db_datareader

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

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

Table 35. Detailed SQL Server database privileges

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

Business Process Choreographer Explorer reporting function is not supported on SQL Server.

Syntax for assigning rights to an SQL user:

```
USE database_name
GO
CREATE USER user_name FOR LOGIN user_name WITH DEFAULT_SCHEMA=user_name
GO
CREATE SCHEMA user_name AUTHORIZATION user_name
GO
EXEC sp_addrolemember 'db_ddladmin', 'user_name';
EXEC sp_addrolemember 'db_datareader', 'user_name';
EXEC sp_addrolemember 'db_datawriter', 'user_name';</p>
```

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

Related reference:

⟶ Database privileges in WebSphere Application Server

User ID or schema name privileges

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

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

Scenario for a single user ID or schema name privileges

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

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

Table 36. Scenario: Single user ID or schema

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

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

Scenario 1 for multiple user ID or schema name privileges

In this scenario, you use a schema name that is the same as the user ID privileges, but you do not use the default schema name or default user ID privileges. This single user ID can access all of the database and also create all needed tables. The following examples show scenario 1 privileges:
- Schema name: dog
- Schema name for SCA.SYSTEM ME : dogSYS
- Schema name for SCA.APP ME: dogAPP
- Schema name for Event ME: dogEvent
- User ID to create schemas: dog
- User ID to select, insert, update, and delete schemas: dog

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

Table 37. Scenario 1: Multiple user ID or schema

Database tables	Database name with DB2	Schema name	User ID to create tables	User ID to select, insert, update, and delete rows
Common database tables	You supply this value in the • Installation wizard • Profile Management Tool • Silent install • Silent profile creation	This schema name is the same as the user ID that is used to select, insert, update, and delete rows.	This value is the same as the user ID that is used to select, insert, update, and delete rows.	You supply this value in the • Installation wizard • Profile Management Tool • Silent install • Silent profile creation

Scenario 2 for multiple user ID or schema name privileges

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

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

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

Table 38. Scenario 2: Multiple user ID or schema

Database tables	Database name with DB2	Schema name	User ID to create tables	User ID to select, insert, update, and delete rows
Common database tables	You supply this value twice: 1. In table creation scripts 2. During the IBM Business Process Manager configuration with one of the following: • Administrative console • Installation wizard • Profile Management Tool • Silent install • Silent profile creation **Restriction:** If you execute the installer first, then you supply the value once because the generated scripts already contain the correct schema name and user ID values.	The table creation scripts need to be modified with the schema name that allows reading and writing rows.	The table creation script needs to be modified with the user ID that allows table creation.	You supply the user ID during profile creation through one of the following: • Installation wizard • Profile Management Tool • Silent install • Silent profile creation

Scenario 3 for multiple user ID or schema name privileges

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

- Schema name: waterCom
- Schema name for common tables: waterCom
- Schema name for SCA.SYSTEM ME: waterSYSME
- Schema name for SCA.APP ME: waterAPPME
- Schema name for Event ME: waterEventME
- Schema name for ESBMessaging tables: waterESB
- User ID to create schemas: milk
- User ID to select, insert, update, and delete schemas:

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

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

Table 39. Scenario 3: Multiple user ID or schema

Database tables	Database name with DB2	Schema name	User ID to create tables	User ID to select, insert, update, and delete rows
Common database tables	You supply this value in the • Installation wizard • Profile Management Tool • Silent install • Silent profile creation	This schema name is the same as the user ID that is used to select, insert, update, and delete rows.	This value is the same as the user ID that is used to select, insert, update, and delete rows.	You supply the user ID during profile creation through one of the following: • Installation wizard • Profile Management Tool • Silent install • Silent profile creation
Messaging tables	You supply this value with the definition of each messaging engine.	The table creation scripts must include the schema name that is used to select, insert, update, and delete rows.	This value is the same as the user ID that is used to select, insert, update, and delete rows.	You supply this value during the creation of the messaging engine. Select the **Create Table** option during the messaging engine configuration.

Planning your component-specific database configurations

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

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

To plan your database configuration, you must know the components that you will use. Table 40 lists the IBM Business Process Manager components that require a database table, and the default names of the databases where the tables associated with these components are stored.

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

Table 40. Databases that are required by individual components

Server component	Database (default name)	Notes
Process Server	BPMDB	The Process Server tables are used to hold, store, and track the data for the process application repository.
Performance Data Warehouse	PDWDB	
Application scheduler	CMNDB (the common database)	You must create the common database before you start IBM Business Process Manager. You must configure the CMNDB tables either during the startup of the deployment manager or stand-alone server or before you start the deployment manager or stand-alone server.

Table 40. Databases that are required by individual components (continued)

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

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

Table and schema creation

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

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

Tip: You configure the common database during profile creation.

Table 41. Table creation based on database provider

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

Table 42. Schema creation based on database provider

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

Table 42. Schema creation based on database provider (continued)

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

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

Planning to configure the common database

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

The IBM Business Process Manager common database is used by the following product components:
- Recovery
- Relationship service
- Enterprise Service Bus (ESB) Logger Mediation Primitive

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

Supported database types

The common database can use the following database products:

Table 43. Supported database products

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

Table 43. Supported database products (continued)

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

User ID privileges

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

Database Management Service instances

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

Configuration actions during profile creation

You can use one of the following options to install the common database:
- Installer
- Profile Management Tool
- Silent installation
- Scripts

Within each of these options are several more choices.

Installer
> Use the Installer if you are going to create your profiles when you install your software. You can install your database products during installation, but you are limited in the types of database products that you can install. To use a supported database product that is not in Table 44, you must use the Profile Management Tool to create your deployment manager.

Table 44. Installer options

Option	Databases you can use
Typical: stand-alone profile	DB2 Express
Typical: all other profiles	DB2 Express
Customized: stand-alone profile	• DB2 Universal • DB2 Data Server • Oracle

Table 44. Installer options (continued)

Option	Databases you can use
Customized: all other profiles	• DB2 Universal • DB2 Data Server • Oracle

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

Profile Management Tool

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

Table 45 lists the databases supported by typical (default) profiles and customized profiles.

Table 45. Profile Management Tool options

Options	Databases you can use
Typical: stand-alone profile	• DB2 DataServer • DB2 Universal • DB2 for z/OS • Microsoft SQL Server • Oracle DB2 for z/OS, Oracle and Microsoft SQL databases must exist so that the Profile Management Tool can configure them.
Typical: all other profiles	• DB2 DataServer • DB2 Universal • DB2 for z/OS • Microsoft SQL Server • Oracle DB2 for z/OS, Oracle and Microsoft SQL databases must exist so that the Profile Management Tool can configure them.
Customized: stand-alone profile	Any of the "Supported database types" on page 72
Customized: all other profiles	Any of the "Supported database types" on page 72

Silent Installations

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

Scripts

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

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

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

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

When you run the scripts, you also perform the following tasks:
- Create a database, if appropriate (valid only for a local database), depending on your choices in the Database configuration panel in the Profile Management Tool.

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

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

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

SQL scripts

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

You can find SQL scripts for each common database client in the following location:
- *install_root*/dbscripts/CommonDB/*dbType* after you install IBM Business Process Manager

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

The SQL script naming convention is:
- For a component-specific script: **createTable_*componentName*.sql**, for example **createTable_Recovery.sql**
- For a component-independent script: **createTable.sql**.

The following table shows the script naming convention.

Table 46. Common database script naming convention

Type of script	Script name
Component specific	*scriptName_componentName*.sql
Component independent	*scriptName*.sql

JDBC provider

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

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

Data source name:
- **WPS DataSource**

Data source JNDI name:
- **jdbc/WPSDB**

Restrictions

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

`Create new database` is disabled for the following database types:
- DB2 for z/OS
- Oracle
- Microsoft SQL Server

Tables

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

Table 47. Tables created by IBM Business Process Manager components

Component	Table names	Scripts
Relationship	Dynamic table, created at runtime	**createTable_Relationship MetadataTable.sql**
Common database	SchemaVersionInfo	**createTable_CommonDB.sql**
ESB Logger Mediation	MSGLOG	**createTable_ESBLogger Mediation.sql**

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

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

Exported scripts

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

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

Database scripts are exported to the

profile_root/dbscripts/CommonDB/*dbType*/*dbName* directory.

Planning to configure the Common Event Infrastructure database

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

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

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

Planning to configure the messaging engine database

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

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

The default database name for the SCA messaging engine is SCADB. For the other messaging engines, the default database name is MEDB. The default schema name is IBMWSSIB.

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

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

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

Supported database types

The messaging engine database can use the following database products:

Table 48. Supported database products

Database Types	Considerations
DB2 Express	Used as the default database type for a stand-alone profile.
DB2 Universal	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 Data Server	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 for z/OS v8 DB2 for z/OS v9	**Important:** When creating a profile for a server that uses DB2 for z/OS v9, the server must be able to connect to the DB2 database.Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
Microsoft SQL Server (Microsoft)	

Table 48. Supported database products (continued)

Database Types	Considerations
Oracle	You need system database administrator privileges to create the database, tables, and schemas. If you do not have these privileges, you might receive errors when you create or access the tables and schemas.

User ID privileges

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

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

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

Database Management Service (DBMS) instances

Each messaging engine has its own database or schema:
- One is used to host each messaging engine for the Service Component Architecture system bus.
- Another is used to host each messaging engine for the Service Component Architecture application bus.
- Another is used to host each messaging engine for the Common Event Infrastructure bus.

The following list contains the naming conventions for the JDBC data source that the messaging engine uses to interact with the database:
- System bus: *<node><server>|<cluster>*-SCA.SYSTEM.*<cell>*.Bus
- Application bus: *<node><server>|<cluster>*-SCA.APPLICATION.*<cell>*.Bus
- CEI bus: *<node><server>|<cluster>*-CEI.*cellName*.BUS

Configuration actions during profile creation

Network deployment

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

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

The following administrative tasks are performed during profile creation:
- Remote Destination Location:
  - configSCAAsyncForServer, configSCAJMSForServer (remoteMELocation is true)
  - configSCAAsyncForCluster, configSCAJMSForCluster (remoteMELocation is true)
- Local Destination Location:
  - configSCAAsyncForServer, configSCAJMSForServer

– configSCAAsyncForCluster, configSCAJMSForCluster

For more information about these tasks, see "configSCAAsyncForCluster command" and "configSCAAsyncForServer command".

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

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

SQL scripts

No SQL scripts are created as part of the product. You can use existing base WebSphere Application Server scripts to create database and tables if necessary. To create the MEDB manually before it is configured, use the **Application servers** > *servername* > **Service Component Architecture** panel of the administrative console.

JDBC provider

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

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

Data source names
- System bus: : `_(node.server|cluster)-SCA.SYSTEM.`*cell*`.Bus/`*cel/cluster/server/node*
- Application bus: `_(node.server|cluster)-SCA.APPLICATION.`*cell*`.Bus/`*cell/cluster/server/node*
- Common Event Infrastructure: `_(node.server| cluster-CEI.`*cellName*`.BUS/`*cluster/server/node*

Data source JNDI names
- System bus: `jdbc/com.ibm.ws.sib/(node.server|cluster)-SCA.SYSTEM.`*cell*`.Bus/`*cell/cluster/server/node*
- Application bus: `jdbc/com.ibm.ws.sib/(node.server|cluster)-SCA.APPLICATION.`*cell*`.Bus/`*cell/cluster/server/node*
- Common Event Infrastructure: `Jdbc/ com.ibm.ws.sib /(node.server|cluster)-CEI.`*cellName*`.BUS/`*cluster/server/node*

Restrictions

There are no known restrictions.

Tables

For information on the tables, see the topic "Data stores" in the WebSphere Application Server Network Deployment information center.

Exported scripts

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

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

Related information:

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

Planning to configure the logger mediation database tables for WebSphere Enterprise Service Bus

You can find logger mediation database table specifications for WebSphere ESB. The specifications contain information about supported database types, script names and their locations, profile creation configuration actions, schema upgrades, and user ID privileges.

The logger mediation database tables are used by the Message Logger mediation primitive in WebSphere ESB. The Message Logger primitive stores message information in the common database. The common database is the default for the WebSphere ESB logger mediation database, but you can use an external database. During the profile augmentation phase, the system creates the *ESB_MESSAGE_LOGGER_QUALIFIER* variable, which is set to the value of the chosen common database schema qualifier.

The database is created automatically for a stand-alone configuration. You can use Data Definition Language (DDL) files to use additional databases for a stand-alone server configuration or for a network deployment environment.

For a stand-alone configuration that uses a DB2 for z/OS database, or for a managed node or deployment manager in a network deployment configuration, you must create the WebSphere ESB database and storage groups first. Then you can run the configuration script for IBM Process Server for z/OS.

Supported database types

The WebSphere ESB logger mediation database can use the following database products:

Table 49. Supported database products

Database Types	Considerations
DB2 Express	Used as the default database type for a stand-alone profile.
DB2 Universal	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 Data Server	Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
DB2 for z/OS v8 DB2 for z/OS v9	**Important:** When creating a profile for a server that uses DB2 for z/OS v9, the server must be able to connect to the DB2 database.Used as the database in network deployment configurations. Optionally, can be used as the database in stand-alone server configurations.
Microsoft SQL Server (Microsoft)	

Table 49. Supported database products (continued)

Database Types	Considerations
Oracle	You need system database administrator privileges to create the database, tables, and schemas. If you do not have these privileges, you might receive errors when you create or access the tables and schemas.

User ID privileges

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

Database Management Service (DBMS) instances

The common database is used for both stand-alone environments and network deployment environments at cell scope. However, you can manually create as many other instances as you require. Each message logger mediation primitive can be configured to use a different data source and therefore a different database.

Configuration actions during profile creation

For the stand-alone profiles and deployment manager profiles, the WebSphere ESB logger profile executes the **createTable** common database script in the common database. For more information, see "Common database specifications".

Stand-alone profile
> In a default stand-alone environment, a DB2 database named `EsbLogMedDB` is automatically created.

Network deployment environment
> The default WebSphere ESB database is not automatically selected during network deployment profile creation. You must select the default database or one of the supported databases.

SQL scripts

The **createTable_ESB.sql** SQL script is located in the `install_root`/dbscripts/CommonDB/`DBTYPE` directory.

The **createMessageLoggerResource.jacl** and **removeMessageLoggerResource.jacl** scripts are located in the `install_root`/bin directory, and can be used to create or delete tables in the requested database type.

JDBC provider

The common database JDBC provider and data source are used by default:

Data source name:
- **WPS DataSource**

Data source JNDI name:
- **jdbc/WPSDB**

You can configure the Message Logger mediation to use a different data source.

Restrictions

There are no known restrictions.

Tables

The WebSphere ESB logger mediation database uses the MSGLOG table in the common database. However, you can choose not to use the common database, and use an external database instead.

Exported scripts

The database scripts are exported to the *install_root*/dbscripts/CommonDB/*DBTYPE*/*dbName* directory.

Schema upgrade scripts

No schema upgrade involved for MSGLOG table. When you migrate to IBM Business Process Manager V6.1, IBM Business Process Manager continues to use the MessageLogger databases used in prior releases. There is no support to migrate this data into the IBM Business Process Manager common database.

If you want to maintain a single location for message information, you can perform one of the following tasks:

- Manually move the data from the old database to the new database
- Continue to use the old database
- Use the `createMessageLoggerResource.jacl` script to move the data.

Planning error prevention and recovery

You can develop error-prevention and recovery strategies to minimize the impact of system and application errors.

Topics in *Planning error prevention and recovery* include links to a variety of resources, such as information center topics, technical articles and IBM Redbooks that provide detailed information on development processes and system configuration patterns designed to take advantage of WebSphere system recovery capabilities.

Overview of error prevention and recovery

The error prevention and recovery information describes how to avoid problems that might cause system failures, and provides or points to information on how to recover from system failures that can result from both ordinary and extraordinary circumstances.

IBM Business Process Manager is a middleware server optimized for enabling the running and management of business process management (BPM) and service-oriented architecture (SOA) solutions. IBM Business Process Manager is built on the foundational capabilities of WebSphere Application Server.

Middleware systems run under various conditions, not all of which are traditionally "good path" conditions. Many of the key features within IBM Business Process Manager are intended to deal with the uncertainty that might arise through what can appear to be normal operations.

Assumptions and expectations

Before using the information about system failure and recovery as described in the *Planning error prevention and recovery* section, read the following list of assumptions:

- You are familiar with IBM Business Process Manager and the basic architectural principles upon which it is built and the basic kinds of applications that it runs.
- You have a foundational understanding of integration projects, including how to plan for and implement integration projects.
- Unless otherwise specified, the information about system failure and recovery is relevant to version 6.1.0 and later of IBM Business Process Manager.

Note: The information contained in the *Planning error prevention and recovery* section assume a remote messaging and remote support pattern, which consists of three separate clusters, one for the IBM Business Process Manager and one each for the messaging engine and CEI event server.

Planning error prevention

As with all IT endeavors, planning against and practicing for extreme situations will increase the possibility for a successful recovery.

There are a number of required considerations associated with preparing for system and application recovery. These considerations can be grouped under two categories as follows:
- Error prevention practices as part of application design
- Error prevention practices as part of development process

Error prevention as part of application design

Including error prevention practices as part of your application design means implementing specific design techniques and using the capabilities of the product to help prevent system and application errors.

A strong system of governance, complete with architectural and design guidelines and appropriate standards combined with reviews and checkpoints are essential to building the right kind of application.

Error prevention practices as part of application design include the following:
- Implementing design considerations for exceptions and faults
- Implementing an error handling strategy that uses existing IBM Business Process Manager error handling capabilities and tools
- Creating connectivity groups and using module application design techniques

Connectivity groups:

A connectivity group represents a specific pattern of behavior found in an SCA module.

Create connectivity groups to represent the possible request sources for the system.

In a connectivity group you:
- Put all the logic to get the inbound data into one module

  This is also true for outbound data when it is going to an external system or legacy system
- Put all the logic to connect and transform the data into one module

  All the other modules can now use a standard set of interfaces and not have to worry about extra transformations.

The connectivity group will not contain stateful component types like long-running business processes and Business State Machines. These connectivity groups provide encapsulation and isolation of the specific endpoint's integration requirements. Commonly, WebSphere ESB mediation modules are used for this purpose as they represent convenient ways implement "infrastructure" related tasks.

The concept of connectivity groups also provide a convenient way to quiesce the system in case there is a need for recovery. Because the connectivity group module is stateless, the module can be temporarily stopped thus cutting off the inbound flow of new events while the system finishes processing the events it has.

Note: If you want to stop the flow of inbound events, then the connectivity modules **should not** support inbound and outbound in the same module (even though the same EIS system may have both inbound and outbound). If inbound and outbound support are in the same module, then the outbound is turned off with the inbound. This may cause internal work to stop from completing. Consider separating inbound and outbound in this case.

When the system is recovered and able to process new work, these modules can be restarted.

The module that is outlined in the following screen capture is considered part of a connectivity group.
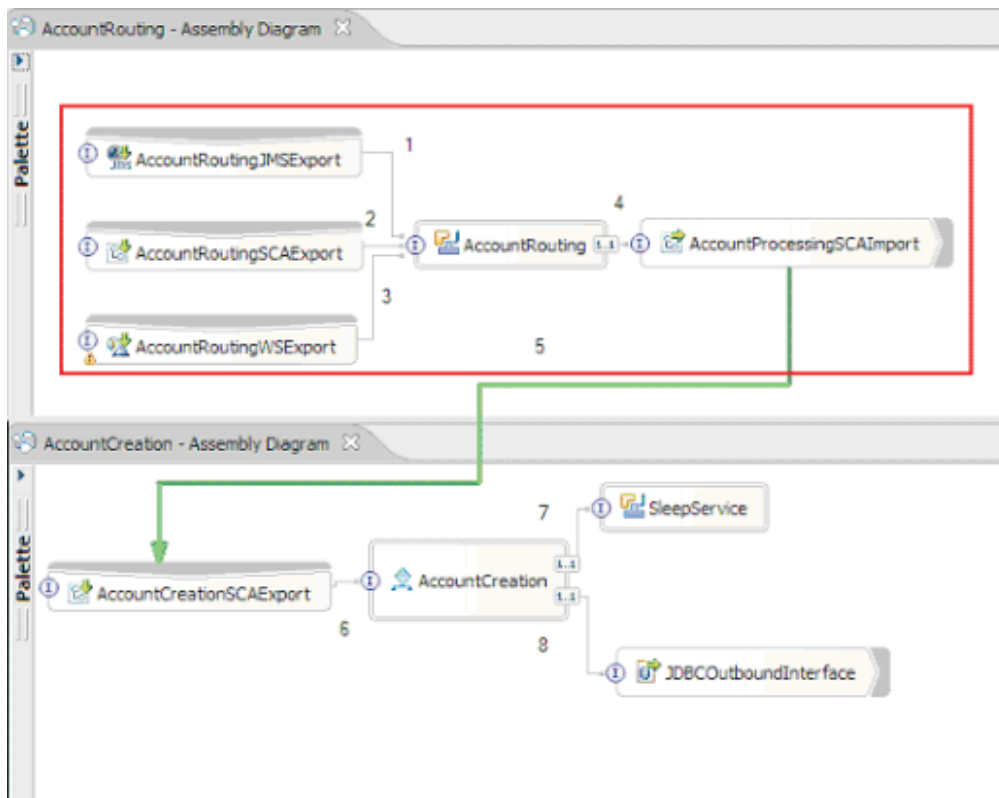
Connectivity groups can be used for input from an external source or an existing system such as SAP or CICS®. Or for new work from a web browser-based clients.

Application design considerations for exceptions and faults:

You need to consider your application design so that it can take advantage of the error handling and fault processing capabilities in IBM Business Process Manager.

In order to create a comprehensive error handling strategy, solution architects need to understand how WebSphere Process Server and WebSphere ESB represent declared and undeclared exceptions.

The SCA programming model provides two types of exceptions:
• Service Business Exceptions

Service Business Exceptions are checked exceptions declared in a business method's function signature (WSDL faults or Java throws). Service Business Exceptions identify error conditions that are anticipated by the application or service. These exceptions are sometimes referred to as "checked exceptions"

An example is an **InvalidSymbolException** for a stock quote service. Such exceptions are wrapped by ServiceBusinessException and passed back to the client.

- Service Runtime Exceptions

Also known as "system exceptions" service runtime exceptions are not declared in the method signature. In general, they represent error conditions that are not anticipated by the application, such as a **NullPointerException** in a Java Component.

These exceptions are wrapped by **ServiceRuntimeException** and passed back to the client, which can interrogate the **ServiceRuntimeException** to determine the cause.

Note: When working at the SCA level these exceptions are sometimes referred to as faults. However, when using Java code they are usually referred to as exceptions.

When a **ServiceRuntimeException** is thrown from a component, the current transaction will be rolled back.

Service Business Exception handling:

Service Business Exceptions represent known and declared exceptions anticipated by the application or service.

Service Business Exceptions are defined on the service interface.

Component developers should take care to declare the possible exceptions that may be thrown, so that the consuming service can handle them. For example, a business fault to a banking application would include "Invalid Account Number", or "Insufficient Funds" as *business exceptions*. So the application that calls the service needs to include logic to handle a situation where they have passed in an invalid account number, or where they tried to transfer $100 but there was only $50 in the account. These are the types of business errors that a calling application is designed to handle. The IBM Business Process Manager business exceptions are returned to the client to catch and handle appropriately.

When handling business service exceptions, service consumers should implement the client such that it will perform one of the following actions for a declared business exception:

1. Catch the exception and create the appropriate Service Business Exception for the calling application.

   This could mean including the original exception in the new exception (wrapping it). This is most often done when the calling module does not have the same Business Exceptions as the service that it is calling. Here is an example of the flow catching an exception and creating a Service Business Exception for the calling application:

   a. **Module A** has SBE "MoneyTransferFailed"
   b. **Module B** has SBE "InsufficientFunds"
   c. **Module A** calls **Module B** and gets "InsufficientFunds" exception
   d. **Module A** must create a new exception "MoneyTransferFailed", which may have a place where a string defining the original error of insufficient funds can be included.

2. Catch the exception and perform alternate logic.

Service Runtime Exception handling:

Service Runtime Exceptions are undeclared exceptions. In general, they represent error conditions that are not anticipated by the application.

Service Runtime Exceptions are used to signal an unexpected condition in the runtime.

Component developers can handle Service Runtime Exceptions in the following ways:

1. Catch them and perform some alternative logic.

   For example, if one partner is not able to service a request perhaps another one might.

2. Catch the exception and "re-throw" it to your client.

3. Remap the exception to a business exception.

   For example, a timeout for a partner may result in a business exception that indicates most of the request was processed but there was one piece of the request that was not completed and should be retried later or tried with different parameters.

If an exception is not caught, the exception is passed on to the component that called the current component. This call chain continues back to the original caller in the chain. For example, **Module A** calls **Module B** and **Module B** calls **Module C** and then **Module C** throws an exception, **Module B** might or might not catch the exception. If **Module B** does not catch the exception, then the exception travels back to **Module A**.

When a **ServiceRuntimeException** is thrown from a component, the current transaction will be rolled back. This type of exception processing is repeated for all components in the chain. For example, if a **ServiceRuntimeException** is thrown from **Module C**, that transaction will be marked for rollback. Then the exception is thrown to **Module B**, where if it is not caught and another transaction is present, that transaction also will be rolled back. Component developers can use quality of service (QoS) qualifiers to control whether invocations occur in the current transaction or a new transaction. So, if **Module A** calls **Module B** and **Module B** is part of a new transaction, then **Module A** can "catch" a ServiceRuntimeException from **Module B** and continue processing, without **Module A's** transaction rolling back.

Note: Because runtime exceptions are not declared as part of the interface, component developers should attempt to resolve the exception and thus prevent a runtime exception from inadvertently being propagated to the client if the client is a user interface.

You should be aware that the contents of the rolled back transaction can vary, depending on the nature of the transaction. For example, long-running BPEL processes can be segmented into many smaller transactions. Asynchronous request and response calls are broken out of a transaction automatically (otherwise the calling application might have to wait a long time for the response).

In instances where a transaction is broken into multiple asynchronous calls (as opposed to one large transaction), the initial work for the transaction would rollback at the occurrence of a ServiceRuntimeException. However, the response from the asynchronous call is sent from a different transaction, and because the response from the asynchronous call would have no place to go, an event is created in the Failed Event Manager (FEM).

The following list is of 4 current subclasses of **ServiceRuntimeException**:

1. **ServiceExpirationRuntimeException**

   This exception is used to indicate that an asynchronous SCA message has expired. Expiration times can be set using the RequestExpiration qualifier on a service reference.

2. **ServiceTimeoutRuntimeException**

   This exception is used to indicate that the response to an asynchronous request was not received within the configured period of time. Expiration times can be set using the ResponseExpiration qualifier on a service reference.

3. **ServiceUnavailableException**

   This exception is used to indicate that there was an exception thrown while invoking an external service via an import.

4. **ServiceUnwiredReferenceRuntimeException**

   This exception is used to indicate that the service reference on the component is not wired correctly.

Chapter 6. Installing and configuring IBM Business Process Manager Standard

IBM Business Process Manager Standard can be installed and configured on a single server (known as a stand-alone configuration), or you can distribute the components across multiple systems (known as a network deployment configuration).

Preparing to install and configure the software

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

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

Table 50. Preparing for installation and configuration

Task	Where to find information	Result after completing the task
Review hardware and software requirements	Depending on your IBM BPM configuration, visit: • IBM Business Process Manager Advanced system requirements • IBM Business Process Manager Standard system requirements • IBM Business Process Manager Express system requirements	You understand the system requirements necessary to support your IBM Business Process Manager installation.
Prepare your operating system	AIX Preparing AIX systems for installation Linux Preparing Linux systems for installation Solaris Preparing Solaris systems for installation Windows Preparing Windows systems for installation	You have prepared the operating system of each workstation to be used.

Table 50. Preparing for installation and configuration (continued)

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

Preparing operating systems for product installation

Before you can install IBM Business Process Manager Standard, you must prepare your operating system. The configuration depends on the type of operating system you are using.

Before preparing the installation environment, complete the following tasks:
- Disable the firewall if you have a firewall running on the system where you plan to install IBM Business Process Manager Standard.
- Ensure that your user login provides access to your DB2 or Oracle database commands.
- Complete additional tasks specific to your operating system.

Attention: While installing IBM Business Process Manager 7.5.0 on **Windows Server 2003** or **Windows 7 (English)** and switching the locale to Czech in the IBM Process Portal**Preference -> Interface language** the text displays with corrupt characters on several panels of the IBM Process Designer. The text displays with corrupt characters even if IBM Process Designer is started with the Czech locale.

Tip: To resolve this issue change system settings as follows.
- In **Windows Server 2003**: **Regional and Language Options -> Advanced -> Language for non-Unicode programs -> set to 'Czech'**
- In **Windows 7: Regional and Language Options -> Administrative Tab -> Change system locale... (under 'Language for non-Unicode programs') -> Select 'Czech**'

.

Preparing AIX systems for installation

Before you can install IBM Business Process Manager, you must prepare your AIX® operating system.

Because WebSphere Application Server is a prerequisite of IBM Business Process Manager, you must complete the required preparation steps in the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

Note: However, please note the following points related to the WebSphere Application Server installation:
- The WebSphere Application Server Network Deployment V7.0 that is installed by IBM Business Process Manager V7.5 uses IBM Installation Manager to install and does not use InstallShield Multiplatform (ISMP). Ignore any prerequisite instructions that refer specifically to running ISMP.
- The WebSphere Application Server Network Deployment V7.0 that is installed by IBM Business Process Manager V7.5 uses IBM Installation Manager to install fix packs and interim fixes and does not use the WebSphere Update Installer. Ignore any prerequisite instructions that refer specifically to running the WebSphere Update Installer.

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system.

Refer to the following technote for additional preparation information for configuring Installation manager to run on 64-bit AIX systems: Installation Manager graphical environment issues.

Complete the following steps on your AIX system before installing IBM Business Process Manager:
1. If you are installing 32-bit WebSphere Application Server on a 64-bit operating system, ensure that the appropriate 32-bit libraries are installed on your 64-bit system.
2. Increase the maximum number of open files. The default setting is usually not enough. You can check your current maximum number of open files by using **ulimit -n**. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The ulimit requirement is dynamically calculated at installation time and might need to be larger based on the options you select.

   Before installing, run the following command:

   > **ulimit -n 8800**

   Alternatively, you can use the following steps to edit the resource limits file:
   a. Open /etc/security/limits.
   b. Edit or add the **default** section and include this line:

      > **nofiles = 8800**

   c. Save and close the file.
   d. Log off from the operating system and log in again.
3. Set the **umask** value to 077 using the following command:

   > **umask 077**

   The value 077 is the most restrictive value that the product will tolerate. You can optionally choose to set a less restrictive **umask** value for the following access levels:
   - 037 for read-only access for a group of human administrators and tools
   - 027 for read and write access for a group of human administrators and tools
   - 007 for read, write, and execute access for a group of human administrators and tools
4. Ensure that you have Mozilla Firefox installed at version 3.5.x.x or higher.
5. Before starting the data movement service, increase the number of processes configured in the AIX operating system to avoid a connection reset error. You can increase the number of processing using a command, or using the AIX interface.
   - Run the command:

```
chgdev  -l sys0 -a maxuproc='256'
```

- In the AIX interface, enter **smitty**, then select **System Environments > Change / Show Characteristics of Operating System > Number of processes allowed per user(Num.)**.

6. Complete the steps to Tune AIX systems.

7. Ensure all servers involved are set to the same time. Use the same network time protocol for all servers on all cluster nodes, including application, support, and database clusters. A time mismatch will cause erratic behavior, including duplicate system tasks.

Preparing Linux systems for installation

Before you can install IBM Business Process Manager, you must prepare your Linux operating system.

Because WebSphere Application Server is a prerequisite of IBM Business Process Manager, you must complete all the required preparation steps in the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

Note: However, please note the following points related to the WebSphere Application Server installation:

- The WebSphere Application Server Network Deployment V7.0 that is installed by IBM Business Process Manager V7.5 uses IBM Installation Manager to install and does not use InstallShield Multiplatform (ISMP). Ignore any prerequisite instructions that refer specifically to running ISMP.

- The WebSphere Application Server Network Deployment V7.0 that is installed by IBM Business Process Manager V7.5 uses IBM Installation Manager to install fix packs and interim fixes and does not use the WebSphere Update Installer. Ignore any prerequisite instructions that refer specifically to running the WebSphere Update Installer.

Ensure that you have Mozilla Firefox installed at version 3.5.x.x or higher.

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system. To install Installation Manager on Red Hat Enterprise Linux 6.0 (64-bit), see Unable to install Installation Manager on RHEL 6.0 (64-bit).

If you are planning to install IBM Business Process Manager using DB2 Express with Red Hat Enterprise Linux 6, you must have administrative privileges (root user), must not have an existing DB2 database server on the system, and you must also ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the **ipcs -l** command.

To change the values:

1. Add the following lines, in the below order, to the /etc/sysctl.conf file:

```
kernel.shmmni=4096
kernel.shmmax=4294967296
kernel.shmall=8388608
#kernel.sem=<SEMMS><SEMMNS><SEMOPM><SEMMNI>
kernel.sem=250 256000 32 4096
kernel.msgmni=16384
kernel.msgmax=65536
kernel.msgmnb=65536
```

2. Add the following lines to the end of /etc/security/limits.conf:

```
# - stack - max stack size (KB)
* soft stack 32768
* hard stack 32768
# - nofile - max number of open files
* soft nofile 65536
* hard nofile 65536
# - nproc - max number of processes
* soft nproc 16384
* hard nproc 16384
```

3. Check for the existence of a file named /etc/security/limits.d/90-nproc.conf, which overrides the **nproc** value set in the limits.conf file. If the 90-nproc.conf file exists, edit it and set the **nproc** values that are specified in the previous step.

4. Reboot your system.

Complete the following steps on your Linux system before installing IBM Business Process Manager:

1. If you are installing 32-bit WebSphere Application Server on a 64-bit operating system, ensure that the appropriate 32-bit libraries are installed on your 64-bit system.

2. If you are planning to install IBM Business Process Manager using DB2 Express with Red Hat Enterprise Linux 6 as a root user, follow the previous instructions and skip this step. Otherwise, increase the maximum number of open files to at least 8800. The default setting is usually not enough. You can check your current maximum number of open files by using **ulimit -n**. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The ulimit requirement is dynamically calculated at installation time and might need to be larger based on the options you select.

   a. Open /etc/security/limits.conf.

   b. Locate the **nofile** parameter and increase the value. If a line containing the **nofile** parameter does not exist, add the following lines to the file:

      * hard nofile 8800

      * soft nofile 8800

   c. Save and close the file.

   d. Log off and log in again.

   For more information about this setting, run **man limits.conf** or see the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

3. Install the following packages for your operating system:

Option	Description
Red Hat Enterprise Linux 5	compat-libstdc++-33-3.2.3-61 compat-db-4.2.52-5.1 libXp-1.0.0-8 rpm-build-4.4.2-37.el5 **64-bit kernel only:** compat-libstdc++-296-2.96-138
Red Hat Enterprise Linux 6	ksh-version.rpm Korn shell See the detailed instructions and list of packages in Unable to install Installation Manager on RHEL 6.0 (64-bit)
SUSE Linux Enterprise Server 9.0	XFree86-libs-32bit-9 glibc-32bit-9 glib-32bit-9 gtk-32bit-9

You can also install a later release of any of these packages if there are new packages as errata. If you have additional packages that are specific to your hardware, install them.

You can use single-line commands to install dependencies (all required packages). The following commands are examples using the default package managers on supported Linux distributions.

- **Red Hat Enterprise Linux 5 (32-bit):**

  ```
  yum install compat-libstdc++-33 compat-db libXp rpm-build  RHEL 5.x
  ```

- **Red Hat Enterprise Linux 5 (64-bit):**

  ```
  yum install compat-libstdc++-33 compat-db libXp rpm-build compat-libstdc++-296
  ```

- **SUSE Linux:**

  ```
  zypper install XFree86-libs-32bit-9 glibc-32bit-9 glib-32bit-9 gtk-32bit-9
  ```

4. Set the **umask** value to 077 using the following command:

   umask 077

   The value 077 is the most restrictive value that the product will tolerate. You can optionally choose to set a less restrictive **umask** value for the following access levels:

   - 037 for read-only access for a group of human administrators and tools
   - 027 for read and write access for a group of human administrators and tools
   - 007 for read, write, and execute access for a group of human administrators and tools

5. On Red Hat Enterprise Linux 5 systems, disable SELinux, or set it to a permissive mode.

6. Restart the computer.

7. Complete the steps to Tune Linux systems.

8. Ensure all servers involved are set to the same time. Use the same network time protocol for all servers on all cluster nodes, including application, support, and database clusters. A time mismatch will cause erratic behavior, including duplicate system tasks.

Preparing Solaris systems for installation

Before you can install IBM Business Process Manager, you must prepare your Solaris operating system.

Because WebSphere Application Server is a prerequisite of IBM Business Process Manager, you must complete the required preparation steps in the Preparing the operating system for product installation topic in the WebSphere Application Server information center.

Note: However, please note the following points related to the WebSphere Application Server installation:

- The WebSphere Application Server Network Deployment V7.0 that is installed by IBM Business Process Manager V7.5 uses IBM Installation Manager to install and does not use InstallShield Multiplatform (ISMP). Ignore any prerequisite instructions that refer specifically to running ISMP.

- The WebSphere Application Server Network Deployment V7.0 that is installed by IBM Business Process Manager V7.5 uses IBM Installation Manager to install fix packs and interim fixes and does not use the WebSphere Update Installer. Ignore any prerequisite instructions that refer specifically to running the WebSphere Update Installer.

Because certain steps are specific to a version of the operating system, all steps might not apply to your environment. If no qualifier is provided for a particular step, complete the step for all versions of the operating system.

Refer to the following technote for additional preparation information for configuring Installation manager to run on Solaris systems: http://www-01.ibm.com/support/docview.wss?uid=swg24027719

Complete the following steps on your Solaris systems before installing IBM Business Process Manager:

1. If you are installing 32-bit WebSphere Application Server on a 64-bit operating system, ensure that the appropriate 32-bit libraries are installed on your 64-bit system.

2. Increase the maximum number of open files. The default setting is usually not enough. You can check your current maximum number of open files by using **ulimit -n**. The following example shows the maximum number of open files being increased to 8800, which is large enough for most systems. The ulimit requirement is dynamically calculated at installation time and might need to be larger based on the options you select.

   Before installing, run the following command:

   ulimit -Hn 8800

   Alternatively, you can use the following steps to edit the resource limits file:

   a. Open /etc/system

   b. Add the following line to the end of the file:

       set rlim_fd_max=8800

   c. Save and close the file.

   d. Log off from the operating system and log in again.

3. Set the **umask** value to 077 using the following command:

     umask 077

The value 077 is the most restrictive value that the product will tolerate. You can optionally choose to set a less restrictive **umask** value for the following access levels:

- 037 for read-only access for a group of human administrators and tools
- 027 for read and write access for a group of human administrators and tools
- 007 for read, write, and execute access for a group of human administrators and tools

4. Complete the steps to Tune Solaris systems.

5. Ensure all servers involved are set to the same time. Use the same network time protocol for all servers on all cluster nodes, including application, support, and database clusters. A time mismatch will cause erratic behavior, including duplicate system tasks.

Preparing Windows systems for installation

Before you can install IBM Business Process Manager, you must prepare your Windows operating system.

If you are planning to use DB2 Express with your IBM Business Process Manager installation, you must have administrative privileges (Administrator user) and must not have an existing DB2 database server on the system where you will perform the installation.

Because WebSphere Application Server is a prerequisite product for IBM Business Process Manager, you must complete all of the preparation tasks for WebSphere Application Server before installing IBM Business Process Manager.

Complete the following steps on your Windows system before installing IBM Business Process Manager:

1. Complete the below steps for the installation of WebSphere Application Server.

   a. Download and install a Web browser. You must install a Web browser if you want to use the launchpad application. If your system does not have a default browser or the browser has been corrupted or installed incorrectly, then you might experience errors when trying to open various hyperlinks in the installation wizard panels.

   b. If you are installing 32-bit WebSphere Application Server on a 64-bit operating system, ensure that the appropriate 32-bit libraries are installed on your 64-bit system.

   c. Stop all WebSphere Application Server-related Java processes on the machine where you are installing the product.

   d. Stop any Web server process such as the IBM HTTP Server.

   e. Stop all instances of the **process_spawner.exe** program. You can skip this step if the **process_spawner.exe** does not exist.The **process_spawner.exe** is a required binary for the WebSphere Application Server to function properly. The **process_spawner.exe** is not intended to be called directly and is a core functionality of the WebSphere Application Server on Windows.

   f. Provide adequate disk space. The amount of disk space required varies with the number of features or products installed. If you are installing the product using the installation wizard, the installation summary panel indicates the approximate amount of disk space required based on the features and products you have selected.If you plan to migrate applications and the configuration from a previous version, verify that the application objects have enough disk space. As a rough guideline, plan for space equal to 110 percent of the size of the applications.

   g. Verify that prerequisites and corequisites are at the required release levels. Although the installation wizard checks for prerequisite operating system patches, review the prerequisites on

the Supported hardware and software website if you have not done so already. Refer to the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

The WebSphere Application Server installation uses an IBM Installation Manager wizard. You can use the graphical interface of the wizard. You can also use the wizard in silent mode. If you are installing remotely and want to use the installation wizard, then you must use a connection mechanism that supports the use of graphics. If your connection does not support the use of graphics, then install the product silently.

To run the install successfully, you must have write permissions to the **USER_HOME** directory. If you do not have permission to write to **USER_HOME**, then be aware of the following install limitations:

- The Profile Management Tool GUI will not function due to a requirement to write to **USER_HOME**. You must use the **manageprofiles** on the command-line to manage profiles instead.

2. Complete the steps to Tune Windows systems.

3. Ensure all servers involved are set to the same time. Use the same network time protocol for all servers on all cluster nodes, including application, support, and database clusters. A time mismatch will cause erratic behavior, including duplicate system tasks.

Databases and installation and configuration scenarios

How databases are configured to work with the software is related to the software you are installing and the scenario you are trying to achieve.

In installation scenarios where you will be using the **Typical** installation option, you must pre-create your database, whether that database resides locally (on the same system on which you are installing IBM Business Process Manager or remotely (on a separate system). The only scenario where you are not required to pre-create your database is when you are installing the DB2 Express database that is embedded with IBM Business Process Manager. DB2 Express is embedded with IBM Business Process Manager on Windows and on Linux for Intel. If you do not already have DB2 Express installed, it is optionally installed when you run the IBM Business Process Manager installer, and the installer also creates the common and component-specific databases on DB2 Express. To install DB2 Express, you must have administrative privileges (Administrator or root user) and must not have an existing DB2 database server on the system.

Any scenario where you are not using DB2 Express for your **Typical** install is considered to be a **Typical** installation with an *existing database*, meaning you already have a database installed and you will use the *Existing database panel* of the installer to specify its configuration properties. See Configuring an existing database during a typical installation for details. To use this option, you must ensure that your database has already been created.

You and your database administrator should review the information on configuring databases in the information center prior to installing and configuring the software.

Installing IBM Business Process Manager Standard

Installing IBM Business Process Manager Standard involves acquiring the software and then installing the software files: prerequisite software, the database product to be used by IBM Business Process Manager Standard, and the IBM Business Process Manager Standard software.

Installing IBM Business Process Manager Standard

You install the software interactively from the launchpad program or silently by running Installation Manager in silent installation mode. In silent mode, the installation wizard does not display a graphical interface, but reads your responses from a response file.

Roadmap: Installing and configuring IBM Business Process Manager Standard

The IBM Business Process Manager installer and configuration programs accommodate a variety of development and production scenarios. This roadmap guides you through the most common and widely used paths for installing and configuring IBM Business Process Manager Standard in a stand-alone or network deployment environment.

The roadmap includes a section for each common installation and configuration scenario. Each section includes a table listing the activities to perform, with links to the instructions and notes about the result of completing each activity.

The sections in this roadmap describe the following scenarios:

- Typical installation.

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

  The Typical installation option is the simplest and quickest method for installing and configuring the software. It is also the ideal method for getting full Business Process Manager features and functionality (for Express, Standard, or Advanced configurations) on a single machine. Typical installation is only available from an electronic installation image (not a DVD).

  See Installing IBM Business Process Manager Standard using the **Typical installation** option.

- Custom installation for a stand-alone environment.

  The **Custom installation** option installs the software on to your system and then runs the Profile Management Tool, giving you more control over how to configure your software. If you are creating a stand-alone Process Center or Process Server environment that requires configuration features that are not available through the Typical installation path, create the stand-alone Process Center or Process Server profiles after the installation of the product binaries by using either the Profile Management Tool or the **manageprofiles** command-line utility.

  See Installing IBM Business Process Manager Standard using the **Custom installation** option and configuring one or more stand-alone profiles.

- Custom installation and configuration for a network deployment environment:

  This installation and configuration scenario provides the capacity, scalability, and robustness that is generally required of a large-scale production environment. A network deployment configuration is made up of a group of servers (known as clusters) that collaborate to provide workload balancing and failover. The servers are managed centrally, using a single administrative console known as the deployment manager.

  After custom installation, you have several configuration options.

  - Use the **configureNode** command-line utility for a simple network deployment configuration:

    The installer installs the software on to your system, and then you use the **configureNode** command to create and configure a deployment manager and one or more custom (managed node) profiles and create a pattern-based network deployment configuration. See Installing IBM Business Process Manager Standard and configuring a network deployment environment using the configureNode command.

  - Use profile configuration and the deployment environment wizard:

    The installer installs the software on to your system, and then you use the profile configuration programs (Profile Management Tool or **manageprofiles** command-line utility) to create and configure one or more deployment manager and custom (managed node) profiles and the deployment environment wizard to create a pattern-based network deployment configuration. See Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the deployment environment wizard.

  - Use profile configuration and the administrative console or wsadmin commands:

    The installer installs the software on to your system, and then you use the profile configuration programs (Profile Management Tool or **manageprofiles** command-line utility) to create and

configure one or more deployment manager and custom (managed node) profiles and the administrative console or wsadmin commands to create a customized network deployment configuration. See Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the administrative console or wsadmin commands.

For conceptual information about network deployment, see Choosing a stand-alone or network deployment environment and Planning your deployment environment.

For information about topology patterns packaged with the software and supported by the deployment environment wizard, see Overview: Deployment environment topologies and patterns.

Installing IBM Business Process Manager Standard using the Typical installation option

If you are installing from an electronic image, you can use the Typical installation option in the installer to install the software and configure a stand-alone profile automatically. In a Typical installation there is no need to run the Profile Management Tool or run **manageprofiles** after the installation.

The Typical option is the quickest way to install and configure the software. This installation option will default the majority of settings but does allow you to specify the installation path and database type.

For a Typical installation, you install Process Center and Process Server separately. You first run the Typical installation for Process Center to completion and then run the Typical installation for Process Server to completion.

For information about installing IBM Business Process Manager Standard using the **Typical** option, see Typical installation: IBM Business Process Manager Standard.

Installing IBM Business Process Manager Standard using the Custom installation option and configuring one or more stand-alone profiles

You can use the Custom installation option in the installer to install the software and configure one or more stand-alone profiles using the Profile Management Tool or the **manageprofiles** command-line utility.

Table 51. Installing IBM Business Process Manager Standard using the Custom installation option and configuring one or more stand-alone profiles

Task	Where to find information	Result after completing the task
Install the software using the Custom installation option.	For information about installing IBM Business Process Manager Standard using the Custom installation option, see Custom installation: IBM Business Process Manager Standard	The custom installation installs the product binaries for Process Center, IBM Process Designer, and Process Server on to your system. The custom installation does not create profiles during the installation.
Launch the Profile Management Tool.	If you did not launch the Profile Management Tool from the installer, you can do so now. See Starting the Profile Management Tool **Restriction:** You cannot use the Profile Management Tool to create profiles on BPM installations on 64-bit architectures except on the Linux on zSeries platform. Instead, proceed to the next step and use the **manageprofiles** command-line utility.	The Profile Management Tool is started and ready to use. Because data from the Process Center profile is used as input to the Process Server profile creation process, you should create your Process Center stand-alone profile first (before creating the Process Server stand-alone profile).

Task	Where to find information	Result after completing the task
Create one or more Process Center stand-alone profiles.	Depending on how you want to create your Process Center stand-alone profiles, see one of the following topics: • Creating Process Center stand-alone profiles using the Profile Management Tool • Creating Process Center stand-alone profiles using the manageprofiles utility	You have created a Process Center stand-alone profile. This profile defines your stand-alone server environment and it contains command files, configuration files, and log files. The profile creation process configures the database (common and component-specific) and generates the database tables required to support your stand-alone server environment. You can now create your Process Server stand-alone profiles, or you can install Process Designer.
Install Process Designer.	If you want to begin using the business process designing capabilities of the software, download and install Process Designer. See Installing IBM Process Designer	You have installed Process Designer. For information about how to use Process Designer, see the following topics: • Hiring Sample Tutorial for IBM Process Designer • Authoring environments • Process development with the Process Center
Create one or more Process Server stand-alone profiles.	If your objective is to fully configure the software, including the runtime environment, create the Process Server stand-alone profiles. Depending on how you want to create your Process Server stand-alone profiles, see one of the following topics: • Creating Process Server stand-alone profiles using the Profile Management Tool • Creating Process Server stand-alone profiles using the manageprofiles utility	The stand-alone server environment is now configured and ready to use.

If you cleared the **Run database scripts to initialize the databases** check box during profile creation, run the database scripts manually and then run the bootstrap utility to load the database with system information.	See Loading the database with system information for information about how to run the bootstrap utility.
	
	  Windows
	bootstrapProcessServerData.bat
	
	  Linux UNIX
	bootstrapProcessServerData.sh
	
	You must run the bootstrap utility prior to the first startup of a cluster member.
		You have loaded the database with system information prior to successfully starting the Process Server / Process Center.

Installing IBM Business Process Manager Standard and configuring a network deployment environment using the configureNode command

You can use the Typical or Custom installation option in the installer to install the software and then use the **configureNode** command to configure a deployment manager and one or more custom (managed

node) profiles and create a pattern-based network deployment environment. You can also use **configureNode** at a later time to add nodes to an existing environment.

The **configureNode** command creates a single cell environment using the Remote Messaging and Remote Support topology pattern, which includes the following three clusters:
- Messaging infrastructure cluster
- Supporting infrastructure cluster
- Application deployment target cluster

Table 52. Installing and configuring IBM Business Process Manager Standard using the configureNode command

Task	Where to find information	Result after completing the task
Install the software	For information about installing IBM Business Process Manager using the Typical installation option, see Typical installation: IBM Business Process Manager Standard For information about installing IBM Business Process Manager using the Custom installation option, see Custom installation: IBM Business Process Manager Standard	The installation for IBM Business Process Manager Standard installs the product binaries for Process Center and Process Designer on to your system. Process Server functionality is part of Process Center, but does not include Advanced Process Server functionality. For example, components for Business Process Choreographer, Business Rules, and Business Space are not part of IBM Business Process Manager Standard.
Use the configureNode command to create the deployment manager, one or more custom nodes, and the deployment environment.	See Creating the deployment environment using the configureNode command	You have created your Process Center or Process Server network deployment environment.

Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the deployment environment wizard

You can use the Custom installation option in the installer to install the software and use the Profile Management Tool or **manageprofiles** command-line utility to configure one or more deployment manager and custom (managed node) profiles. After profile creation, you can use the deployment environment wizard to generate a pattern-based network deployment configuration.

You can choose which pattern to select from the deployment environment wizard based on the planned usage of your environment (proof of concept, testing, or production) and the processing capabilities (such as high-availability and failover processing) required by your enterprise.

This installation and configuration scenario assumes that your business needs and requirements can be met by one of the topology patterns included with the software. If you have requirements that cannot be addressed by the patterns included with the software, you can create a customized network deployment environment using the administrative console. See "Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the administrative console or wsadmin commands" on page 7.

Table 53. Installing and configuring IBM Business Process Manager Standard using the deployment environment wizard

Task	Where to find information	Result after completing the task
Install the software using the Custom installation option.	For information about installing IBM Business Process Manager using the Custom installation option, see Custom installation: IBM Business Process Manager Standard.	The custom installation installs the product binaries for Process Center, Process Designer and WebSphere ESB on to your system. Process Server functionality is part of Process Center, but does not include Advanced Process Server functionality. For example, components for Business Process Choreographer, Business Rules, and Business Space are not part of IBM Business Process Manager Standard. You can now begin the profile creation process. The Profile Management Tool for IBM Business Process Manager Standard does not support importing a database design document.
Launch the Profile Management Tool.	If you did not launch the Profile Management Tool from the installer, you can do so now. See Starting the Profile Management Tool **Restriction:** You cannot use the Profile Management Tool to create profiles on BPM installations on 64-bit architectures except on the Linux on zSeries platform. Instead, proceed to the next step and use the **manageprofiles** command-line utility.	The Profile Management Tool is started and ready to use. Because data from the Process Center profile is used as input to the Process Server profile creation process, you should create your Process Center stand-alone profile first (before creating the Process Server stand-alone profile).
Create a Process Center deployment manager profile.	Depending on how you want to create your profiles, see one of the following topics: • Creating Process Center deployment manager profiles using the Profile Management Tool • Creating Process Center deployment manager and custom profiles using manageprofiles	You have a deployment manager profile. This profile defines your environment and it contains command files, configuration files, and log files. A deployment manager is a server that manages operations for a logical group, or cell, of other servers. The deployment manager is the central location for administering the servers and clusters.
Start the deployment manager and verify that the start operation was successful.	For information about how to start the deployment manager, see Starting deployment managers	The deployment manager server is started.

Table 53. Installing and configuring IBM Business Process Manager Standard using the deployment environment wizard (continued)

Task	Where to find information	Result after completing the task
Create the Process Center custom (managed node) profile or profiles.	This task assumes that you **are not federating nodes** as part of the profile creation process. It assumes that you will set parameters in the Profile Management Tool or **manageprofiles** command for *federating nodes later* (after creating the custom (managed node) profiles.) Depending on how you want to create your profiles, see one of the following topics: • Creating Process Center custom profiles (managed nodes) using the Profile Management Tool • Creating Process Center deployment manager and custom profiles using manageprofiles Repeat this task for each managed node.	You have your custom (managed node) profiles. These nodes are managed by and administered from the deployment manager. You can now federate the node into the Process Center deployment manager cell. The managed node contains a node agent and can contain managed servers. In a managed node, you can configure and run managed servers.
Federate the custom (managed) node to the deployment manager.	See Federating custom nodes to a deployment manager for instructions about how to use the **addNode** command to federate the Process Center custom node or nodes into the Process Center deployment manager cell.	The custom profile is federated into the deployment manager.
Using the deployment environment wizard, create the network deployment configuration.	For information about how to use the deployment environment wizard, see Creating the Process Center deployment environment using a pattern	As part of this task you need to address any deferred configuration items and then generate the Process Center deployment environment. You have created your Process Center network deployment environment.

Task	Where to find information	Result after completing the task

Create a cell for Process Server.	To run Process Server, you must create a separate cell.
	
	This involves the same sequence of steps, except that you create a Process Server deployment manager and Process Server custom (managed nodes) and then use the deployment environment wizard to create the Process Server deployment environment, as described in the following topics:
	• Creating Process Server deployment manager profiles using the Profile Management Tool
	• Creating Process Server custom profiles (managed nodes) using the Profile Management Tool
	• Creating Process Server deployment manager and custom profiles using manageprofiles
	  – Federating custom nodes to a deployment manager
	• Creating the Process Server deployment environment using a pattern
	  As part of this task, you must address any deferred configuration items and then generate the Process Server deployment environment.
		You have created your Process Server network deployment environment.

Perform manual database configuration steps.	If you have deferred configuration steps, you must complete additional database configuration before starting the clusters in your deployment environment. See Completing the database configuration for your network deployment environment	You have completed the required manual database configuration activities for the network deployment environment.

Load the database with system information.	See Loading the database with system information for information about how to run the bootstrap utility.
	
	  Windows
	bootstrapProcessServerData.bat
	
	  Linux **UNIX**
	bootstrapProcessServerData.sh
	
	You must run the bootstrap utility prior to the first startup of a cluster member.
		You have loaded the database with system information prior to successfully starting the Process Server / Process Center.

Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the administrative console or wsadmin commands

The topology patterns packaged with the software and implemented using the deployment environment wizard are intended to address a broad spectrum of business processing requirements. However, if you have a scenario that the topology patterns do not address sufficiently, you can use the administrative console to create a customized network deployment environment.

Note: This scenario is intended for users who have an advanced understanding of how to configure product components and functionality using the administrative console. Before you embark on the installation and configuration scenario described in this section, consider using the deployment environment wizard to create your network deployment environment.

Command assistance is available for a subset of administrative console actions. When available, command assistance displays the wsadmin scripting command for the last console action you performed. You can then use this data to create wsadmin scripts that automate certain administrative tasks. For more information on command assistance, see Administrative console actions with command assistance.

Table 54. Installing and configuring IBM Business Process Manager Standard using the administrative console

Task	Where to find information	Result after completing the task
Install the software and configure the required profiles.	Complete the tasks listed in "Installing IBM Business Process Manager Standard, configuring deployment manager and custom profiles, and using the deployment environment wizard" on page 5 up to the task *Using the deployment environment wizard, create the network deployment configuration.* Instead of using the deployment environment wizard, you will create a customized deployment environment manually using the administrative console.	The custom installation for IBM Business Process Manager Standard installs the product binaries for Process Center and Process Designer on to your system. Process Server functionality is part of Process Center, but does not include Advanced Process Server functionality. For example, components for Business Process Choreographer, Business Rules, and Business Space are not part of IBM Business Process Manager Standard. You have created the deployment manager and custom managed nodes and have federated the nodes into the deployment manager cell. You can now use the administrative console to create servers, server clusters, and the components that will comprise your network deployment environment.
Create and configure servers and clusters using the administrative console.	See Creating and configuring servers and clusters using the administrative console	You have created the servers and server clusters for your environment.
Configure a server or cluster as a Process Server.	For information about using the administrative console to configure your server or server cluster as a Process Server, see Configuring a server or cluster as a Process Server You can also configure a server or cluster as a Process Server using wsadmin. See the configureProcessServer command	You have configured the servers or server clusters as Process Servers.
Configure the Performance Data Warehouse component on a server or cluster.	For information about using the administrative console to configure Performance Data Warehouse on a server or cluster, see Configuring the Performance Data Warehouse component on a server or cluster You can also configure Performance Data Warehouse on a server or cluster using wsadmin. See the configurePerfDW command	You have configured Performance Data Warehouse on the Process Server.

Table 54. Installing and configuring IBM Business Process Manager Standard using the administrative console (continued)

Task	Where to find information	Result after completing the task
Set up the Messaging Service.	For information about using the administrative console to configure the Messaging Service, see Setting up the messaging server environment	You have set up the messaging server environment.
Configure the JNDILookup web service.	For information about using the administrative console to configure the JNDILookup web service, see Configuring the JNDILookup Web Service	You have configured the JNDILookup Web Service.
Load the database with system information.	See Loading the database with system information for information about how to run the bootstrap utility. **Windows** bootstrapProcessServerData.bat **Linux** **UNIX** bootstrapProcessServerData.sh You must run the bootstrap utility prior to the first startup of a cluster member.	You have loaded the database with system information prior to successfully starting the Process Server / Process Center.

Typical installation: IBM Business Process Manager Standard

Use the Typical installation option to install IBM Business Process Manager Standard in environments for development, quality assurance, and non-clustered staging and production. This installation option sets most settings by default, but allows you to specify the installation path and database type. You must perform a Typical installation from an electronic installation image (not a DVD).

Understand the usage scenarios for a Typical installation to make sure that the installation environment that is created by this option will meet your needs. Select the Typical option to maintain a personal copy of IBM Business Process Manager on your system. In this way, you can develop integration services or business processes locally, or you can contribute artifacts using the export and import functions in the product. A Typical install is also recommended for proof of concept work and for learning about the features and functions of the product.

Important: For typical installations, the bit architecture of the product must match the bit architecture of the system. If you are on a 32-bit system, a 32-bit version of the product is installed. If you are on a 64-bit system, a 64-bit version of the product is installed. To install a 32-bit product on a 64-bit system, you must use the Custom installation option.

Tip: To use an existing database server rather than the embedded DB2 Express database, prepare the configuration information for DB2, Microsoft SQL Server, or Oracle before you begin this procedure. To use this option, you must ensure that your databases have already been created. For DB2, the databases must be created with at least a 32K page size.

If you are planning to install IBM Business Process Manager using DB2 Express with Red Hat Enterprise Linux 6, you must have administrative privileges (root user), must not have an existing DB2 database server on the system, and you must also ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the **ipcs -l** command.

Windows To install or run IBM Business Process Manager Standard on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click `launchpad.exe` and select **Run as administrator**.

Important: Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager Standard.

This task describes the procedure for installing IBM Business Process Manager Standard using the Typical install option from the launchpad.

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

1. Optional: If you are connected to the Internet, the typical installation upgrades your product to the recommended interim fix level automatically. If you want these upgrades to be installed from a local directory instead, or if you want to specify the fix level, you can use a properties file to tell the Installation Manager where to find the upgrades and which upgrades to install.

   Create the following file:

   - **Linux** **UNIX** `<user.home>/bpm_updates.properties`

     Note: <user.home> refers to the users home directory.

   - **Windows** `C:/HOMEPATH\bpm_updates.properties`

     On Windows XP, the HOMEPATH environment variable points to `C:\Documents and Settings\user_name\`

     On Windows 7, Windows Vista, or Windows Server 2008, the HOMEPATH environment variable points to `C:\Users\user_name\`

   Note: Ensure that you have read/write access to the folders specified in the `bpm_updates.properties` file.

   The file uses the ifix prefix. The prefix must be followed by a dot. The part of the name after the prefix and the dot can be anything you want, which enables you to point to multiple locations for ifix upgrades. The locations can be either local directories or URLs. For example:

   `ifix.1=C:\bpmUpdates`

   `ifix.2=C:\WASUpdates`

2. Enter one of the following commands to start the launchpad:

   Linux **UNIX** `extract_directory/launchpad.sh`

   Windows (from a command line) `extract_directory\launchpad.exe`

3. Click **Typical installation** on the Welcome page.
4. Select one of the following options to install:
   - **Process Center**: The Process Center is a shared development environment that is used to build process applications. The Process Center includes the Process Designer development tool.
   - **Process Server**: The Process Server is the location where process applications are deployed.

5. Provide the information for the option you chose to install:
   - **Process Center**: If you selected the **Process Center**, specify the following information:
     – **Hostname**
     – **Location** for the installation
     – Installation location for Process Designer
   - **Process Server**: If you selected the **Process Server**, specify the following information:
     – **Hostname**
     – **Location** for the installation
     – Environment **Type**
     – Environment **Name**

   If you do not select **Use this server offline**, provide the following information for the Process Center that this server is to connect:
     – **Hostname**
     – **Port**
     – **User name**
     – **Password**

   If you select **Use this server offline**, the **Hostname**, **Port**, **User name**, and **Password** fields are disabled.

   Attention: If you are installing DB2 Express, your install location cannot contain National Language Strings (NLS).

6. Click **Next**.

7. Select a database:
   - If you install an embedded DB2 Express database, the database will be installed and fully configured. The default values are:
     – `Linux` Instance user name and password: `bpminst` and `bpminst1`
     – `Linux` Fenced user name and password: `bpmfenc` and `bpmfenc1`
     – `Linux` Administration server (DAS) user name and Password: `bpmadmin` and `bpmadmin1`
     – `Windows` Administrative user name and Password: `bpmadmin` and `bpmadmin1`

     Important: If the default password does not comply with the password policy on your operating system (such as Windows 2008), you must use Custom Installation instead of Typical Installation so that you can specify a password that meets the password policy on your system.

     Restriction: User names must not contain National Language Strings (NLS).
   - If you use an existing database, click the **Setup instructions** link for database-specific field descriptions. See the information about configuring an existing database during a typical installation for the required field values.

8. Click **Next**.

9. On the Licenses page, verify the installation options and read the license agreement. If you agree to the terms of the license agreement, click **I have read and accepted the license agreement and notices** and click **Next**.

10. Click **Install Software**.

    Important:

    If you receive the following warning message during the prerequisite checking, use the platform-specific steps below to increase the **ulimit** number.

```
Current system has detected a lower level of ulimit than the recommended value of recommended_value. Please increase tl
Shutdown your installer. If you are a root user open a command prompt and issue ulimit -n recommended_value and then re
```

The required value is calculated based on the version of WebSphere Application Server, the feature packs, and the configuration that you are installing.

a. Set the maximum number of open files using the following steps: AIX

   1) Open /etc/security/limits.

   2) Edit or add the **default** section and include this line:

     nofiles = *recommended_value*

   3) Save and close the file.

   4) Log off and log in again.

Linux

   1) Open /etc/security/limits.conf.

   2) Locate the **nofile** parameter and increase the value. If a line containing the **nofile** parameter does not exist, add the following lines to the file:

     * **hard nofile** *recommended_value*

     * **soft nofile** *recommended_value*

   3) Save and close the file.

   4) Log off and log in again.

Solaris

   1) Open /etc/system and add the following line to the end of the file:

     set rlim_fd_max=8800

   2) Save and close the file.

   3) Log off and log in again.

b. Restart the computer.

c. Restart the installer.

The result of a Typical installation is a fully functional Process Center or Process Server. If you install a Process Center on Windows you will also have a Process Designer installed. The Typical installation creates either a Process Center or a Process Server profile and optionally allows you to launch the first steps console after a successful completion.

Launch the first steps console to verify your installation, start or stop the server, access the administrative console, access the process center console, start the Profile Management Tool, and access the product documentation.

Related tasks:

"Creating databases manually" on page 556
IBM Business Process Manager Standard requires a Process Server database and Performance Data Warehouse database.. You must make sure the databases are available before you install IBM Business Process Manager Standard using a typical installation.

"Configuring an existing database during a typical installation" on page 107
Use the information in this topic to determine the correct database values for configuring your existing database server during a typical installation.

"Custom installation: Standard" on page 109
Use the Custom installation option to install IBM Business Process Manager Standard if your deployment environment requires additional cluster nodes or if your business requires complex deployment environment and database topologies.

Configuring an existing database during a typical installation:

Use the information in this topic to determine the correct database values for configuring your existing database server during a typical installation.

If you are planning to use an existing database server during a typical installation of IBM Business Process Manager Standard, you must confirm that the databases already exist and that they are empty. For DB2, the databases must be created with at least a 32K page size.

During a typical installation, you need to specify information for your existing database server. You need to specify the user name and password for database authentication, the database server host name and port, and the Process Server and Performance Data Warehouse databases.

On the Database page, provide the required information for that database:
- Table 55 lists the database configuration fields for DB2
- Table 56 lists the database configuration fields for Microsoft SQL Server
- Table 57 on page 108 lists the database configuration fields for Oracle

Table 55. Required database configuration fields for DB2 Database

Field	Action needed
Hostname	Accept the default value of `localhost` or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 50000 or enter the correct server port number.
Username	Enter the user name to authenticate with the database.
Password	Enter a password to authenticate with the database.
Process Center or Process Server database name	Enter the name for the Process Server database.
Performance Data Warehouse database name	Enter the name for the Performance Data Warehouse database.

Table 56. Required database configuration fields for Microsoft SQL Server

Field	Action needed
Hostname	Accept the default value of `localhost` or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1433 or enter the correct server port number.
Process Center or Process Server database name	Enter the Process Center or Process Server database name.
Performance Data Warehouse database name	Enter the Performance Data Warehouse database name.
Select the **Apply Windows authentication** option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the remaining fields are made inactive.	

Common database	For the Common database, enter values for the following parameters:
	• `User name`
	  Enter the Common database user name.
	• `Password`
	  Enter a password to authenticate with the Common database.

Table 56. Required database configuration fields for Microsoft SQL Server (continued)

Field	Action needed

Process Server database	For the Process Server database, enter values for the following parameters:
	• **User name**
	  Enter the Process Server database user name.
	• **Password**
	  Enter a password to authenticate with the Process Server database.

Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters:
	• **User name**
	  Enter the Performance Data Warehouse database user name.
	• **Password**
	  Enter a password to authenticate with the Performance Data Warehouse database.

Table 57. Required database configuration fields for Oracle

Field	Action needed
Hostname	Accept the default value of `localhost` or enter the correct database server host name. For example, the IP address.
Port	Accept the default value of 1521 or enter the correct server port number.
Process Center database System Identifier	Enter the correct Process Center database System Identifier.

Process Server database	For the Process Server database, enter values for the following parameters:
	• **User name**
	  Enter the Process Server database user name.
	• **Password**
	  Enter a password to authenticate with the Process Server database.

Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters:
	• **User name**
	  Enter the Performance Data Warehouse database user name.
	• **Password**
	  Enter a password to authenticate with the Performance Data Warehouse database.

Related concepts:

"Configuring a Microsoft SQL Server database" on page 560
You can create a stand-alone profile for use with Microsoft SQL Server.

Related tasks:

"Typical installation: IBM Business Process Manager Standard" on page 103
Use the Typical installation option to install IBM Business Process Manager Standard in environments for development, quality assurance, and non-clustered staging and production. This installation option sets most settings by default, but allows you to specify the installation path and database type. You must

perform a Typical installation from an electronic installation image (not a DVD).

"Creating databases manually" on page 556

IBM Business Process Manager Standard requires a Process Server database and Performance Data Warehouse database.. You must make sure the databases are available before you install IBM Business Process Manager Standard using a typical installation.

"Creating and configuring the DB2 for z/OS database" on page 583

If your deployment environment relies on a remote DB2 for z/OS database, use the procedures and reference information in this section to help you configure the database and create the database tables.

Custom installation: Standard

Use the Custom installation option to install IBM Business Process Manager Standard if your deployment environment requires additional cluster nodes or if your business requires complex deployment environment and database topologies.

If you are planning to install IBM Business Process Manager using DB2 Express with Red Hat Enterprise Linux 6, you must have administrative privileges (root user), must not have an existing DB2 database server on the system, and you must also ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the **ipcs -l** command.

Windows To install or run IBM Business Process Manager Standard on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click launchpad.exe and select **Run as administrator**.

Important: Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager Standard.

1. Access the media in one of the following ways, depending on whether you are installing from the product DVD or from images downloaded from Passport Advantage®.

   - If you are installing from the product DVD, insert the product disk labeled IBM Business Process Manager Standard into the disk drive. Mount the disk drive if necessary. If autorun is enabled on your workstation, the launchpad program automatically opens, and you can proceed to step 2 If autorun is not enabled on your workstation, enter one of the following commands to start the launchpad manually:

     – Linux   UNIX   *mount_point*/launchpad.sh

     – Windows (from a command line) *DVD_root*\launchpad.exe

   - If you are installing from images downloaded from Passport Advantage, perform the following steps:

     a. Go to the directory into which you extracted the images.

     b. Enter one of the following commands to start the launchpad:

       – Linux   UNIX   *extract_directory*/launchpad.sh

       – Windows (from a command line) *extract_directory*\launchpad.exe

2. Click **Custom installation** on the Welcome page.

   Important: If you are on a 64-bit system, you might receive the following message:

   ```
   Your operating system failed the launchpad prerequisites check. The following 32-bit GTK Library for running IBM Ins
   ```

   If you see this message, your server does not have the 32-bit version of the GTK library installed, or the library is an incorrect version. Update your server with the correct version of the 32-bit GTK library, using the DVD or official web site of your operating system, before you continue the installation.

3. Optional: Click **Help System installation** to Install the product Help System and documentation on your computer. The Help System is an Eclipse framework for displaying the documentation. After the installation, you can display the documentation in stand-alone mode or in server mode. This option is only supported on Windows XP 32 bit, Linux 32 bit, and AIX PPC64.

4. Click **Install as administrative user** to install as an administrative user. If you are in the Administrator group on Windows, or if you are a root user on Linux or UNIX systems, you can install as an administrative user. If you are not an administrative user, or if you want to install to your own user name without administrative privileges, clear this check box.

5. Click **Install**. When you install IBM Business Process Manager Standard, the required WebSphere Application Server Network Deployment and Feature Pack for XML, are automatically installed.

6. From the Install Packages page of the Installation Manager, select the **IBM Business Process Manager Standard** package.

7. Optional: Select IBM DB2 Express to install and use an embedded DB2 Express database.

8. Click **Next** to continue.

   If you receive any of the following error messages during the prerequisite checking, address the product incompatibility issues, click **Back**, fix the problem, and click **Next** to continue:

   - If you selected to install DB2 Express and DB2 is already installed on this system, you will receive the following error message:

     ```
     DB2 is already installed on this system. Either uninstall existing DB2 or deselect DB2 Express from being installed.
     ```

   - If you select to install any of the IBM Business Process Manager Version 7.5.1 products to the same WebSphere Application Server location as any other IBM Business Process Manager Version 7.5.1 or equivalent, prior release products, you receive the following error message:

     ```
     IBM Business Process Manager V7.5.1 can not coexist with the following offerings:
     ```

     Installation Manager lists all incompatible offerings for your reference.

   If you receive the following warning message during the prerequisite checking, use the platform-specific steps below to increase the **ulimit** number.

   ```
   Current system has detected a lower level of ulimit than the recommended value of recommended_value. Please increase th
   Shutdown your installer. If you are a root user open a command prompt and issue ulimit -n recommended_value and then re
   ```

   The required value is calculated based on the version of WebSphere Application Server, the feature packs, and the configuration that you are installing.

   a. Set the maximum number of open files using the following steps: **AIX**
      1) Open /etc/security/limits.
      2) Edit or add the **default** section and include this line:

         nofiles = *recommended_value*
      3) Save and close the file.
      4) Log off and log in again.

      Linux

      1) Open /etc/security/limits.conf.
      2) Locate the **nofile** parameter and increase the value. If a line containing the **nofile** parameter does not exist, add the following lines to the file:

         * hard nofile *recommended_value*

         * soft nofile *recommended_value*
      3) Save and close the file.
      4) Log off and log in again.

      Solaris

      1) Open /etc/system and add the following line to the end of the file:

> **set rlim_fd_max=8800**

   2) Save and close the file.

   3) Log off and log in again.

   b. Restart the computer.

   c. Restart the installer.

9. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.

10. On the Location page, the **Create new package group** option is selected by default. Select a location for the shared resources directory and a location for the Installation Manager and click **Next**. The Install Packages wizard checks your system for operating system prerequisites. If you are at a higher major release of a supported operating system, or if the operating system itself is not on the supported list, you might encounter a warning. You can continue with the installation, but the installation or product operation might not succeed until you apply maintenance.

   If you see such a warning, go to the product support web pages and obtain the latest maintenance packages to apply after installation. Refer to the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

   Attention: If you are installing DB2 Express, your install location cannot contain National Language Strings (NLS).

11. Optional: If you selected to install and use an embedded DB2 Express database, specify the DB2 administrative **User name** and **Password** on the Common Configurations page. The default values are:

   - ▌ Linux ▐   Instance user name and password: `bpminst` and `bpminst1`

   - ▌ Linux ▐   Fenced user name and password: `bpmfenc` and `bpmfenc1`

   - ▌ Linux ▐   Administration server (DAS) user name and Password: `bpmadmin` and `bpmadmin1`

   - ▌ Windows ▐   Administrative user name and Password: `bpmadmin` and `bpmadmin1`

   Important: If the default password does not comply with the password policy on your operating system (such as Windows 2008), you must use Custom Installation instead of Typical Installation so that you can specify a password that meets the password policy on your system.

   Restriction: User names must not contain National Language Strings (NLS).

12. On the Summary page, review your choices before installing the IBM Business Process Manager Standard package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.

13. When the installation process is complete, a message confirms the success of the process.

   a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. You must close the Installation Log window to continue.

   b. Select **Profile Management Tool** if you want to launch the Profile Management Tool when you finish or select **None** to complete the installation.

   c. Click **Finish** to close the Installation Manager.

Launch the first steps console to verify your installation, start or stop the server or deployment manager, access the administrative console, start the Profile Management Tool, and access the product documentation.

Related tasks:

"Typical installation: IBM Business Process Manager Standard" on page 103
Use the Typical installation option to install IBM Business Process Manager Standard in environments for development, quality assurance, and non-clustered staging and production. This installation option sets most settings by default, but allows you to specify the installation path and database type. You must

perform a Typical installation from an electronic installation image (not a DVD).

Installing IBM Business Process Manager Standard silently

You can install the IBM Business Process Manager Standard product package in *silent* installation mode. When you install in silent mode, the user interface is not available.

Important: Only one IBM Installation Manager is required to install multiple instances of IBM Business Process Manager Standard.

Installing IBM Business Process Manager Standard silently using the command line:

You can install IBM Business Process Manager Standard using the command line. You must install silently from an electronic installation image (not a DVD).

Before you install IBM Business Process Manager Standard, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

If you are planning to install IBM Business Process Manager using DB2 Express with Red Hat Enterprise Linux 6, you must have administrative privileges (root user), must not have an existing DB2 database server on the system, and you must also ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the **ipcs -l** command.

If you receive the following warning message during the prerequisite checking, use the platform-specific steps below to increase the **ulimit** number.

```
Current system has detected a lower level of ulimit than the recommended value of recommended_value. Please increase the ul

Shutdown your installer. If you are a root user open a command prompt and issue ulimit -n recommended_value and then restar
```

The required value is calculated based on the version of WebSphere Application Server, the feature packs, and the configuration that you are installing.

1. Set the maximum number of open files using the following steps: **AIX**
   a. Open /etc/security/limits.
   b. Edit or add the **default** section and include this line:

      nofiles = *recommended_value*
   c. Save and close the file.
   d. Log off and log in again.

   Linux

   a. Open /etc/security/limits.conf.
   b. Locate the **nofile** parameter and increase the value. If a line containing the **nofile** parameter does not exist, add the following lines to the file:

      * **hard nofile** *recommended_value*

      * **soft nofile** *recommended_value*
   c. Save and close the file.
   d. Log off and log in again.

   Solaris

a. Open /etc/system and add the following line to the end of the file:

   set rlim_fd_max=8800

b. Save and close the file.

c. Log off and log in again.

2. Restart the computer.

3. Restart the installer.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Standard installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment
- Feature Pack for XML

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Standard.

To silently install IBM Business Process Manager Standard, complete the following steps:

1. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.

2. Run the following command:

   Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

   `Windows`

   ```
   extract_directory\imcl install list_of_product_IDs -acceptLicense -installationDirectory location -repositories repos
   ```

   `UNIX` `Linux`

   ```
   extract_directory/imcl install list_of_product_IDs -acceptLicense -installationDirectory location -repositories repos
   ```

   where:

   - *list_of_product_IDs* is a list of the IDs for the products and features you want to install. The syntax is *productID,feature,feature*, with multiple products separated by spaces.

Table 58. Product IDs

Product	Product ID
IBM BPM Standard	com.ibm.ws.WBPMSTD
WebSphere Application Server Network Deployment	com.ibm.websphere.ND.v70,core.feature,samples,import.productProviders (includes all required features)
Feature Pack for XML	com.ibm.websphere.XML.v10
Installation Manager	com.ibm.cic.agent,agent_core,agent_jre
DB2 for Linux 32-bit	com.ibm.ws.DB2EXP97.linuxia32
DB2 for Linux 64-bit	com.ibm.ws.DB2EXP97.linuxia64
DB2 for Windows 32-bit	com.ibm.ws.DB2EXP97.winia32
DB2 for Windows 64-bit	com.ibm.ws.DB2EXP97.winia64

- *location* is the path to the directory where you want to install the products.

- *repository* is the path to the repository where you have extracted the files, one of the following directories:

  extract_directory/repository/repos_32bit
  extract_directory/repository/repos_64bit

  For more than one repository, separate the repository locations with commas.
- *key=value* is a list of the keys and values you want to pass to the installation, separated by commas. Do not put spaces between the commas.

Table 59. Keys

Key	Description
user.select.64bit.image	If you are installing on a 64-bit operating system, add the following line exactly: user.select.64bit.image,,com.ibm.websphere.ND.v70=true The default value is false.
user.db2.admin.username	Windows only. User name with authority to access the DB2 database. The default value is bpmadmin.
user.db2.admin.password	Windows only. Password for the user name above. The default value is bpmadmin1.
user.db2.port	Port for the DB2 database. The default value is 50000.
user.db2.instance.username	Linux and UNIX only. DB2 instance user name. The default value is bpminst.
user.db2.instance.password	Linux and UNIX only. Password for the user name above. The default value is bpminst1.
user.db2.fenced.username	Linux and UNIX only. Fenced user name. The default value is bpmfenc.
user.db2.fenced.password	Linux and UNIX only. Password for the user name above. The default value is bpmfenc1.
user.db2.das.username	Linux and UNIX only. Administration server (DAS) user name. The default value is bpmadmin.
user.db2.das.password	Linux and UNIX only. Password for the user name above. The default value is bpmadmin1.

- *logName* is the name of the log file to record messages and results.

  Running this command installs the product with the default features. If you want to install specific features or make other changes, see the reference link for the command-line arguments for imcl.

Installation Manager installs the products that are listed and writes a log file to the directory that you specified.

The following example installs IBM Business Process Manager Standard with a test environment, WebSphere Application Server Network Deployment, Feature Pack for XML, and DB2 for Windows 32-bit on Windows.

```
imcl install com.ibm.ws.WBPMSTD com.ibm.websphere.ND.v70,core.feature,samples,import.productProviders.feature,import.config
```

Related reference:

⇨ Command-line arguments for imcl

⇨ IBM Business Process Manager Standard system requirements

Installing IBM Business Process Manager Standard silently using a response file:

You can install IBM Business Process Manager Standard by creating a response file and then running a command to use that response file to install the product.You must install silently from an electronic installation image (not a DVD).

Before you install IBM Business Process Manager Standard, review the system requirements for the product.

Operating system and software prerequisite levels are particularly important. Although the installation process automatically checks for prerequisite operating system patches, review the system requirements if you have not already done so. The system requirements link lists all supported operating systems and the operating system fixes and patches that you must install to have a compliant operating system. It also lists the required levels of all prerequisite software.

If you are planning to install IBM Business Process Manager using DB2 Express with Red Hat Enterprise Linux 6, you must have administrative privileges (root user), must not have an existing DB2 database server on the system, and you must also ensure that all kernel requirements are met before the DB2 Express installation begins. You can locate the current values by parsing the output of the **ipcs -l** command.

If you receive the following warning message during the prerequisite checking, use the platform-specific steps below to increase the **ulimit** number.

```
Current system has detected a lower level of ulimit than the recommended value of recommended_value. Please increase the
Shutdown your installer. If you are a root user open a command prompt and issue ulimit -n recommended_value and then res
```

The required value is calculated based on the version of WebSphere Application Server, the feature packs, and the configuration that you are installing.

1. Set the maximum number of open files using the following steps: AIX
   a. Open /etc/security/limits.
   b. Edit or add the **default** section and include this line:

      nofiles = *recommended_value*
   c. Save and close the file.
   d. Log off and log in again.

   Linux

   a. Open /etc/security/limits.conf.
   b. Locate the **nofile** parameter and increase the value. If a line containing the **nofile** parameter does not exist, add the following lines to the file:

      * hard nofile *recommended_value*

      * soft nofile *recommended_value*
   c. Save and close the file.
   d. Log off and log in again.

   Solaris

   a. Open /etc/system and add the following line to the end of the file:

      set rlim_fd_max=8800
   b. Save and close the file.
   c. Log off and log in again.
2. Restart the computer.
3. Restart the installer.

If you do not have the prerequisite base products necessary for IBM Business Process Manager Standard installation, you must install them as part of the silent installation. The required base products are:

- Installation Manager
- WebSphere Application Server Network Deployment
- Feature Pack for XML

The silent installation performs the following tasks:

- Installs Installation Manager if it is not already installed or updates it to the appropriate level if it is installed.
- Installs the required base products and IBM Business Process Manager Standard.

To silently install IBM Business Process Manager Standard, complete the following steps:

1. Create the response file that will install the required base products and IBM Business Process Manager Standard. Copy the sample response file from the following directory to create your own response file:

    extract_directory/responsefiles/BPM/template_response.xml

2. Modify the parameters as directed in the text of the response file template to create your response file.

    Make the following changes:

    - Set your repository location. Choose the appropriate 32-bit or 64-bit repository, and comment out the one that you are not using. If you are not running directly from the *extract_directory*/responsefiles/BPM/ directory, point to the location of your installation repository. The repository can be local or remote. If you are running from the DVD, copy the response file from the DVD and point it back at the repository on the DVD.

    - Set the replacement variables in the response file as follows:

        – ${INSTALL_LOCATION_IM} - The location where IBM Installation Manager is already installed or will be installed.

        – ${LOCATION_ECLIPSE_CACHE} - The location of the Eclipse cache. This location must be set only if the location has not already been defined. Comment out this line if the location has already been set.

        – ${INSTALL_LOCATION} - The location where you want to install the product.

        – ${FEATURE_LIST} - The list of features for the product. This list must be replaced with the list of features that you want installed. Follow the instructions included in the `template_response.xml` file.

    - If you are installing DB2 Express, follow the instructions in the `template_response.xml` file for providing the necessary user IDs and passwords.

3. Read and accept the license terms before installing. Adding **-acceptLicense** to the command line means that you accept all licenses.

4. Run the following command:

    Important: If you are running Windows 7, Windows Vista, or Windows Server 2008, start your command prompt by right-clicking and selecting **Run as administrator**.

    Administrator or Root user: `Windows`

    ```
    extract_directory\IM\installc.exe -acceptLicense input
    extract_directory\responsefiles\productID\template_response.xml -log preferred_log_location\silent_install.log
    ```

    `UNIX`   `Linux`

    ```
    extract_directory/IM/installc -acceptLicense input
    extract_directory/responsefiles/productID/template_response.xml -log preferred_log_location/silent_install.log
    ```

    Nonadministrator/nonroot user: `Windows`

```
extract_directory\IM\userinstc.exe -acceptLicense input
extract_directory\responsefiles\productID\template_response.xml -log preferred_log_location\silent_install.log
```

| UNIX | Linux |

```
extract_directory/IM/userinstc -acceptLicense input
extract_directory/responsefiles/productID/template_response.xml -log preferred_log_location/silent_install.log
```

Installation Manager installs any required prerequisites and IBM Business Process Manager Standard, and writes a log file to the directory you specified.

Related reference:

➡ IBM Business Process Manager Standard system requirements

Related information:

➡ Installing silently with Installation Manager

➡ Recording a response file with Installation Manager

Upgrading configurations

Upgrade a configuration of IBM Business Process Manager by using IBM Installation Manager to install the new configuration over a previously installed configuration in the same package group.

The configuration you want to upgrade must have been installed using Installation Manager. Before you perform the upgrade, shut down any servers or network deployment environments.

The supported product upgrade paths are:
- IBM Business Process Manager Express to IBM Business Process Manager Standard
- IBM Business Process Manager Express to IBM Business Process Manager Advanced
- IBM Business Process Manager Standard to IBM Business Process Manager Advanced

There are no upgrade paths to or from the IBM Business Process Manager Advanced - Process Server configuration.

Note: When you perform a configuration upgrade, any existing IBM Business Process Manager profiles are left unchanged.

The supported upgrade paths include the ability to upgrade from a previous version to V7.5.1. For example, you can upgrade your IBM Business Process Manager Express, version 7.5.0.1 to either IBM Business Process Manager Standard, V7.5.1 or IBM Business Process Manager Advanced, V7.5.1. You do not need to upgrade the edition prior to upgrading the version.

CAUTION:
Upgrading from V7.5.1 of a configuration to a previous version of an upgraded configuration is not recommended. For example, if you are using IBM Business Process Manager Express, V7.5.1, and you want to upgrade to IBM Business Process Manager Standard, you should upgrade to V7.5.1 of IBM Business Process Manager Standard, and not to a previous version. Upgrading to a previous version can cause issues with your profiles and if there are interim fixes, those interim fixes must first be removed or you encounter errors.

1. Start the Installation Manager.
2. Click **File** > **Preferences** > **Repositories** and browse to the repository that contains the new configuration package. Click **Add Repository**.
3. From the Installation Manager Welcome page, click **Install**.
4. On the Install Packages page, select the configuration that you want to install.
5. Click **Next** to continue.

6. On the Licenses page, read the license agreement. If you agree to the terms of the license agreement, click **I accept the terms in the license agreements** and click **Next**.

7. On the Install Location page, select the package group that contains the configuration which you want to upgrade.

8. On the Summary page, review your choices before installing the package. If you want to change the choices that you made on previous pages, click **Back** and make your changes. When you are satisfied with your installation choices, click **Install** to install the package. A progress indicator shows the percentage of the installation completed.

9. When the installation process is complete, a message confirms the success of the process.

   a. Optional: Click **View Log File** to open the installation log file for the current session in a new window. Close the Installation Log window to continue.

   b. Select **Profile Management Tool** to launch the Profile Management Tool when you finish, or select **None** to complete the installation.

   c. Click **Finish** to close the Installation Manager.

Configuring IBM Business Process Manager Standard

You can configure IBM Business Process Manager to form a stand-alone or network deployment environment.

The configurations (Advanced, Standard, or Express) provide different levels of business process management functionality. For example, Business Process Choreographer functionality is available only for IBM Business Process Manager Advanced. For a full description of IBM Business Process Manager Standard configurations, see IBM Business Process Manager, Version 7.5.1 configuration capabilities.

The configuring phase consists of two types of tasks: product configuration tasks and environment configuration tasks. Product configuration tasks are for setting up the product profiles and configuring the database, while environment configuration tasks are for setting up and generating the deployment environment.

One of the environment configuration tasks is creating the clusters of the deployment environment. Clusters in a deployment environment require specific tables, schemas, and user permissions based on the functional purpose of the cluster. For example, a deployment environment could include both a messaging infrastructure cluster and an application deployment target cluster. The messaging infrastructure cluster accesses messaging engine database tables and the application deployment target cluster accesses Business Process Choreographer database tables.

Figure 8 on page 119 illustrates the task flow for planning, installing, and configuring the product and the environment.
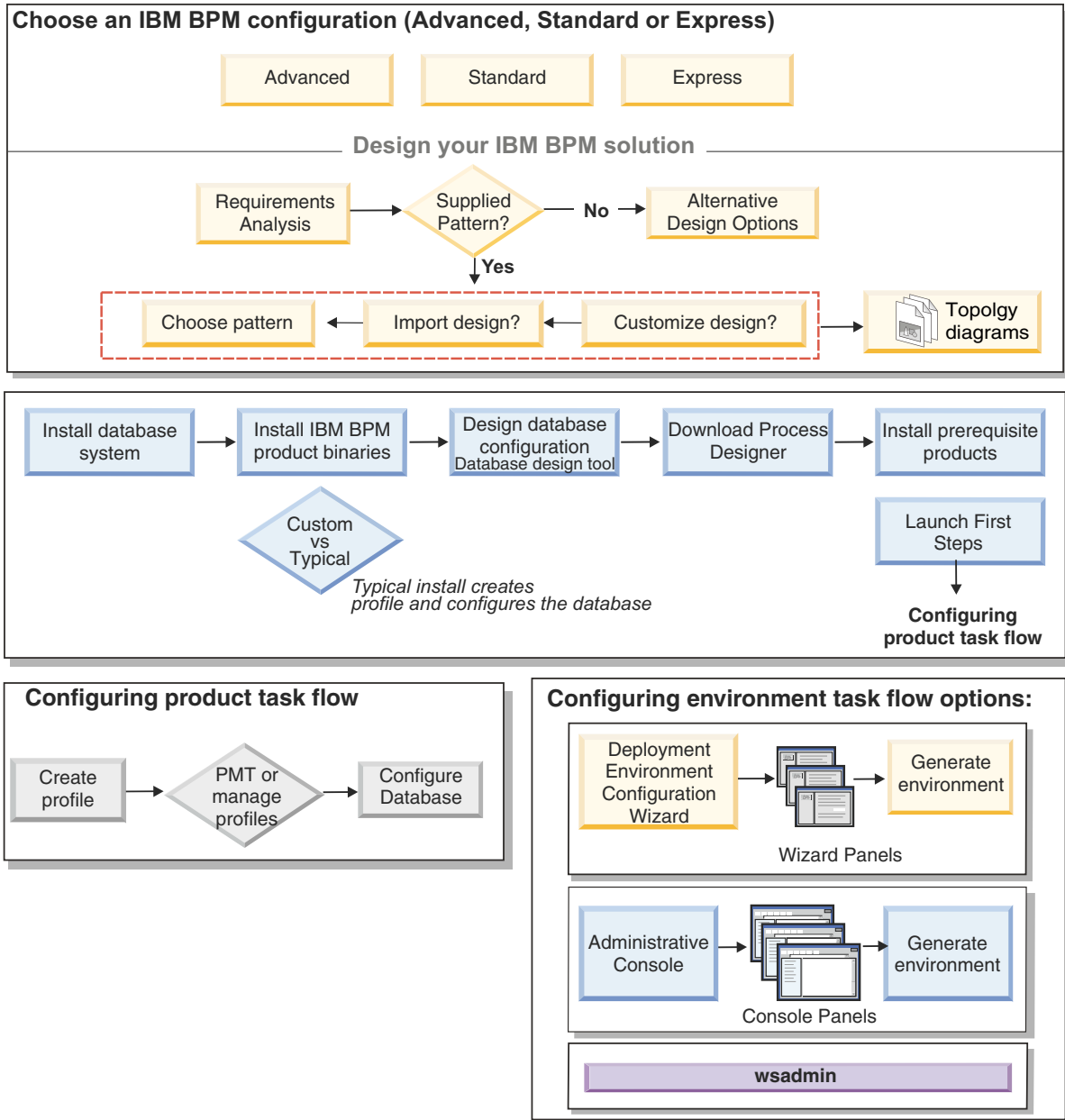
Figure 8. Task flow for planning, installing, and configuring the product and the environment

You can create the product databases before installing the product, after installing the product (but before profile creation), or after you create the profiles.

Configuring the network deployment environment using the configureNode command

To configure a typical network deployment environment, you can use the **configureNode** command instead of using the Profile Management Tool and deployment environment wizard or the **manageprofiles** and **wsadmin** commands. You can also use **configureNode** later to extend the deployment environment.

Related concepts:

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

Creating the deployment environment using the configureNode command

You can use the **configureNode** command to create a typical network deployment environment.

Note: You can use the **configurenode** properties available in the input file for common configurations. For configuring using advanced options such as advanced database properties, you must use the Deployment Environment wizard.

You must have installed the product and all the databases that are specified in the properties file.

Important: You must have already installed the database software and created the databases (but do not create tables). You must also have created all the users (especially for SQL Server and Oracle) that you specify in the properties file. You can create the databases manually or by using the following SQL scripts:

- For the Process Server database:

  install_root\dbscripts\ProcessServer*db_type*\createDatabase.sql

- For the Performance Data Warehouse database:

  install_root\dbscripts\PerformanceDW*db_type*\createDatabase.sql

- For the Common database:

  install_root\dbscripts\CommonDB*db_type*\createDatabase_CommonDB.sql

- For the Business Process Choreographer database:

  install_root\dbscripts\ProcessChoreographer*db_type*\createDatabase.sql

SQL Server considerations:

- Make sure that the username and schema exist before the configuration is done. The schema value should be the default schema for the user chosen.
- To indicate that users will connect to the databases using Windows credentials, select the individual data source, click **Edit**, and select **Apply Windows authentication**.

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

Restriction: Run **configureNode** commands sequentially; do not try to run the command on multiple machines at the same time.

The **configureNode** command creates the following network deployment topology:

- A single cell environment
- The environment contains the number of cluster members that you specify in the properties file
- The environment uses the Remote Messaging and Remote Support topology pattern, which includes the following three clusters:
  - Messaging infrastructure cluster
  - Supporting infrastructure cluster
  - Application deployment target cluster

The command performs the following tasks:

- Creates the deployment manager node based on the values in the deployment manager properties file and starts the deployment manager.
- Creates the deployment environment definition.

- Creates a custom node based on the values in the custom node properties file.
- Federates the node and adds the node to the deployment environment.
- Generates the deployment environment.
- Creates the database tables. (You must already have created the databases. See "Before you begin" for instructions.)
- Runs the bootstrap utility to load the Process Server database with system information.

The default port is used to create the deployment manager. If a deployment manager is already running on the machine when you run **configureNode**, creating the deployment environment fails. If a deployment manager has already been created but is not running, you must change the port of one of the deployment managers or you might get a port conflict error when you start the other deployment manager.

To create the deployment environment for the first time, complete the following steps:

1. On the machine where you want to create the deployment environment, locate the sample properties files: *install_root*/util/ndUtils/samples/.
2. Copy the sample files that are most similar to the properties files that you need. For example, for a deployment manager for IBM BPM Standard for Process Center, choose the sample_std_pc_dmgr.properties file.For example, for a deployment manager for IBM BPM Advanced for Process Center, choose the sample_adv_pc_dmgr.properties file. Modify the new files to reflect your environment.

   For each database that you want to create, copy the database parameters section and specify the database name, user name, password, and schema name. The Process Server database and Performance Data Warehouse database are required, and they must have different names.

   For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.

   Important: If you are using an Oracle database, you must include the database user name and password for all databases, including the optional ones.
3. Run the **configureNode** command, passing it the names of the two files you created. The custom node file is optional, but if you do not include it, only the deployment manager is created. For example:

   Windows   *install_root*\util\ndUtils**configureNode.bat -dmgr_response** *my_dmgr_response_file.properties* **-response** *node01_response_file.properties*

   Linux   UNIX   *install_root*/util/ndUtils/**configureNode.sh -dmgr_response** *my_dmgr_response_file.properties* **-response** *node01_response_file.properties*

You now have a deployment manager, a custom node, and a deployment environment that use the Remote Messaging and Remote Support topology pattern.

Messages are recorded in the file *install_root*/logs/config/configureNode.log.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

➡ Virtual hosts

Related tasks:

"Encrypting passwords" on page 657
When you install IBM Business Process Manager product components, passwords in the installed files are generally encrypted by default. If you need to change these passwords, you can use the IBM BPM

EncryptPassword utility to encrypt the new passwords.

�['] Configuring virtual hosts

Related reference:

"**configureNode** examples" on page 123
The examples in this section show how to create network deployment topologies using the **configureNode** command-line utility.

Extending the deployment environment using the configureNode command

After you have created a deployment environment using the **configureNode** command, you can run the command at a later time to extend the deployment environment.

You need the following prerequisites before running the **configureNode** command:
- You must have installed the product on the machine where you want to extend the deployment environment.
- The deployment manager must be running.
- You must have created the deployment environment by running the **configureNode** command.

    Restriction: Run **configureNode** commands sequentially; do not try to run the command on multiple machines at the same time.

To extend the deployment environment, complete the following steps:
1. On the machine where you want to extend the deployment environment, locate the sample custom node files: *install_root*/util/ndUtils/samples/
2. Copy the sample file that is most similar to the properties file that you need. For example, to extend the deployment environment for IBM BPM Standard for Process Center, choose the sample_std_pc_node.properties file.For example, to extend the deployment environment for IBM BPM Advanced for Process Center, choose the sample_adv_pc_node.properties file. Modify the new file to reflect your environment, including the details of the deployment manager. For more information about the available properties, read the comments in the sample files, or see the **configureNode** command reference and the examples.
3. Run the **configureNode** command, passing it the name of the sample file. For example:

    `Windows` ***install_root*\util\ndUtils\configureNode.bat -response** *node01_response_file.properties*

    `Linux`   `UNIX`   ***install_root*/util/ndUtils/configureNode.sh -response** *node01_response_file.properties*

You have extended the deployment environment.

Messages are recorded in the file *install_root*/logs/config/configureNode.log.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

➚ Virtual hosts

Related tasks:

"Encrypting passwords" on page 657
When you install IBM Business Process Manager product components, passwords in the installed files are generally encrypted by default. If you need to change these passwords, you can use the IBM BPM

EncryptPassword utility to encrypt the new passwords.

⬐ Configuring virtual hosts

Related reference:

"**configureNode** examples"
The examples in this section show how to create network deployment topologies using the **configureNode** command-line utility.

configureNode examples

The examples in this section show how to create network deployment topologies using the **configureNode** command-line utility.

Related tasks:

"Creating the deployment environment using the configureNode command" on page 120
You can use the **configureNode** command to create a typical network deployment environment.

"Extending the deployment environment using the configureNode command" on page 122
After you have created a deployment environment using the **configureNode** command, you can run the command at a later time to extend the deployment environment.

Example: Single machine hosting the deployment manager and one node: Standard, PC, DB2:

Use the **configureNode** command-line utility to create a Process Center topology for IBM Business Process Manager Standard with a deployment manager and a single node on the same machine, using a DB2 database.

This example assumes that the machine is called M01 and you want to create the following configuration:

Machine	Node	Messaging cluster	Support cluster	Application target cluster
M01	Dmgr	-	-	-
M01	PCNode01	1	1	1

1. Copy the following sample properties files:

    install_root/util/ndUtils/std/samples/sample_std_pc_dmgr.properties

    install_root/util/ndUtils/std/samples/sample_std_pc_node.properties

2. Rename the new files, for example to m01_std_pc_dmgr.properties and m01_std_pc_node.properties. Update the files with the required values.

    Example of deployment manager file:

```
# filename: m01_std_pc_dmgr.properties

server.type = PCS       # Process Center (PCS) or Process Server (PS)

# Dmgr settings
hostName=M01.ibm.com
adminUserName=admin
adminPassword=admin

nodeName=Dmgr

# Database settings
dbType=DB2
dbServerName=M01.ibm.com
dbPortNumber=50000

# DB for Process Server
prcsvr.dbName = BPMDB
prcsvr.dbUserName = bpmadmin
```

```
prcsvr.dbPassword = s0m3passw0rd

# DB for Performance Data Warehouse
perfDW.dbName = PDWDB
perfDW.dbUserName = bpmadmin
perfDW.dbPassword = s0m3passw0rd
```

Example of custom node file:

```
# filename: m01_std_pc_node.properties

nodeName=Node01

# Basic Install environment information
server.type = PCS        # Process Center (PCS) or Process Server (PS)

#Install clusters on this node
install.cluster.support = 1
install.cluster.AppTarget = 1
install.cluster.message = 1

# Database settings
dbType=DB2

# Deployment Manager Properties
dmgr.hostName=M01.ibm.com
dmgr.adminUserName=admin
dmgr.adminPassword=admin
```

3. Run the **configureNode** command on the M01 machine:

> Windows *install_root*\util\ndUtils\configureNode.bat -dmgr_response
> **m01_dmgr_response_file.properties -response m01_response_file.properties**

> Linux   UNIX *install_root*/util/ndUtils/configureNode.sh -dmgr_response
> **m01_dmgr_response_file.properties -response m01_response_file.properties**

Example: Single machine hosting the deployment manager and one node: Standard, PC, Oracle:

Use the **configureNode** command-line utility to create a Process Center topology for IBM Business Process Manager Standard with a deployment manager and a single node on the same machine, using an Oracle database.

Note: You can use the **configurenode** properties available in the input file for common configurations. The **configurenode** command can only be used for a database on a single machine. For configuring using advanced options such as advanced database properties or for specifying information for multiple databases, you must use the Deployment Environment wizard.

This example assumes that the machine is called M01 and you want to create the following configuration:

Machine	Node	Messaging cluster	Support cluster	Application target cluster
M01	Dmgr	-	-	-
M01	PCNode01	1	1	1

1. Copy the following sample properties files:

   install_root/util/ndUtils/std/samples/sample_std_pc_dmgr.properties

   install_root/util/ndUtils/std/samples/sample_std_pc_node.properties

2. Rename the new files, for example to m01_std_pc_dmgr.properties and m01_std_pc_node.properties. Update the files with the required values. For Oracle database, you must include the database user name and password for all databases, including the ones that would otherwise be optional. The database user name must be different for each database.

Example of deployment manager file:

```
# filename: m01_std_pc_dmgr.properties

server.type = PCS        # Process Center (PCS) or Process Server (PS)

# Dmgr settings
hostName=M01.ibm.com
adminUserName=admin
adminPassword=admin

nodeName=Dmgr

# Database settings
dbType=Oracle
dbServerName=M01.ibm.com
dbPortNumber=1521

# DB for Process Server
prcsvr.dbName = orcl
prcsvr.dbUserName = bpmdba
prcsvr.dbPassword = s0m3passw0rd

# DB for Process Server ME
prcsvrME.dbName = orcl
prcsvrME.dbUserName = bpmmedba
prcsvrME.dbPassword = s0m3passw0rd

# DB for Performance Data Warehouse
perfDW.dbName = orcl
perfDW.dbUserName = pdwdba
perfDW.dbPassword = s0m3passw0rd

# DB for PerformanceDW ME
perfDWME.dbName = orcl
perfDWME.dbUserName = pdwmedba
perfDWME.dbPassword = s0m3passw0rd
```

Example of custom node file:

```
# filename: m01_std_pc_node.properties

nodeName=Node01

# Basic Install environment information
server.type = PCS        # Process Center (PCS) or Process Server (PS)

#Install clusters on this node
install.cluster.support = 1
install.cluster.AppTarget = 1
install.cluster.message = 1

# Database settings
dbType=Oracle

# Deployment Manager Properties
dmgr.hostName=M01.ibm.com
dmgr.adminUserName=admin
dmgr.adminPassword=admin
```

3. Run the **configureNode** command on the M01 machine:

> `Windows` *install_root*\util\ndUtils\configureNode.bat -dmgr_response
> **m01_dmgr_response_file.properties -response m01_response_file.properties**

> `Linux`   `UNIX`   *install_root*/util/ndUtils/configureNode.sh -dmgr_response
> **m01_dmgr_response_file.properties -response m01_response_file.properties**

Example: Single machine hosting the deployment manager and one node: Standard, PC, SQL Server:

Use the **configureNode** command-line utility to create a Process Center topology for IBM Business Process Manager Standard with a deployment manager and a single node on the same machine, using a SQL Server database.

This example assumes that the machine is called M01 and you want to create the following configuration:

Machine	Node	Messaging cluster	Support cluster	Application target cluster
M01	Dmgr	-	-	-
M01	PCNode01	1	1	1

1. Copy the following sample properties files:

   install_root/util/ndUtils/std/samples/sample_std_pc_dmgr.properties

   install_root/util/ndUtils/std/samples/sample_std_pc_node.properties

2. Rename the new files, for example to m01_std_pc_dmgr.properties and m01_std_pc_node.properties. Update the files with the required values.

   Example of deployment manager file:

   ```
   # filename: m01_std_pc_dmgr.properties

   install.type = Standard # Optional
   server.type = PCS        # Process Center (PCS) or Process Server (PS)

   # Dmgr setting
   hostName=M01.ibm.com
   adminUserName=admin
   adminPassword=admin

   nodeName=Dmgr

   # Database settings
   dbType=SQLServer
   dbServerName=M01.ibm.com
   dbPortNumber=1433
   # When dbWinAuth=true, Windows Authentication is used
   dbWinAuth=false

   # DB for Process Server
   prcsvr.dbName = BPMDB
   prcsvr.dbUserName = bpmadmin
   prcsvr.dbPassword = s0m3passw0rd

   # DB for Performance Data Warehouse
   perfDW.dbName = PDWDB
   perfDW.dbUserName = bpmadmin
   perfDW.dbPassword = s0m3passw0rd
   ```

   Example of custom node file:

   ```
   # filename: m01_std_pc_node.properties

   nodeName=Node01

   # Basic Install environment information
   server.type = PCS        # Process Center (PCS) or Process Server (PS)

   #Install clusters on this node
   install.cluster.support = 1
   install.cluster.AppTarget = 1
   install.cluster.message = 1
   ```

```
# Database settings
dbType=SQLServer

# Deployment Manager Properties
dmgr.hostName=M01.ibm.com
dmgr.adminUserName=admin
dmgr.adminPassword=admin
```

3. Run the **configureNode** command on the M01 machine:

   Windows *install_root*\util\ndUtils\configureNode.bat -dmgr_response
   m01_dmgr_response_file.properties -response m01_response_file.properties

   Linux   UNIX   *install_root*/util/ndUtils/configureNode.sh -dmgr_response
   m01_dmgr_response_file.properties -response m01_response_file.properties

Example: Single machine hosting the deployment manager and one node: Standard, PS, DB2:

After creating a Process Center topology, you can also use the **configureNode** command-line utility to create a Process Server topology for IBM Business Process Manager Standard with a deployment manager and a single node on the same machine, using a DB2 database.

This example assumes that the machine is called M02 and you want to create the following configuration:

Machine	Node	Messaging cluster	Support cluster	Application target cluster
M02	Dmgr	-	-	-
M02	PSNode01	1	1	1

1. Copy the following sample properties files:

   install_root/util/ndUtils/std/samples/sample_std_ps_dmgr.properties

   install_root/util/ndUtils/std/samples/sample_std_ps_node.properties

2. Rename the new files, for example to m02_std_ps_dmgr.properties and m02_std_ps_node.properties. Update the files with the required values.

   Example of deployment manager file:

```
# filename: m02_std_ps_dmgr.properties

server.type = PS        # Process Center (PCS) or Process Server (PS)

# Dmgr setting
hostName=M02.ibm.com
adminUserName=admin
adminPassword=admin

nodeName=Dmgr

# Environment settings
environment.type = Test
processCenterURL= http://M01:9080 #(optional, if not set, the environment is offline PS)
repository.user = admin #(optional, if not set, the environment is offline PS)
repository.credential = admin #(optional, if not set, the environment is offline PS)

# Database settings
dbType=DB2
dbServerName=M01
dbPortNumber=50000

# DB for Process Server
prcsvr.dbName = BPMPSDB
prcsvr.dbUserName = bpmadmin
prcsvr.dbPassword = s0m3passw0rd
```

```
# DB for Performance Data Warehouse
perfDW.dbName = PDWPSDB
perfDW.dbUserName = bpmadmin
perfDW.dbPassword = s0m3passw0rd
```

Example of custom node file:

```
# filename: m02_std_ps_node.properties

nodeName=PSNode01

# Basic Install environment information
server.type = PS        # Process Center (PCS) or Process Server (PS)
install.type = Standard # Optional

#Install clusters on this node
install.cluster.support = 1
install.cluster.AppTarget = 1
install.cluster.message = 1

# Database setting
dbType=DB2

# Deployment Manager Properties
dmgr.hostName=M02.ibm.com
dmgr.adminUserName=admin
dmgr.adminPassword=admin
```

3. Run the **configureNode** command on the M02 machine:

> `Windows` *install_root*\util\ndUtils\configureNode.bat -dmgr_response
> **m01_dmgr_response_file.properties -response m01_response_file.properties**

> `Linux` `UNIX` *install_root*/util/ndUtils/configureNode.sh -dmgr_response
> **m01_dmgr_response_file.properties -response m01_response_file.properties**

Example: Multiple machines hosting the deployment manager and three nodes: Advanced, PS, DB2:

Use the **configureNode** command-line utility to create a Process Server topology for IBM Business Process Manager Advanced with a deployment manager and three nodes on four different machines, using a DB2 database.

Note: You can use the **configurenode** properties available in the input file for common configurations. For configuring using advanced options such as advanced database properties, you must use the Deployment Environment wizard.

This example assumes that you have four machines and you want to create the following configuration:

Machine	Node	Messaging cluster	Support cluster	Application target cluster	Web cluster
M03	Dmgr	-	-	-	
M04	Node01	1	0	1	0
M05	Node02	0	1	0	1
M06	Node03	1	1	1	1

1. Copy the following sample properties files:

   > *install_root*/util/ndUtils/std/samples/sample_adv_ps_dmgr.properties

   > *install_root*/util/ndUtils/std/samples/sample_adv_ps_node.properties

2. Create a new deployment manager file, for example adv_dmgr.properties and three custom node files, for example node01.properties, node02.properties, and node03.properties. Update the files with the required values.

Important: If you are using an Oracle database instead of a DB2 database, you must include the database user name and password for all databases, including the optional ones. The database user name must be different for each database.

Example of deployment manager file:

```
# filename: adv_dmgr.properties

server.type = PS          # Process Center (PCS) or Process Server (PS)

# Dmgr setting
hostName=M03.ibm.com
adminUserName=admin
adminPassword=admin

nodeName=Dmgr

# Environment settings
environment.type = Test
processCenterURL= http://M01:9080 #(optional, if not set, the environment is offline PS)
repository.user = admin #(optional, if not set, the environment is offline PS)
repository.credential = admin #(optional, if not set, the environment is offline PS)

# Database settings
dbType=DB2
dbServerName=M01
dbPortNumber=50000

# DB for Process Server
prcsvr.dbName = BPMDB 2
prcsvr.dbUserName = bpmadmin
prcsvr.dbPassword = s0m3passw0rd

# DB for Performance Data Warehouse
perfDW.dbName = PDWDB 2
perfDW.dbUserName = bpmadmin
perfDW.dbPassword = s0m3passw0rd

# DB for Common DB
cmn.dbName=CMNDB2
cmn.dbUserName=bpmadmin
cmn.dbPassword=s0m3passw0rd

# DB for Business Process Choreographer
bpc.dbName=BPEDB2
bpc.dbUserName=s0m3passw0rd
bpc.dbPassword=bpmadmin
```

Examples of custom node files:

```
# filename: node01.properties

nodeName=Node01

# Basic Install environment information
server.type = PS          # Process Center (PCS) or Process Server (PS)
install.type = Advanced # Optional

#Install clusters on this node
install.cluster.support = 1
install.cluster.AppTarget = 1
install.cluster.message = 0
install.cluster.web = 0

# Database settings
dbType=DB2
```

```
# Deployment Manager Properties
dmgr.hostName=M03.ibm.com
dmgr.adminUserName=admin
dmgr.adminPassword=admin

# filename: node02.properties

nodeName=Node02

# Basic Install environment information
server.type = PS        # Process Center (PCS) or Process Server (PS)
install.type = Advanced # Optional

#Install clusters on this node
install.cluster.support = 0
install.cluster.AppTarget = 0
install.cluster.message = 1
install.cluster.web = 1

# Database settings
dbType=DB2

# Deployment Manager Properties
dmgr.hostName=M03.ibm.com
dmgr.adminUserName=admin
dmgr.adminPassword=admin

# filename: node03.properties

nodeName=Node03

# Basic Install environment information
server.type = PS        # Process Center (PCS) or Process Server (PS)
install.type = Advanced # Optional

#Install clusters on this node
install.cluster.support = 1
install.cluster.AppTarget = 1
install.cluster.message = 1
install.cluster.web = 1

# Database settings
dbType=DB2

# Deployment Manager Properties
dmgr.hostName=M03.ibm.com
dmgr.adminUserName=admin
dmgr.adminPassword=admin
```

3. Run the **configureNode** command on the four machines.

   a. Log on to machine M03 and run the following command:

      Windows   *install_root***util\ndUtils\configureNode.bat -dmgr_response adv_dmgr.properties**

      Linux   UNIX   *install_root*/**util/ndUtils/configureNode.sh -dmgr_response adv_dmgr.properties**

   b. Log on to machine M04 and run the following command:

      Windows   *install_root***util\ndUtils\configureNode.bat -response node01.properties**

      Linux   UNIX   *install_root*/**util/ndUtils/configureNode.sh -response node01.properties**

   c. Log on to machine M05 and run the following command:

      Windows   *install_root***util\ndUtils\configureNode.bat -response node02.properties**

      Linux   UNIX   *install_root*/**util/ndUtils/configureNode.sh -response node02.properties**

   d. Log on to machine M06 and run the following command:

| Windows | *install_root*\util\ndUtils\configureNode.bat -response node03.properties |

| Linux | UNIX | *install_root*/util/ndUtils/configureNode.sh -response node03.properties |

Creating and augmenting profiles

After you install the product, if you did not create profiles automatically using the Typical installation or the configureNode command, you must create one or more profiles to define the runtime environment.

Prerequisites and considerations for profiles

Before you create or augment profiles for IBM Business Process Manager, review the prerequisites and other considerations.

Prerequisites for creating or augmenting profiles:

Before creating or augmenting a profile, you must ensure that a series of prerequisites have been met.

- You must have an existing installation of IBM Business Process Manager. If you do not, see Installing and configuring IBM Business Process Manager for installation procedures.
- If you are not the user ID who installed the product, you must have write permission to selected directories within the IBM Business Process Manager installation. See "Granting write permission of files and directories to nonroot users for profile creation" on page 132 for instructions on how to obtain these permissions. You must create your profiles in a directory other than *install_root*/profiles.
- Windows To install or run the Profile Management Tool on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges. Whether you are an administrative user or a non-administrative user, right-click the pmt.bat file and select **Run as administrator**. Alternatively, use the **runas** command at the command line. For example, the following command can be run from the install_root\bin\ProfileManagement directory:

  ```
  runas /user:MyAdminName /env pmt.bat
  ```

  Non-administrative users are prompted for the administrator password.

- Windows To install or run the **manageprofiles** command on Windows 7, Windows Vista, or Windows Server 2008, you must elevate your Microsoft Windows user account privileges using the **runas** command. Remember to put double quotation marks around the **manageprofiles** command and all parameters. For example, the following command can be run from the install_root\bin directory:

  ```
  runas /env /user:MyAdminName "manageprofiles -create -templatePath install_root/profileTemplates/BPM/default.procctr"
  ```

  Non-administrative users are prompted for the administrator password.

- You must know the type of profile you want to create or augment. For more information about profiles, see Profiles.
- You must follow the correct procedure to create or augment the profile:
  - If you want to create profiles for a IBM Business Process Manager stand-alone configuration, see Creating stand-alone profiles using the Profile Management Tool.

    The documentation includes information about creating process center profiles and process server profiles.
  - If you want to create profiles for a IBM Business Process Manager network deployment configuration, see Creating deployment manager profiles using the Profile Management Tool.
  - If you want to augment an existing profile into a IBM Business Process Manager profile, see one of the following topics:

    Important: A profile that you plan to augment cannot define a managed node that is already federated.

    - To augment stand-alone server profiles, see Augmenting Process Server stand-alone server profiles or Augmenting Process Center stand-alone server profiles

- To augment profiles for a network deployment environment configuration, see Augmenting profiles in a network deployment configuration or Augmenting profiles in a network deployment configuration

- You cannot use the Profile Management Tool to create or augment profiles on IBM Business Process Manager installations on 64-bit architectures except on the Linux on zSeries platform. To create or augment profiles on other 64-bit architectures, you can use the `manageprofiles` command-line utility.

  Note: You can use the Profile Management Tool on these architectures if you use a IBM Business Process Manager 32-bit installation.

- You must shut down any servers associated with a profile you plan to augment.
- You must review Naming considerations for profiles, nodes, servers, hosts, and cells for information about reserved terms and issues that you must consider when naming your profile, node, host, server (if applicable), or cell (if applicable).
- You must have enough disk and temporary space to create or augment the new profile. For information about space requirements, see the detailed system requirements web page, and then select the link to your version of IBM Business Process Manager.
  - For Advanced configuration: http://www.ibm.com/software/integration/business-process-manager/advanced/sysreqs/
  - For Standard configuration: http://www.ibm.com/software/integration/business-process-manager/standard/sysreqs/
  - For Express configuration: http://www.ibm.com/software/integration/business-process-manager/express/sysreqs/

Related concepts:

"Database prerequisites for creating or augmenting profiles" on page 135
Before creating or augmenting a profile, you must ensure that a series of database prerequisites have been met.

Related tasks:

"Creating Process Server deployment manager and custom profiles using manageprofiles" on page 408
Use the `manageprofiles` command-line utility to create Process Server deployment manager and custom profiles for a network deployment configuration.

"Creating Process Server stand-alone profiles using manageprofiles" on page 240
Use the `manageprofiles` command-line utility to create Process Server stand-alone profiles for IBM Business Process Manager Standard .

"Creating Process Center stand-alone profiles using manageprofiles" on page 191
Use `manageprofiles` command-line utility to create Process Center stand-alone profiles for IBM Business Process Manager Standard .

Granting write permission of files and directories to nonroot users for profile creation:

The product installer (who can be a root/Administrator or nonroot user) can grant write permission to the appropriate IBM Business Process Manager files and directories to nonroot users. The nonroot users can then create profiles. Alternatively, the product installer can create a group for users who are authorized to create profiles or give individual users the authority to create profiles.

The following example task shows how to create a group that is authorized to create profiles.

Throughout this text, the terms "installer" and "product installer" refer to the user ID that installed IBM Business Process Manager.

Restriction: IBM Business Process Manager does not support changing ownership of existing profiles from the product installer to nonroot users. Thus, profile augmentation by nonroot users of profiles owned by another user is not supported.

Nonroot users create their own profiles so that they can manage their own environments. Typically, they manage environments for development purposes.

Nonroot users must store their profiles in their private directory structure, not in the *install_root*/**profiles** directory of the product.

Restriction: An ease-of-use limitation exists for nonroot users who create profiles. Mechanisms within the Profile Management Tool that suggest unique names and port values are disabled for nonroot users. The nonroot user must change the default field values in the Profile Management Tool for the profile name, node name, cell name, and port assignments. The product installer can assign nonroot users a range of values for each of the fields, and assign responsibility to the nonroot users for adhering to their assigned value ranges and for maintaining the integrity of their own definitions.

If you already created at least one profile, then certain directories and files were created. Because these directories and files were created, skip the steps in this topic that create these directories and files. If no profile was previously created, then you must complete the steps to create the required directories and files. In most cases, a profile has been created previously.

Steps the product installer must perform to grant appropriate permissions

The installer can perform the following steps to create the `profilers` group and give the group appropriate permissions to create a profile.

1. Log on to the IBM Business Process Manager system as the product installer. (The product installer can be a root/Administrator or nonroot user.)
2. Using operating system commands, perform the following steps:
   - Create a group named profilers, which will contain all users who can create profiles.
   - Create a user named user1, who can create profiles.
   - Add users product_installer and user1 to the profilers group.
3. Linux   UNIX   Log off and log back on as the installer to pick up the new group.
4. Create the following directories as the installer if no profile exists:
   - Linux   UNIX   Create the *install_root*/**logs/manageprofiles** directory:

     `mkdir install_root/logs/manageprofiles`

      Windows Create the *install_root***logs\manageprofiles** directory by following instructions in the Windows documentation. For this example procedure, the directory is:

     `install_root\logs\manageprofiles`

   - Linux   UNIX   Create the *install_root*/**properties/fsdb** directory:

     `mkdir install_root/properties/fsdb`

      Windows Create the *install_root***properties\fsdb** directory by following instructions in the Windows documentation. For this example procedure, the directory is:

     `install_root\properties\fsdb`

5. As the installer, follow directions for your operating system to create the profileRegistry.xml file if no profile exists. For this example, the file paths are:

    Linux   UNIX

   `install_root/properties/profileRegistry.xml`

    Windows

   `install_root\properties\profileRegistry.xml`

   Follow instructions for your operating system to add the following information to the profileRegistry.xml file. The file must be encoded as UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
<profiles/>
```

6. As the product installer, use operating system tools to change directory and file permissions.

   `Linux` `UNIX` The following example assumes that the variable **$WASHOME** is the IBM Business Process Manager root installation directory .

```
export WASHOME=/opt/IBM/WebSphere/
echo $WASHOME
echo "Performing chggrp/chmod per WAS directions..."
chgrp profilers $WASHOME/logs/manageprofiles
chmod g+wr  $WASHOME/logs/manageprofiles
chgrp profilers $WASHOME/properties
chmod g+wr  $WASHOME/properties
chgrp profilers $WASHOME/properties/fsdb
chmod g+wr  $WASHOME/properties/fsdb
chgrp profilers $WASHOME/properties/profileRegistry.xml
chmod g+wr  $WASHOME/properties/profileRegistry.xml
chgrp -R profilers $WASHOME/profileTemplates
```

   `Linux` Issue the following additional commands:

```
chgrp profilers $WASHOME/properties/Profiles.menu
chmod  g+wr $WASHOME/properties/Profiles.menu
```

   `Windows` The following example assumes that the variable **$WASHOME** is the IBM Business Process Manager root installation directory . Follow instructions in the Windows documentation to give the profilers group read and write permission to the following directories and their files:

```
@WASHOME\logs\manageprofiles
@WASHOME\properties
@WASHOME\properties\fsdb
@WASHOME\properties\profileRegistry.xml
```

   You might have to change the permissions on additional files if the nonroot user encounters permission errors. For example, if the product installer authorizes a nonroot user to delete a profile, then the product installer might have to delete the following file:

   `Linux` `UNIX` *install_root*/**properties/profileRegistry.xml_LOCK**

   `Windows` *install_root***properties\profileRegistry.xml_LOCK**

   Give write access to the nonroot user for the file to authorize the user to delete the file. If the nonroot user still cannot delete the profile, then the product installer can delete the profile.

Result

The installer created the **profilers** group and gave the group proper permissions to certain directories and files to create profiles. These directories and files are the only ones in the installation root of IBM Business Process Manager to which a nonroot user needs to write to create profiles.

What to do next

The nonroot user that belongs to the **profilers** group can create profiles in a directory that the nonroot user owns and to which the nonroot user has write permission. However, the nonroot user cannot create profiles in the installation root directory of the product.

A nonroot user ID can manage multiple profiles. The same nonroot user ID can manage an entire profile, whether it is the deployment manager profile, a profile that contains the servers and the node agent, or a custom profile. A different user ID can be used for each profile in a cell, whether global security or administrative security is enabled or disabled. The user IDs can be a mix of root and nonroot user IDs. For example, the root user might manage the deployment manager profile, while a nonroot user might manage a profile that contains servers and the node agent, or vice versa. However, typically, a root user or a nonroot user can manage all profiles in a cell.

The nonroot user can use the same tasks to manage a profile that the root user uses.

Database prerequisites for creating or augmenting profiles:

Before creating or augmenting a profile, you must ensure that a series of database prerequisites have been met.

The following prerequisites relate to product databases:
- Database administrator (DBA) privileges are required for the database configuration panels that are part of creating a deployment manager profile.

  If the user ID does not have DBA privileges, use this workaround:
  1. Install the product without creating a profile.
  2. Use the Profile Management Tool to create the deployment manager and the custom profiles using the Advanced path for all. Do not use the Typical path. Do not create database tables as part of the profile creation process.
  3. Federate the custom profiles to the deployment manager.
  4. Using the administrative console, create the required deployment environment. See Creating a deployment environment using a pattern for more information.

- Linux UNIX If you plan to use DB2 Universal Database™:

  You must run the **db2profile** script to set the required DB2 environment that is used to invoke the DB2 commands, which are used during profile creation. Add the **db2profile** script to the /etc/profile directory:

  ```
  vi /etc/profile and add below lines:

  export PATH=/opt/IBM/db2/V9.7/bin:$PATH
  . /home/db2inst1/sqllib/db2profile
  ```

  After adding the db2profile script to the /etc/profile directory, you must run the **db2profile** script to set the DB2 environment.

  You must add the user ID that will be used during profile creation to the DB2 administrative groups. For example, if you log in as the root user and are creating the database using db2inst1 as the user ID, add the root to the /etc/group administrative groups:

  ```
  vi /etc/group and update below lines:

  dasadm:|;101:dasusr1,db2inst1,root
  db2iadm:|:102;root
  db2fadm:|:103;db2fenc1,root
  ```

  Typical profile creation Exceptions:

  When the **db2profile** script is not run:

  ```
  /opt/HJJ/wps4013/util/dbUtils/profileHelpers/commonDBUtility.ant:841: Execute failed:
  java.io.IOException: Cannot run program "db2" (in directory "/opt/HJJ/
  wps4013/profiles/Dmgr01/dbscripts/CommonDB/DB2/WPSDB1")
  ```

  When the DB2 database manager is not running:

  ```
  SQL1032N No start database manager command was issued. SQLSTATE=57019
  ```

  When the user who installed IBM Business Process Manager and is creating the profile is not added to the DB2 administrative groups:

  ```
  SQL1092N "ROOT" does not have the authority to perform the requested command.

  When DB2 database manager is down or not running...

  SQL1032N  No start database manager command was issued.  SQLSTATE=57019
  ```

Related concepts:

"Prerequisites for creating or augmenting profiles" on page 131
Before creating or augmenting a profile, you must ensure that a series of prerequisites have been met.

Starting the Profile Management Tool:

Before you start the Profile Management Tool, be aware of the restrictions and ensure that certain prerequisites are met. You can start the Profile Management Tool in several ways, depending on the platform on which it is running.

Restrictions:

- You cannot use the Profile Management Tool to create or augment profiles on IBM Business Process Manager installations on 64-bit architectures except on the Linux on zSeries platform. To create profiles on other 64-bit architectures, you can use the **manageprofiles** command-line utility. For information about using the **manageprofiles** command-line utility, see "Creating profiles using the manageprofiles command-line utility" on page 187. You can also use the Profile Management Tool on IBM Business Process Manager 32-bit installations on these architectures.

- `Vista` `Windows 7` **Restriction for nonadministrative users with multiple instances:** If you install multiple instances of IBM Business Process Manager as the root user and give a nonadministrative user access to only a subset of those instances, the Profile Management Tool does not function correctly for the nonadministrative user. In addition, a `com.ibm.wsspi.profile.WSProfileException` or `Access is denied` message occurs in the `install_root`\bin\ProfileManagement\pmt.bat file. By default, nonadministrative users do not have access to the `Program Files` directory, which is the default installation location for the product. To resolve this issue, nonadministrative users either install the product by themselves or be given permission to access the other product instances.

`Linux` `UNIX` `Windows` The language of the Profile Management Tool is determined by the default language on the system. If the default language is not one of the supported languages, then English is used. You can override the default language by starting the Profile Management Tool from the command line and using the **java user.language** setting to replace the default language. Run the following command:

- `Linux` `UNIX` *install_root*/**java/bin/java -Duser.language=***locale install_root*/**bin/ProfileManagement/startup.jar**

- `Windows` *install_root***java\bin\java -Duser.language=***locale install_root***bin\ProfileManagement\startup.jar**

For example, to start the Profile Management Tool in the German language on a Linux system, type the following command:

```
install_root/java/bin/java -Duser.language=de install_root/  \
bin/ProfileManagement/startup.jar
```

Starting the tool on all platforms

Start the tool on any platform from the First steps console. See Starting the First steps console for more information.

Starting the tool on Linux and UNIX platforms

`Linux` `UNIX` You can start the tool on Linux and UNIX platforms by running the command *install_root*/**bin/ProfileManagement/pmt.sh**

`Linux` On Linux platforms only, you can also use operating system menus to start the Profile Management Tool. For example, click *Linux_operating_system_menus_to_access_programs* > **IBM WebSphere** > *your_product* > **Profile Management Tool**.

Starting the tool on Windows platforms

`Windows` You can use the following methods to start the tool on Windows platforms:

- Run the command *install_root***bin\ProfileManagement\pmt.bat**

Creating stand-alone profiles after a custom installation

After you perform a Custom installation, you can use the Profile Management Tool or the `manageprofiles` command-line utility to create a stand-alone server profile.

Important: If you are launching the Profile Management Tool on an AIX platform from the Installation Complete page, and you get a JVM terminate error with exit code 721035, launch the Profile Management Tool in one of the following ways:

64-bit AIX platform
> If you are using the 64-bit version of AIX, use the **manageprofiles** command.

64-bit AIX platform
> If you are using the 32-bit version of AIX, use the command:
>
> `install_root/bin/ProfileManagement/pmt.sh`

Choose one of the following links for information about creating profiles using the Profile Management Tool or the `manageprofiles` command-line utility.

Related concepts:

"Configuring the environment using manageprofiles and wsadmin" on page 187
You can achieve the same configurations that you set up using the Profile Management Tool and deployment environment wizard by using the `manageprofiles` command-line utility and the `wsadmin` command.

Related tasks:

"Creating Process Server stand-alone profiles using manageprofiles" on page 240
Use the `manageprofiles` command-line utility to create Process Server stand-alone profiles for IBM Business Process Manager Standard .

Creating Process Center stand-alone profiles using the Profile Management Tool:

Use the Profile Management Tool to create Process Center stand-alone profiles.

Federating Process Center stand-alone profiles is not supported in version 7.5.

- Review Prerequisites for creating or augmenting profiles.

- `Solaris` When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/.Xdefaults file:

  `Eclipse*spacing:0`
  `Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1`

  After adding the lines, run the following command before starting the Profile Management Tool:

  `xrdb -load user_home/.Xdefaults`

Use the procedure in this task topic if you are:

- Creating a *Process Center* stand-alone profile for IBM Business Process Manager Standard .

The steps describe both **Advanced profile creation** and **Typical profile creation**.

1. Start the Profile Management Tool.

   Use one of the following commands:

- • `Linux` `UNIX` *install_root*/**bin/ProfileManagement/pmt.sh**
- • `Windows` *install_root***bin\\ProfileManagement\\pmt.bat**

  The Welcome page displays.

  See Starting the Profile Management Tool for other methods of starting this tool.

2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

   The **Profiles** tab displays.

   The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.

3. In the **Profiles** tab, click **Create**.

   The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the configuration for which you are creating the profile and expand the section.

   For example, if you are creating profiles for IBM BPM Advanced, select the option for **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**. If you are creating profiles for IBM BPM Standard, select the option for **IBM BPM Standard, Process Server Stand-alone** profile and click **Next**. If you are creating profiles for IBM BPM Express, select the option for **IBM BPM Express, Process Server Stand-alone** profile and click **Next**

   Note: You can also create WebSphere Application Server profiles with this Profile Management Tool. However, this documentation addresses creating IBM Business Process Manager profiles only.

   The Profile Creation Options page displays.

From the Profile Creation Options page, complete the stand-alone profile using either the **Typical profile creation** method or the **Advanced profile creation** method.

Related tasks:

"Using the typical option to create Process Center stand-alone profiles with the Profile Management Tool" Use the Profile Management Tool typical option to create Process Center stand-alone profiles.

"Using the advanced option to create Process Center stand-alone profiles with the Profile Management Tool" on page 140 Use the Profile Management Tool advanced option to create Process Center stand-alone profiles.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551 To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Using the typical option to create Process Center stand-alone profiles with the Profile Management Tool:

Use the Profile Management Tool typical option to create Process Center stand-alone profiles.

Complete the initial profile creation steps as described in "Creating Process Center stand-alone profiles using the Profile Management Tool" on page 137.

The steps describe the **Typical profile creation**. Use the typical profile creation option when you want the Profile Management Tool to:

- • Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
- • Install the administrative console.
- • Create a personal security certificate for the profile.

  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year.

- Create a root signing security certificate for signing other certificates.

  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years.

- Create a system service to run the server.

  Applicable only when your operating system and the privileges of your user account permit.

- Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.

1. On the Administrative Security page, enter values for the **User name** and **Password** and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

   Note: The **Next** button is disabled initially. There is no option to disable administrative security, thus you are required to specify the userid, password, and password confirmation in order to enable the **Next** button.

   Click **Next**.

2. On the Database Configuration - Part 1 page, perform the following actions:

   - From the **Select a database product** drop-down, select the database product to be used by the profile.

   - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

     If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

     If the database product you are using with the software already exists, select **Use an existing local or remote database** .

   - In the **Process Server database name** field, enter a name for the database or accept the **BPMDB** default value.

   - In the **Performance Data Warehouse database name** field, enter a name for the database or accept the **PDWDB** default value.

   - Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

     If you do not select the checkbox, the scripts are output to the default directory.

   - Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

   Click **Next** to display the Database Configuration - Part 2 page.

   The information on the Database Configuration - Part 2 page varies depending on the value specified in the **Select a database product** menu on the Database Configuration - Part 1 page.

3. In the Database Configuration - Part 2 page, complete the configuration for the database you selected. See the Database configuration fields for Profile Management Tool configuration information for details about each required field.

   You can configure parameters for the following databases:

   DB2   After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

   DB2 Universal Database for z/OS
           You cannot create a new database using DB2 Universal Database for z/OS. The Performance Data Warehouse database and, for a stand-alone server profile, the Process Server database, must exist. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

Microsoft SQL Server

After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

Oracle You cannot create a new database using this database.

Important: You must have a user ID that has SYSDBA privileges before creating any profile. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

4. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

You can proceed to the First Steps Console and start the server for the Process Center stand-alone profile you created.

If you did not run the database scripts during profile creation, you must load the database with system information after the database has been created.

Related tasks:

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

"Creating Process Center stand-alone profiles using the Profile Management Tool" on page 137
Use the Profile Management Tool to create Process Center stand-alone profiles.

Using the advanced option to create Process Center stand-alone profiles with the Profile Management Tool:

Use the Profile Management Tool advanced option to create Process Center stand-alone profiles.

Complete the initial profile creation steps as described in "Creating Process Center stand-alone profiles using the Profile Management Tool" on page 137.

The steps describe the **Advanced profile creation** . Use the advanced option to:
- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Deploy the administrative console.
- Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

1. In the Optional Application Deployment page, select whether to deploy the administrative console and the default application.

    Deploy the Sample applications
    Installs the IBM Business Process Manager and WebSphere Application Server sample applications. The sample applications are not recommended for deployment to production environments.

    Deploy the administrative console (recommended)
    Installs a web-based administrative console that manages the server.

    Deploy the default application
    Installs the default application that contains the Snoop, Hello, and HitCount Servlets.

    Click **Next** to display the Profile Name and Location page.

2. In the Profile Name and Location page, perform the following steps:

    a. In the **Profile name** field, specify a unique name or accept the default value.

    Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Naming considerations for profiles, nodes, servers, hosts, and cells for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

    b. In the **Profile directory** field, enter the directory for the profile or use the **Browse. . .** button to go to the profile directory.

    The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform. The following examples show the platform differences and *profile_name* is the name you specify:

    - `Linux` `UNIX` `install_root/profiles/profile_name`
    - `Windows` `install_root\profiles\profile_name`

    An error message is displayed if any of the following issues occur:
    - The *profile_name* you specify is not unique.
    - The directory you specify is not empty.
    - Your user ID does not have sufficient permissions for the directory.
    - There is not sufficient space to create the profile.

    c. Optional: Select the **Make this profile the default** check box to make the profile you are creating the default profile.

    Note: This check box is shown only if you have an existing profile on your system.

    When a profile is made to be the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See Profile commands in a multiprofile environment for more information.

    d. From the **Server runtime performance tuning setting** pull-down list, select a performance tuning level appropriate for the profile you are creating.

    e. Click **Next**.

    Note: If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

3. In the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

    - In the **Node name** field, enter a name for the node or accept the default value.

Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.

- In the **Server name** field, enter a name for the server or accept the default value.
- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next** to display the Administrative Security page.

4. On the Administrative Security page, enter values for the **User name** and **Password** and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

   Note: The **Next** button is disabled initially. There is no option to disable administrative security, thus you are required to specify the userid, password, and password confirmation in order to enable the **Next** button.

   Click **Next**.

5. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

   Perform the following actions:

   - To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.
   - To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:
     - In the **Path** field, enter the directory path to the existing certificate.
     - In the **Password** field, enter the password for the certificate
     - In the **Keystore type** field, select the keystore type for the certificate you are importing.
     - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
     - Click **Next** to display the Security Certificate (Part 2) page

   When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

6. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

   If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

   When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

   - `key.p12`: Contains the default personal certificate.
   - `trust.p12`: Contains the signer certificate from the default root certificate.
   - `root-key.p12`: Contains the root signing certificate.
   - `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower® signer certificate are in this keystore file.

- `deleted.p12`: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

An imported certificate is added to the `key.p12` file or the `root-key.p12` file.

If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

7. From the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**.

   The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

   Ports are recognized as being in use if the following conditions are satisfied:
   - The ports are assigned to a profile created under an installation performed by the current user.
   - The ports are currently in use.

   Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

   If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

   - Linux UNIX *profile_root*/**properties/portdef.props**

   - Windows *profile_root***properties\portdef.props**

   Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see the topic Updating ports in an existing profile in the WebSphere Application Server Network Deployment information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

If you are installing	Next step
On a Linux or Windows platform, *and have root or Administrator group privileges*	The Linux or Windows Service Definition page is displayed. Proceed to step 8.
On any other platform or as a nonroot user on a Linux or Windows platform	The Web Server Definition page is displayed. Proceed to step 9 on page 144.

8. In the Service Definition page, indicate whether or not to use a Windows service or Linux service to run IBM Business Process Manager

   Windows The Windows Service Definition page displays for the Windows platform only when the ID that installs the Windows service has the Administrator group privilege. If the profile is configured as a Windows service, the product starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Windows service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

   Important: If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is `Manual`). The user ID must not have spaces in its name, it must belong to the Administrator group, and it must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

IPv6 considerations when running profiles as Windows services

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

`Linux` The Linux Service Definition page is displayed only if the current operating system is a supported version of Linux and the current user has the appropriate permissions.

IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Linux service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

By default, IBM Business Process Manager is not selected to run as a Linux service.

To create the service, the user who runs the Profile Management Tool must be the root user. If you run the Profile Management Tool with a non-root user ID, the Linux Service Definition page is not displayed, and no service is created.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have proper privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

9. To include a Web server definition in the profile now, perform the following steps:
   a. Select the **Create a Web server definition** check box.
   b. Specify the Web server characteristics on the page, and click **Next**
   c. Specify the Web server characteristics on Part 2 of the page.

   If you use a Web server to route requests to IBM Business Process Manager, you need to include a Web server definition. You can include the definition now, or define the Web server to IBM Business Process Manager later. If you define the Web server definition during the creation of this profile, you can install the Web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the Web server to IBM Business Process Manager after you create this profile, you must define the Web server in a separate profile.

   d. Click **Next**.
10. Configure the databases using a design file.
    a. Select **Use a database design file** .
    b. Click **Browse**.
    c. Specify the fully qualified path name for the design file.
    d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

    Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

    e. Click **Next**.

11. On the Database Configuration - Part 1 page, perform the following actions:
    - From the **Select a database product** drop-down, select the database product to be used by the profile.
    - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

      If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

      If the database product you are using with the software already exists, select **Use an existing local or remote database** .
    - In the **Process Server database name** field, enter a name for the database or accept the **BPMDB** default value.
    - In the **Performance Data Warehouse database name** field, enter a name for the database or accept the **PDWDB** default value.
    - Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

      If you do not select the checkbox, the scripts are output to the default directory.
    - Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

    Click **Next** to display the Database Configuration - Part 2 page.

    The information on the Database Configuration - Part 2 page varies depending on the value specified in the **Select a database product** menu on the Database Configuration - Part 1 page.
12. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. See the Database configuration fields for Profile Management Tool configuration information for details about each required field.

    You can configure parameters for the following databases:

    DB2   After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

    DB2 Universal Database for z/OS
    > You cannot create a new database using DB2 Universal Database for z/OS. The Performance Data Warehouse database and, for a stand-alone server profile, the Process Server database, must exist. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

    Microsoft SQL Server
    > After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

    Oracle   You cannot create a new database using this database.

    > **Important:** You must have a user ID that has SYSDBA privileges before creating any profile. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.
13. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

    The progress of the configuration displays on the Profile Configuration Progress window.

    When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

    Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

You can proceed to the First Steps Console and start the server for the Process Center stand-alone profile you created.

If you did not run the database scripts during profile creation, you must load the database with system information after the database has been created.

Related tasks:

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

"Creating Process Center stand-alone profiles using the Profile Management Tool" on page 137
Use the Profile Management Tool to create Process Center stand-alone profiles.

Creating stand-alone profiles using the Profile Management Tool:

You can use the Profile Management Tool to create the profile for a stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

Federating Process Server stand-alone profiles is not supported in version 7.5.

- Review Prerequisites for creating or augmenting profiles.

- Solaris When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/ .Xdefaults file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

Tip: Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create profiles, as described in Configuring the software using command-line utilities and wsadmin.

Use the procedure in this task topic if you are creating a stand-alone *Process Server* profile for IBM Business Process Manager Standard. The steps describe both **Typical profile creation** and **Advanced profile creation** options.

1. Start the Profile Management Tool. For methods to start this tool, see Starting the Profile Management Tool. The Welcome page displays.

2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

   The **Profiles** tab displays.

   The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.

3. In the **Profiles** tab, click **Create**.

The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the configuration for which you are creating the profile and expand the section.

   For example, if you are creating profiles for IBM BPM Advanced, select the option for **IBM BPM Advanced, Process Server Stand-alone** profile and click **Next**. If you are creating profiles for IBM BPM Standard, select the option for **IBM BPM Standard, Process Server Stand-alone** profile and click **Next**. If you are creating profiles for IBM BPM Express, select the option for **IBM BPM Express, Process Server Stand-alone** profile and click **Next**

   Note: You can also create WebSphere Application Server profiles with this Profile Management Tool. However, this documentation addresses creating IBM Business Process Manager profiles only.

   The Profile Creation Options page displays.

5. On the Profile Creation Options page, create the stand-alone profile using one of the following options:

   Typical profile creation
   > Creates a profile with default configuration settings.

   Advanced profile creation
   > Creates a profile using the configuration values you specify.

   Table 60 provides more information about the options for creating a stand-alone profile.

Table 60. Selecting the creation option for your stand-alone profile

Select	When you want to . . .

Typical profile creation	Allow the Profile Management Tool to perform the following actions:
	• Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
	• Install the administrative console.
	• Create a personal security certificate for the profile.
	  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year.
	• Create a root signing security certificate for signing other certificates.
	  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years.
	• Create a system service to run the server.
	  Applicable only when your operating system and the privileges of your user account permit.
	• Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.

Advanced profile creation	• Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable). • Deploy the administrative console. • Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets). • Create a web server definition. • Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

Continue creating one of the following types of Process Server stand-alone profiles:

• **Typical profile creation**

- **Advanced profile creation**

Related tasks:

"Creating a typical Process Server stand-alone profile with the Profile Management tool"
You can use the Profile Management Tool to create the profile for a typical stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

"Creating an advanced Process Server stand-alone profile with the Profile Management tool" on page 151
You can use the Profile Management Tool to create an advanced profile for a stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Related information:

Creating database design files by using the database design tool

Creating a typical Process Server stand-alone profile with the Profile Management tool:

You can use the Profile Management Tool to create the profile for a typical stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

Complete the initial configuration steps provided in "Creating stand-alone profiles using the Profile Management Tool" on page 146.

Use the procedure in this task topic if you are creating a typical stand-alone *Process Server* profile for IBM Business Process Manager Standard. The steps describe the **Typical profile creation**.

1. On the Administrative Security page, enter values for the **User name** and **Password** and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

   Note: The **Next** button is disabled initially. There is no option to disable administrative security, thus you are required to specify the userid, password, and password confirmation in order to enable the **Next** button.

   Click **Next**.

2. From the Process Server Configuration page, set the values for the following parameters

   Environment name
   Specifies the name of the environment you are configuring.

   Select an environment type

   The following options are available:

   Production
   Select **Production** if the server is to serve in a production capacity.

   Stage Select **Stage** if the server is to serve as a staging platform to be used as a preproduction server.

   Test Select **Test** if the server you are configuring is to be used as a testing environment.

   The environment type, that is set during installation, refers to how the Process Server is used and determines the environment variables and values that are applicable to the server. Environment-specific variables can be set for each process application, and then define values for each type of environment in which a process runs. For example, in what capacity is the Process

Server to be used - *production*, *stage* or *test* . Load testing might be done on a test server, while a stage environment type might be used as a temporary location to host changes before putting those changes into production. You might specify **Stage** as the **Environment type** if the Process Server you are configuring is to be accessed and used to review content and new functionality.

Process Center configuration information

Specify the configuration details to connect to this Process Center. Alternatively, if you want to run in offline mode, select the check box to use the server in offline mode.

Use server offline

Indicate whether the server you are configuring is an offline server. An offline server is a Process Server that is not connected to the Process Center.

Offline servers can be used when deploying snapshots of process applications. However the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

Protocol

Select either **http://** or **https://** as the connection protocol to the Process Center.

Host name

Type the host or virtual host that this Process Server needs to communicate with Process Center. Use a fully qualified host name.

In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

Port Type the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.

User name

Type a valid user name that exists on the Process Center. Process Server will connect to Process Center as this user.

Password

Type the password for the above user.

You can test the connection to the Process Center by clicking **Test Connection**.

Click **Next**.

3. On the Database Configuration page, perform the following actions:

- From the **Select a database product** drop-down, select the database product to be used by the profile.

- Select whether to **Create a new local database** or to **Use an existing local or remote database**.

  If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

  If the database product you are using with the software already exists, select **Use an existing local or remote database** .

- In the **Process Server database name** field, enter a name for the database or accept the **BPMDB** default value.

- In the **Performance Data Warehouse database name** field, enter a name for the database or accept the **PDWDB** default value.

- Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

  If you do not select the checkbox, the scripts are output to the default directory.

- Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

  For Oracle, the **Process Server database name** and the **Performance Data Warehouse database** name (actually this is the database instance name) can be the same as the **Common database name**. For all other database types, the database names must be unique.

4. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. See the Database configuration fields for Profile Management Tool configuration information for details about each required field.

   You can configure parameters for the following databases:

   DB2   After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

   DB2 Universal Database for z/OS
   You cannot create a new database using DB2 Universal Database for z/OS. The Performance Data Warehouse database and, for a stand-alone server profile, the Process Server database, must exist. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

   Microsoft SQL Server
   After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

   Oracle You cannot create a new database using this database.

   Important: You must have a user ID that has SYSDBA privileges before creating any profile. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

5. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When the profile creation is complete, the Profile Complete page is displayed with the message `The Profile Management tool created the profile successfully`.

   Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:
   - `The Profile Management tool created the profile but errors occurred`, which indicates that profile creation completed but errors were generated.
   - `The Profile Management tool cannot create the profile`, which indicates that profile creation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.
   You have completed the steps to create the profile with default configuration settings.

From the First Steps Console, you can start the Process Server stand-alone profile you have just created.

If you did not run the database scripts during profile creation, you must load the database with system information after the database has been created.

Related tasks:

"Creating stand-alone profiles using the Profile Management Tool" on page 146
You can use the Profile Management Tool to create the profile for a stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating an advanced Process Server stand-alone profile with the Profile Management tool:

You can use the Profile Management Tool to create an advanced profile for a stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

Complete the initial configuration steps provided in "Creating stand-alone profiles using the Profile Management Tool" on page 146.

Use the procedure in this task topic if you are creating an advanced stand-alone *Process Server* profile for IBM Business Process Manager Standard. The steps describe the **Advanced profile creation** option.

1. In the Optional Application Deployment page, select the applications you want to deploy to the profile environment.

   Deploy the Sample applications
   Installs the IBM Business Process Manager and WebSphere Application Server sample applications. The sample applications are not recommended for deployment to production environments.

   Deploy the administrative console (recommended)
   Installs a web-based administrative console that manages the server.

   Deploy the default application
   Installs the default application that contains the Snoop, Hello, and HitCount Servlets.

2. In the Profile Name and Location page, perform the following steps:

   a. In the **Profile name** field, specify a unique name or accept the default value.

      Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Naming considerations for profiles, nodes, servers, hosts, and cells for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

   b. In the **Profile directory** field, enter the directory for the profile or use the **Browse. . .** button to go to the profile directory.

      The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform. The following examples show the platform differences and *profile_name* is the name you specify:

      - **Linux** **UNIX** `install_root/profiles/profile_name`
      - **Windows** `install_root\profiles\profile_name`

      An error message is displayed if any of the following issues occur:
      - The *profile_name* you specify is not unique.
      - The directory you specify is not empty.
      - Your user ID does not have sufficient permissions for the directory.
      - There is not sufficient space to create the profile.

   c. Optional: Select the **Make this profile the default** check box to make the profile you are creating the default profile.

Note: This check box is shown only if you have an existing profile on your system.

When a profile is made to be the default profile, commands work automatically with it. The first profile that you create on a workstation is the default profile. The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See Profile commands in a multiprofile environment for more information.

d. From the **Server runtime performance tuning setting** pull-down list, select a performance tuning level appropriate for the profile you are creating.

e. Click **Next**.

Note: If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

3. In the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

   - In the **Node name** field, enter a name for the node or accept the default value.

     Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.

   - In the **Server name** field, enter a name for the server or accept the default value.

   - In the **Host name** field, enter a name for the host or accept the default value.

   - In the **Cell name** field, enter a name for the cell or accept the default value.

   Click **Next** to display the Administrative Security page.

4. On the Administrative Security page, enter values for the **User name** and **Password** and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

   Note: The **Next** button is disabled initially. There is no option to disable administrative security, thus you are required to specify the userid, password, and password confirmation in order to enable the **Next** button.

   Click **Next**.

5. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

   Perform the following actions:

   - To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.

   - To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:

     – In the **Path** field, enter the directory path to the existing certificate.

     – In the **Password** field, enter the password for the certificate

     – In the **Keystore type** field, select the keystore type for the certificate you are importing.

     – In the **Keystore alias** field, select the keystore alias for the certificate you are importing.

     – Click **Next** to display the Security Certificate (Part 2) page

   When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

6. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is WebAS. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the java.security file.

When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

- key.p12: Contains the default personal certificate.
- trust.p12: Contains the signer certificate from the default root certificate.
- root-key.p12: Contains the root signing certificate.
- default-signers.p12: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- deleted.p12: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

An imported certificate is added to the key.p12 file or the root-key.p12 file.

If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

7. From the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**.

The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:
- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

- `Linux` `UNIX` *profile_root*/**properties/portdef.props**

- `Windows` *profile_root***properties\portdef.props**

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see the topic Updating ports in an existing profile in the WebSphere Application Server Network Deployment information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

If you are installing	Next step
On a Linux or Windows platform, *and have root or Administrator group privileges*	The Linux or Windows Service Definition page is displayed. Proceed to step 8 on page 154.

If you are installing	Next step
On any other platform or as a nonroot user on a Linux or Windows platform	The Web Server Definition page is displayed. Proceed to step 9.

8. In the Service Definition page, indicate whether or not to use a Windows service or Linux service to run IBM Business Process Manager

   Windows The Windows Service Definition page displays for the Windows platform only when the ID that installs the Windows service has the Administrator group privilege. If the profile is configured as a Windows service, the product starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Windows service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

   Important: If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, it must belong to the Administrator group, and it must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

   During profile deletion, you can remove the Windows service that is added during profile creation.

   IPv6 considerations when running profiles as Windows services
   Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

   Linux The Linux Service Definition page is displayed only if the current operating system is a supported version of Linux and the current user has the appropriate permissions.

   IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Linux service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

   By default, IBM Business Process Manager is not selected to run as a Linux service.

   To create the service, the user who runs the Profile Management Tool must be the root user. If you run the Profile Management Tool with a non-root user ID, the Linux Service Definition page is not displayed, and no service is created.

   You must specify a user name under which the service runs.

   To delete a Linux service, the user must be the root user or have proper privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

   Click **Next** to continue

9. To include a Web server definition in the profile now, perform the following steps:
   a. Select the **Create a Web server definition** check box.
   b. Specify the Web server characteristics on the page, and click **Next**
   c. Specify the Web server characteristics on Part 2 of the page.

If you use a Web server to route requests to IBM Business Process Manager, you need to include a Web server definition. You can include the definition now, or define the Web server to IBM Business Process Manager later. If you define the Web server definition during the creation of this profile, you can install the Web server and its plug-in after you create the profile. However, you must install both to the paths that you specify on the Web Server Definition pages. If you define the Web server to IBM Business Process Manager after you create this profile, you must define the Web server in a separate profile.

   d. Click **Next**.

10. Configure the databases using a design file.

   a. Select **Use a database design file** .

   b. Click **Browse**.

   c. Specify the fully qualified path name for the design file.

   d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

      Important: If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

   e. Click **Next**.

11. If you chose not to configure the databases using a design file, specify the database details on the database configuration panels. For details, see the following steps:

   a. On the Database Configuration page, perform the following actions:

      • From the **Select a database product** drop-down, select the database product to be used by the profile.

      • Select whether to **Create a new local database** or to **Use an existing local or remote database**.

         If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

         If the database product you are using with the software already exists, select **Use an existing local or remote database** .

      • In the **Process Server database name** field, enter a name for the database or accept the **BPMDB** default value.

      • In the **Performance Data Warehouse database name** field, enter a name for the database or accept the **PDWDB** default value.

      • Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

         If you do not select the checkbox, the scripts are output to the default directory.

      • Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

      For Oracle, the **Process Server database name** and the **Performance Data Warehouse database** name (this is the database instance name) can be the same as the **Common database name**. For all other database types, the database names must be unique.

   b. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. See the Database configuration fields for Profile Management Tool configuration information for details about each required field.

      You can configure parameters for the following databases:

      DB2   After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

DB2 Universal Database for z/OS

You cannot create a new database using DB2 Universal Database for z/OS. The Performance Data Warehouse database and, for a stand-alone server profile, the Process Server database, must exist. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

Microsoft SQL Server

After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

Oracle You cannot create a new database using this database.

Important: You must have a user ID that has SYSDBA privileges before creating any profile.

After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Summary page.

12. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

From the First Steps Console, you can start the Process Server stand-alone profile you have just created.

If you did not run the database scripts during profile creation, you must load the database with system information after the database has been created.

Related tasks:

"Creating stand-alone profiles using the Profile Management Tool" on page 146
You can use the Profile Management Tool to create the profile for a stand-alone environment. The stand-alone environment functions independently from all other servers and is managed from its own administrative console.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Creating network deployment profiles

If you use the **configureNode** command to create and configure your deployment environment, no further configuration is required. Otherwise, you must create a deployment manager profile and one or more custom profiles before creating the deployment environment.

Creating Process Center deployment manager profiles using the Profile Management Tool:

You can use the Profile Management Tool to create the Process Center deployment manager profiles for your network deployment configuration.

- Review Prerequisites for creating or augmenting profiles.

- Solaris When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root/*`.Xdefaults` file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

  After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

Use the procedure in this task topic if you are:

- Creating a *process center* deployment manager profile for an IBM Business Process Manager Standard network deployment configuration

Note: Creating the deployment manager profile is *one step in a multistep process* for creating a network deployment configuration.

A network deployment environment configuration is any *non-standalone configuration* of IBM Business Process Manager Standard.

1. Start the Profile Management Tool.

   Use one of the following commands:

   - Linux UNIX *install_root*/**bin/ProfileManagement/pmt.sh**

   - Windows *install_root***bin\\ProfileManagement\\pmt.bat**

   The Welcome page displays.

   See Starting the Profile Management Tool for other methods of starting this tool.

2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

   The **Profiles** tab displays.

   The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.

3. In the **Profiles** tab, click **Create**.

   The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager Standard configuration for which you are creating the deployment manager profile and expand the section. Select the option for IBM BPM Standard, Process Center deployment manager profile and click **Next**.

   Note: You can also create WebSphere Application Server profiles with this Profile Management Tool. However, this documentation addresses creating IBM Business Process Manager profiles only.

   The Profile Creation Options page displays.

5. From the Profile Creation Options page, decide whether to create the deployment manager profile using the **Advanced** or **Typical** option.

Table 61. Selecting the profile creation option for your deployment manager profile

Select	When you want to . . .
Advanced profile creation	• Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable). • Deploy the administrative console. • Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets). • Create a web server definition. • Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

Typical profile creation	Let the Profile Management Tool
	• Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
	• Install the administrative console.
	• Create a personal security certificate for the profile.
	  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year.
	• Create a root signing security certificate for signing other certificates.
	  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years.
	• Create a system service to run the server.
	  Applicable only when your operating system and the privileges of your user account permit.
	• Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.

You have created the Process Center deployment manager profile.

You can choose either the typical or the advanced option of the Profile Management Tool to complete configuration for your Process Center deployment manager profiles.

Related tasks:

"Using the advanced option to create Process Center deployment manager profiles with the Profile Management Tool"
You can use the advanced option of the Profile Management Tool to create the Process Center deployment manager profiles for your network deployment configuration.

"Using the typical option to create Process Center deployment manager profiles with the Profile Management Tool" on page 163
You can use the typical option of the Profile Management Tool to create the Process Center deployment manager profiles for your network deployment configuration.

Using the advanced option to create Process Center deployment manager profiles with the Profile Management Tool:

You can use the advanced option of the Profile Management Tool to create the Process Center deployment manager profiles for your network deployment configuration.

Complete the initial profile creation steps as described in "Creating Process Center deployment manager profiles using the Profile Management Tool" on page 157.

Use the advanced option to perform the following tasks:

- Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
- Deploy the administrative console.
- Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets).
- Create a web server definition.
- Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

1. In the Optional Application Deployment page, select whether to deploy the administrative console

   Click **Next** to display the Profile Name and Location page.

2. In the Profile Name and Location page, perform the following steps:

   a. In the **Profile name** field, specify a unique name or accept the default value.

      Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Profile commands in a multiprofile environment for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

   b. In the **Profile directory** field, enter the directory for the profile or use the **Browse. . .** button to go to the profile directory.

      The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform:

      - Linux   UNIX   `install_root`/profiles/`profile_name`
      - Windows   `install_root`\profiles\`profile_name`

      where *profile_name* is the name you specified. An error message is displayed if:
      - The *profile_name* you specify is not unique.
      - The directory you specify is not empty.
      - Your user ID does not have sufficient permissions for the directory.
      - There is not sufficient space to create the profile.

   c. Optional: Select the **Make this profile the default** check box if you wish to make the profile you are creating the default profile.

      When a profile is made to be the default profile, commands work automatically with it.

      Note: This check box appears only if you have an existing profile on your system.

      The first profile that you create on a workstation is the default profile.

      The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See Profile commands in a multiprofile environment for more information.

3. In the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

   - In the **Node name** field, enter a name for the node or accept the default value.

     Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.

   - In the **Host name** field, enter a name for the host or accept the default value.

   - In the **Cell name** field, enter a name for the cell or accept the default value.

   Click **Next** to display the Administrative Security page.

4. On the Administrative Security page, enter values for the **User name** and **Password** and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

Note: The **Next** button is disabled initially. There is no option to disable administrative security, thus you are required to specify the userid, password, and password confirmation in order to enable the **Next** button.

Click **Next**.

5. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

   Perform the following actions:

   - To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.

   - To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:

     – In the **Path** field, enter the directory path to the existing certificate.

     – In the **Password** field, enter the password for the certificate

     – In the **Keystore type** field, select the keystore type for the certificate you are importing.

     – In the **Keystore alias** field, select the keystore alias for the certificate you are importing.

     – Click **Next** to display the Security Certificate (Part 2) page

   When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

6. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

   If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

   When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

   - `key.p12`: Contains the default personal certificate.

   - `trust.p12`: Contains the signer certificate from the default root certificate.

   - `root-key.p12`: Contains the root signing certificate.

   - `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.

   - `deleted.p12`: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.

   - `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

   These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

   An imported certificate is added to the `key.p12` file or the `root-key.p12` file.

   If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

7. From the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**.

The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:
- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

- `Linux` `UNIX` *profile_root*/**properties/portdef.props**
- `Windows` *profile_root***properties\portdef.props**

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see the topic Updating ports in an existing profile in the WebSphere Application Server Network Deployment information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

8. In the Service Definition page, indicate whether or not to use a Windows service or Linux service to run IBM Business Process Manager

   `Windows` The Windows Service Definition page displays for the Windows platform only when the ID that installs the Windows service has the Administrator group privilege. If the profile is configured as a Windows service, the product starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Windows service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

   Important: If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is `Manual`). The user ID must not have spaces in its name, it must belong to the Administrator group, and it must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

   During profile deletion, you can remove the Windows service that is added during profile creation.

   IPv6 considerations when running profiles as Windows services
   Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6. Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

   `Linux` The Linux Service Definition page is displayed only if the current operating system is a supported version of Linux and the current user has the appropriate permissions.

IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Linux service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

By default, IBM Business Process Manager is not selected to run as a Linux service.

To create the service, the user who runs the Profile Management Tool must be the root user. If you run the Profile Management Tool with a non-root user ID, the Linux Service Definition page is not displayed, and no service is created.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have proper privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

9. On the Database Configuration page, perform the following actions:

   - From the **Select a database product** drop-down, select the database product to be used by the profile.
   - Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the SQL scripts used to create the database tables are written.

     If you do not select the checkbox, the scripts are output to the default directory.
   - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

     If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

     If the database product you are using with the software already exists, select **Use an existing local or remote database** .
   - In the **Common database name** field, enter a name for the common database or accept the default **CMNDB** value.
   - Select **Run database scripts to create database tables** if you want to run the database scripts automatically (as part of the profile creation process). If you do not select **Run database scripts to create database tables**, you or the database administrator can run the scripts manually after profile creation completes.

     Attention:

     – If you select **Use an existing local or remote database** and the tables already exist, make sure **Run database scripts to create database tables** is *not* selected. If both options are selected, errors occur and profile creation fails.

     – If you select **Run database scripts to create database tables** and you are configuring Oracle or DB2 for z/OS, the Profile Management Tool needs access to the database system to create the tables. If the Profile Management Tool cannot connect to the database system to create the tables, a database validation error is thrown.

10. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

    The progress of the configuration displays on the Profile Configuration Progress window.

    When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

    Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

    - **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
    - **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

You have created the Process Center deployment manager profile.

You can add custom (managed nodes) profiles to be managed by the deployment manager and then build the clusters and configure the components on the clusters.

The components available for configuration in a network deployment environment vary based on the IBM Business Process Manager Standard (Advanced or Standard) installed on your system.

You can build the clusters and configure the IBM Business Process Manager Standard components for your network deployment configuration.

Related concepts:

"Configuring a network deployment environment using the deployment environment wizard" on page 592
After performing a Custom installation and creating the deployment manager and custom (managed node) profiles, you can create a network deployment configuration.

"Configuring a network deployment environment using the administrative console" on page 606
After you perform a custom installation and create the deployment manager and custom (managed node) profiles, you can create a network deployment environment using the administrative console.

Related tasks:

"Creating Process Center deployment manager profiles using the Profile Management Tool" on page 157
You can use the Profile Management Tool to create the Process Center deployment manager profiles for your network deployment configuration.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Using the typical option to create Process Center deployment manager profiles with the Profile Management Tool:

You can use the typical option of the Profile Management Tool to create the Process Center deployment manager profiles for your network deployment configuration.

Complete the initial profile creation steps as described in "Creating Process Center deployment manager profiles using the Profile Management Tool" on page 157.

Use the typical option if you want the Profile Management tool to perform the following tasks:
- Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
- Install the administrative console.
- Create a personal security certificate for the profile.

  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year.
- Create a root signing security certificate for signing other certificates.

  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years.

- Create a system service to run the server.

  Applicable only when your operating system and the privileges of your user account permit.
- Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.

1. On the Administrative Security page, enter values for the **User name** and **Password** and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

   Note: The **Next** button is disabled initially. There is no option to disable administrative security, thus you are required to specify the userid, password, and password confirmation in order to enable the **Next** button.

   Click **Next**.

2. On the Database Configuration page, perform the following actions:
   - From the **Select a database product** drop-down, select the database product to be used by the profile.
   - Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the SQL scripts used to create the database tables are written.

     If you do not select the checkbox, the scripts are output to the default directory.
   - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

     If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

     If the database product you are using with the software already exists, select **Use an existing local or remote database** .
   - In the **Common database name** field, enter a name for the common database or accept the default **CMNDB** value.
   - Select **Run database scripts to create database tables** if you want to run the database scripts automatically (as part of the profile creation process). If you do not select **Run database scripts to create database tables**, you or the database administrator can run the scripts manually after profile creation completes.
     Attention:
     – If you select **Use an existing local or remote database** and the tables already exist, make sure **Run database scripts to create database tables** is *not* selected. If both options are selected, errors occur and profile creation fails.
     – If you select **Run database scripts to create database tables** and you are configuring Oracle or DB2 for z/OS, the Profile Management Tool needs access to the database system to create the tables. If the Profile Management Tool cannot connect to the database system to create the tables, a database validation error is thrown.

3. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

   Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:
   - **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
   - **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

You have created the Process Center deployment manager profile.

You can add custom (managed nodes) profiles to be managed by the deployment manager and then build the clusters and configure the components on the clusters.

The components available for configuration in a network deployment environment vary based on the IBM Business Process Manager Standard (Advanced or Standard) installed on your system.

You can build the clusters and configure the IBM Business Process Manager Standard components for your network deployment configuration.

Related concepts:

"Configuring a network deployment environment using the deployment environment wizard" on page 592
After performing a Custom installation and creating the deployment manager and custom (managed node) profiles, you can create a network deployment configuration.

"Configuring a network deployment environment using the administrative console" on page 606
After you perform a custom installation and create the deployment manager and custom (managed node) profiles, you can create a network deployment environment using the administrative console.

Related tasks:

"Creating Process Center deployment manager profiles using the Profile Management Tool" on page 157
You can use the Profile Management Tool to create the Process Center deployment manager profiles for your network deployment configuration.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Creating Process Center custom profiles (managed nodes) using the Profile Management Tool:

You can use the Profile Management Tool to create and configure Process Center custom profiles (managed node) for your network deployment configuration.

- You have created the deployment manager profile for process center.
- The Deployment manager is running.

Use the procedure in this task topic if you are:

- Creating a *process center* custom profile (managed nodes) for your IBM Business Process Manager Standard network deployment configuration

Note: Information that applies to process center only or to process server only, is so noted in this procedure.

This topic describes both the **Typical** and **Advanced** methods for creating custom profiles (managed nodes).

You can choose to federate the node to an existing deployment manager during the creation process, or federate it later using the **addNode** command.

1. Start the Profile Management Tool.

   Use one of the following commands:

   - Linux   UNIX   *install_root*/**bin/ProfileManagement/pmt.sh**

   - Windows   *install_root***bin\ProfileManagement\pmt.bat**

   The Welcome page displays.

2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

   The **Profiles** tab displays.

   The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.

3. In the **Profiles** tab, click **Create**.

   The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager configuration (Advanced or Standard) for which you are creating the custom profile and expand the section. Select the option for Custom profile and click **Next**.

   Note: You can also create WebSphere Application Server profiles with this Profile Management Tool. However, this documentation addresses creating IBM Business Process Manager profiles only.

   The Profile Creation Options page displays.

5. From the Profile Creation Options page, decide whether to create the custom profile using the **Advanced** or **Typical** option.

Table 62. Selecting the profile creation option for your custom profile

Select	When you want to . . .
Advanced profile creation	Perform the following actions: • Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable). • Deploy the administrative console. • Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets). • Create a web server definition. • Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

Table 62. Selecting the profile creation option for your custom profile (continued)

Select	When you want to . . .
Typical profile creation	Let the Profile Management Tool perform the following actions: • Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell. • Install the administrative console. • Create a personal security certificate for the profile. The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year. • Create a root signing security certificate for signing other certificates. The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years. • Create a system service to run the server. Applicable only when your operating system and the privileges of your user account permit. • Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.

You can choose either the typical or the advanced option to complete configuration for your Process Center custom profiles (managed nodes) using the Profile Management Tool.

Related tasks:

"Using the advanced option to create Process Center custom profiles with the Profile Management Tool"
Use the Profile Management Tool advanced option to create and configure Process Center custom profiles (managed nodes) for your network deployment configuration.

"Using the typical option to create Process Center custom profiles with the Profile Management Tool" on page 171
Use the Profile Management Tool typical to create and configure Process Center custom profiles (managed nodes) for your network deployment configuration.

Using the advanced option to create Process Center custom profiles with the Profile Management Tool:

Use the Profile Management Tool advanced option to create and configure Process Center custom profiles (managed nodes) for your network deployment configuration.

Complete the initial profile creation steps as described in "Creating Process Center custom profiles (managed nodes) using the Profile Management Tool" on page 165.

Use the advanced option if you want to perform the following actions:
• Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
• Deploy the administrative console.
• Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets).
• Create a web server definition.
• Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

1. In the Profile Name and Location page, perform the following steps:
   a. In the **Profile name** field, specify a unique name or accept the default value.

Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Profile commands in a multiprofile environment for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

b. In the **Profile** field, enter the directory for the profile or use the **Browse. . .** button to go to the profile directory.

The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform:

- Linux UNIX *install_root*/profiles/*profile_name*

- Windows *install_root*\profiles*profile_name*

where *profile_name* is the name you specified. An error message is displayed if:

- The *profile_name* you specify is not unique.
- The directory you specify is not empty.
- Your user ID does not have sufficient permissions for the directory.
- There is not sufficient space to create the profile.

c. Optional: Select the **Make this profile the default** check box if you wish to make the profile you are creating the default profile.

When a profile is made to be the default profile, commands work automatically with it.

Note: This check box appears only if you have an existing profile on your system.

The first profile that you create on a workstation is the default profile.

The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See Profile commands in a multiprofile environment for more information.

d. Click **Next**.

Note: If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

2. In the Node and Host Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value.

Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.

- In the **Host name** field, enter a name for the host or accept the default value.

Click **Next** to display the Federation page.

3. In the **Federation** page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port number of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Check **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.

- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation

- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

Click **Next** to continue.

4. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

   Perform the following actions:

   - To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.

   - To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:
     - In the **Path** field, enter the directory path to the existing certificate.
     - In the **Password** field, enter the password for the certificate
     - In the **Keystore type** field, select the keystore type for the certificate you are importing.
     - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
     - Click **Next** to display the Security Certificate (Part 2) page

   When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

5. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

   If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

   When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

   - `key.p12`: Contains the default personal certificate.
   - `trust.p12`: Contains the signer certificate from the default root certificate.

- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.
- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

An imported certificate is added to the `key.p12` file or the `root-key.p12` file.

If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

6. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

You have created your IBM Business Process Manager profile.

The node within the profile is empty until you federate it and use the administrative console to customize it.

In a deployment environment, you must create and configure databases, create other custom profiles and federate them to your deployment manager, create servers, create clusters if you want workload management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

Related tasks:

"Creating Process Center custom profiles (managed nodes) using the Profile Management Tool" on page 165
You can use the Profile Management Tool to create and configure Process Center custom profiles (managed node) for your network deployment configuration.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Using the typical option to create Process Center custom profiles with the Profile Management Tool:

Use the Profile Management Tool typical to create and configure Process Center custom profiles (managed nodes) for your network deployment configuration.

Complete the initial profile creation steps as described in "Creating Process Center custom profiles (managed nodes) using the Profile Management Tool" on page 165.

Use the typical option if you want the Profile Management Tool to perform the following actions:
- Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
- Install the administrative console.
- Create a personal security certificate for the profile.

  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year.
- Create a root signing security certificate for signing other certificates.

  The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years.
- Create a system service to run the server.

  Applicable only when your operating system and the privileges of your user account permit.
- Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.

1. In the **Federation** page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port number of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

   Important:

   Check **Federate this node later** if any one of the following situations is true:
   - You plan to use this custom node as a migration target.
   - Another profile is being federated. (Node federation must be serialized.)
   - The deployment manager is not running or you are not sure if it is running.
   - The deployment manager has the SOAP connector disabled
   - The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
   - The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
   - The deployment manager does not have a JMX administrative port enabled.
   - The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

   Processing associated with federating the node as part of custom profile creation:
   - The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).

- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

   Click **Next** to continue.
2. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

   Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:
   - **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
   - **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

You have created your IBM Business Process Manager profile.

The node within the profile is empty until you federate it and use the administrative console to customize it.

In a deployment environment, you must create and configure databases, create other custom profiles and federate them to your deployment manager, create servers, create clusters if you want workload management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

Related tasks:

"Creating Process Center custom profiles (managed nodes) using the Profile Management Tool" on page 165
You can use the Profile Management Tool to create and configure Process Center custom profiles (managed node) for your network deployment configuration.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Creating Process Server deployment manager profiles using the Profile Management Tool:

You can use the Profile Management Tool to create the Process Server deployment manager profiles for your network deployment configuration.
- Review Prerequisites for creating or augmenting profiles.

- ▩ Solaris ▩ When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/ `.Xdefaults` file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

Use the procedure in this task topic if you are:

- Creating a *Process Server* deployment manager profile for an IBM Business Process Manager Standard network deployment configuration

Note: Creating the deployment manager profile is *one step in a multistep process* for creating a network deployment configuration.

A network deployment environment configuration is any *non-standalone configuration* of IBM Business Process Manager Standard.

1. Start the Profile Management Tool.

   Use one of the following commands:

   - **Linux** **UNIX** *install_root*/**bin/ProfileManagement/pmt.sh**

   - **Windows** *install_root***bin\ProfileManagement\pmt.bat**

   The Welcome page displays.

   See Starting the Profile Management Tool for other methods of starting this tool.

2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

   The **Profiles** tab displays.

   The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.

3. In the **Profiles** tab, click **Create**.

   The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager Standard configuration for which you are creating the deployment manager profile and expand the section. Select the option for IBM BPM Standard, Process Server deployment manager profile and click **Next**.

   Note: You can also create WebSphere Application Server profiles with this Profile Management Tool. However, this documentation addresses creating IBM Business Process Manager profiles only.

   The Profile Creation Options page displays.

5. From the Profile Creation Options page, decide whether to create the deployment manager profile using the **Advanced** or **Typical** option.

Table 63. Selecting the profile creation option for your deployment manager profile

Select	When you want to . . .
Advanced profile creation	Perform the following steps: • Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable). • Deploy the administrative console. • Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets). • Create a web server definition. • Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

Table 63. Selecting the profile creation option for your deployment manager profile (continued)

Select	When you want to . . .

Typical profile creation	Let the Profile Management Tool perform the following steps:
	
	• Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
	• Install the administrative console.
	• Create a personal security certificate for the profile.
	
	  The certificate has a personal key and private key, each with a default value of `WebAS` (you must change this password). The expiration period is one year.
	• Create a root signing security certificate for signing other certificates.
	
	  The certificate has a personal key and private key, each with a default value of `WebAS` (you must change this password). The expiration period is 15 years.
	• Create a system service to run the server.
	
	  Applicable only when your operating system and the privileges of your user account permit.
	• Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.

You can choose either the typical or the advanced option to complete configuration for your Process Server deployment manager profile.

Related tasks:

"Using the typical option to create Process Server deployment manager profiles with the Profile Management Tool"
Use the Profile Management Tool typical option to create Process Server deployment manager profiles for your network deployment configuration.

"Using the advanced option to create Process Server deployment manager profiles with the Profile Management Tool" on page 176
Use the Profile Management Tool advanced option to create Process Server deployment manager profiles for your network deployment configuration.

Using the typical option to create Process Server deployment manager profiles with the Profile Management Tool:

Use the Profile Management Tool typical option to create Process Server deployment manager profiles for your network deployment configuration.

Complete the initial profile creation steps as described in "Creating Process Server deployment manager profiles using the Profile Management Tool" on page 172.

Use the typical option if you want the Profile Management Tool to perform the following actions:
• Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
• Install the administrative console.
• Create a personal security certificate for the profile.

  The certificate has a personal key and private key, each with a default value of `WebAS` (you must change this password). The expiration period is one year.
• Create a root signing security certificate for signing other certificates.

  The certificate has a personal key and private key, each with a default value of `WebAS` (you must change this password). The expiration period is 15 years.

- Create a system service to run the server.

  Applicable only when your operating system and the privileges of your user account permit.

- Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.

1. On the Administrative Security page, enter values for the **User name** and **Password** and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

   Note: The **Next** button is disabled initially. There is no option to disable administrative security, thus you are required to specify the userid, password, and password confirmation in order to enable the **Next** button.

   Click **Next**.

2. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

   Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

   - **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
   - **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

You have created the Process Server deployment manager profile for the IBM Business Process Manager Standard network deployment configuration.

Build the clusters and configure the components on the clusters. The components available for configuration in a network deployment environment vary based on the IBM Business Process Manager Standard (Advanced or Standard) installed on your system.

You have options on how to build the clusters and configure the IBM Business Process Manager Standard components for your network deployment configuration.

Related tasks:

"Creating Process Server deployment manager profiles using the Profile Management Tool" on page 172
You can use the Profile Management Tool to create the Process Server deployment manager profiles for your network deployment configuration.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Using the advanced option to create Process Server deployment manager profiles with the Profile Management Tool:

Use the Profile Management Tool advanced option to create Process Server deployment manager profiles for your network deployment configuration.

Complete the initial profile creation steps as described in "Creating Process Server deployment manager profiles using the Profile Management Tool" on page 172.

Use the advanced option if you want to perform the following actions:
* Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
* Deploy the administrative console.
* Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets).
* Create a web server definition.
* Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

1. In the Optional Application Deployment page, select whether to deploy the administrative console.

   Click **Next** to display the Profile Name and Location page.

2. In the Profile Name and Location page, perform the following steps:

   a. In the **Profile name** field, specify a unique name or accept the default value.

      Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Profile commands in a multiprofile environment for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

   b. In the **Profile directory** field, enter the directory for the profile or use the **Browse. . .** button to go to the profile directory.

      The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform:

      * Linux   UNIX   `install_root/profiles/profile_name`

      * Windows   `install_root\profiles\profile_name`

      where *profile_name* is the name you specified. An error message is displayed if:

      * The *profile_name* you specify is not unique.
      * The directory you specify is not empty.
      * Your user ID does not have sufficient permissions for the directory.
      * There is not sufficient space to create the profile.

   c. Optional: Select the **Make this profile the default** check box if you wish to make the profile you are creating the default profile.

      When a profile is made to be the default profile, commands work automatically with it.

      Note: This check box appears only if you have an existing profile on your system.

      The first profile that you create on a workstation is the default profile.

      The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See Profile commands in a multiprofile environment for more information.

3. In the Node, Host and Cell Names page, perform the following actions for the profile you are creating:

   * In the **Node name** field, enter a name for the node or accept the default value.

Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.

- In the **Host name** field, enter a name for the host or accept the default value.
- In the **Cell name** field, enter a name for the cell or accept the default value.

Click **Next** to display the Administrative Security page.

4. On the Administrative Security page, enter values for the **User name** and **Password** and **Confirm password**. The password specified during install for administrator will be used for all internal users including tw_admin and tw_user.

   Note: The **Next** button is disabled initially. There is no option to disable administrative security, thus you are required to specify the userid, password, and password confirmation in order to enable the **Next** button.

   Click **Next**.

5. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

   Perform the following actions:

   - To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.
   - To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:
     - In the **Path** field, enter the directory path to the existing certificate.
     - In the **Password** field, enter the password for the certificate
     - In the **Keystore type** field, select the keystore type for the certificate you are importing.
     - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
     - Click **Next** to display the Security Certificate (Part 2) page

   When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

6. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

   If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

   When you create either or both certificates, or import either or both certificates, the keystore files that are created are:

   - `key.p12`: Contains the default personal certificate.
   - `trust.p12`: Contains the signer certificate from the default root certificate.
   - `root-key.p12`: Contains the root signing certificate.
   - `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
   - `deleted.p12`: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.

- ltpa.jceks: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

An imported certificate is added to the key.p12 file or the root-key.p12 file.

If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

7. From the Port Values Assignment page, verify that the ports specified for the profile are unique and click **Next**.

The Profile Management Tool detects ports currently used by other WebSphere products and displays recommended port values that do not conflict with existing ones. If you have applications other than WebSphere ones that use specified ports, verify that the ports do not conflict. If you chose not to deploy the administrative console on the Optional Application Deployment page, the administrative console ports are not available on the Port Values Assignment page.

Ports are recognized as being in use if the following conditions are satisfied:
- The ports are assigned to a profile created under an installation performed by the current user.
- The ports are currently in use.

Although the tool validates ports when you access the Port Values Assignment page, port conflicts can still occur resulting from selections you make on subsequent Profile Management Tool pages. Ports are not assigned until profile creation completes.

If you suspect a port conflict, you can investigate it after the profile is created. Determine the ports used during profile creation by examining the following file:

- [Linux] [UNIX] *profile_root*/**properties/portdef.props**

- [Windows] *profile_root***properties\portdef.props**

Included in this file are the keys and values used in setting the ports. If you discover port conflicts, you can reassign ports manually. To reassign ports, see the topic Updating ports in an existing profile in the WebSphere Application Server Network Deployment information center. Run the **updatePorts.ant** file through the **ws_ant** script detailed in this topic.

If you are installing	Next step
On a Linux or Windows platform, *and have root or Administrator group privileges*	The Linux or Windows Service Definition page is displayed. Proceed to step 8.

8. In the Service Definition page, indicate whether or not to use a Windows service or Linux service to run IBM Business Process Manager

[Windows] The Windows Service Definition page displays for the Windows platform only when the ID that installs the Windows service has the Administrator group privilege. If the profile is configured as a Windows service, the product starts Windows services for processes started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Windows service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

Important: If you choose to log on as a specified user account, you must specify the user ID and the password for the user who is to run the service, and the startup type (default is Manual). The user ID must not have spaces in its name, it must belong to the Administrator group, and it must have the advanced user right "Log on as a service." If the user ID belongs to the Administrator group, the Profile Management Tool grants it the advanced user right if it does not already have it.

During profile deletion, you can remove the Windows service that is added during profile creation.

IPv6 considerations when running profiles as Windows services

Profiles created to run as a Windows service fail to start when using IPv6 if the service is configured to run as Local System. Create a user-specific environment variable to enable IPv6.

Because this environment variable is a user variable instead of a Local System variable, only a Windows service that runs as that specific user can access this environment variable. By default, when a new profile is created and configured to run as a Windows service, the service is set to run as Local System. When the IBM Business Process Manager Windows service tries to run, the service is unable to access the user environment variable that specifies IPv6, and thus tries to start as IPv4. The server does not start correctly in this case. To resolve the problem, when creating the profile, specify that the IBM Business Process Manager Windows service runs as the same user ID under which the environment variable that specifies IPv6 is defined, instead of as Local System.

`Linux`   The Linux Service Definition page is displayed only if the current operating system is a supported version of Linux and the current user has the appropriate permissions.

IBM Business Process Manager attempts to start Linux services for processes that are started by the **startServer** or **startManager** commands. For example, if you configure a server or deployment manager as a Linux service and issue the **startServer** or **startManager** commands, the **wasservice** command starts the defined services.

By default, IBM Business Process Manager is not selected to run as a Linux service.

To create the service, the user who runs the Profile Management Tool must be the root user. If you run the Profile Management Tool with a non-root user ID, the Linux Service Definition page is not displayed, and no service is created.

You must specify a user name under which the service runs.

To delete a Linux service, the user must be the root user or have proper privileges for deleting the service. Otherwise, a removal script is created that the root user can run to delete the service on behalf of the user.

9. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

   Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

   - **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
   - **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

You have created the Process Server deployment manager profile for the IBM Business Process Manager Standard network deployment configuration.

Build the clusters and configure the components on the clusters. The components available for configuration in a network deployment environment vary based on the IBM Business Process Manager Standard (Advanced or Standard) installed on your system.

You have options on how to build the clusters and configure the IBM Business Process Manager Standard components for your network deployment configuration.

Related tasks:

"Creating Process Server deployment manager profiles using the Profile Management Tool" on page 172
You can use the Profile Management Tool to create the Process Server deployment manager profiles for your network deployment configuration.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Creating Process Server custom profiles (managed nodes) using the Profile Management Tool:

You can use the Profile Management Tool to create and configure Process Server custom profiles (managed nodes) for your network deployment configuration.

You have created the Process Server deployment manager profile and you have run the database configuration scripts to create Common database and tables.

To use the node (custom profile), you must federate the node to an existing deployment manager. If you want to federate the node during the creation process, the deployment manager must be running.

Tip: Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create profiles, as described in Configuring the software using command-line utilities and wsadmin.

Use the procedure in this task topic if you are:

- Creating a *Process Server* custom profile (managed nodes) for your IBM Business Process Manager Standard network deployment configuration

Note: Information that applies to process center only or to process server only, is so noted in this procedure.

This topic describes both the **Typical** and **Advanced** methods for creating custom profiles (managed nodes).

You can choose to federate the node (custom profile) to an existing deployment manager during the creation process, or federate the node later using the **addNode** command. If you decide to federate the node during the creation process, the tool sets the Common database configuration to the same database as the deployment manager. If you decide not to federate the node, the database configuration is left unconfigured.

1. Start the Profile Management Tool.

   Use one of the following commands:

   - ▓ Linux ▓ ▓ UNIX ▓ *install_root*/**bin/ProfileManagement/pmt.sh**

   - ▓ Windows ▓ *install_root***bin\ProfileManagement\pmt.bat**

   The Welcome page displays.

2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

   The **Profiles** tab displays.

   The **Profiles** tab can contain a list of profiles that have been created on your machine. You can use the Profile Management Tool to create new profiles or augment existing profiles.

3. In the **Profiles** tab, click **Create**.

   The Environment Selection page opens in a separate window.

4. On the Environment Selection page, locate the IBM Business Process Manager configuration (Advanced or Standard) for which you are creating the custom profile and expand the section. Select the option for Custom profile and click **Next**.

   Note: You can also create WebSphere Application Server profiles with this Profile Management Tool. However, this documentation addresses creating IBM Business Process Manager profiles only.

   The Profile Creation Options page displays.

5. On the Profile Creation Options page, decide whether to create the custom profile using the **Typical profile creation** or **Advanced profile creation** option.

Table 64. Selecting the profile creation option for your custom profile

Select	When you want to . . .
Typical profile creation	Let the Profile Management Tool
	• Assign default values to ports, to the location of the profile, and to the names of the profile, node, host, and cell.
	• Install the administrative console.
	• Create a personal security certificate for the profile.
	The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is one year.
	• Create a root signing security certificate for signing other certificates.
	The certificate has a personal key and private key, each with a default value of WebAS (you must change this password). The expiration period is 15 years.
	• Create a system service to run the server.
	Applicable only when your operating system and the privileges of your user account permit.
	• Select any of the supported database products and the database configuration is set for , the Process Server database and the Performance Data Warehouse database.
Advanced profile creation	• Assign customized values to ports, to the location of the profile, and to the names of the profile, node, host, and cell (when applicable).
	• Deploy the administrative console.
	• Deploy the default application (which contains the Snoop, Hello, and HitCount Servlets).
	• Create a web server definition.
	• Create a system service to run the server, if your operating system and the privileges of your user account permit the creation of services.

   If you selected **Typical profile creation**, go to step 6.

   If you selected **Advanced profile creation** , go to step 9 on page 182.

6. In the **Federation** page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

   Important:

   Check **Federate this node later** if any one of the following situations is true:
   • You plan to use this custom node as a migration target.
   • Another profile is being federated. (Node federation must be serialized.)
   • The deployment manager is not running or you are not sure if it is running.

- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Processing associated with federating the node as part of custom profile creation:
- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

Click **Next** to continue.
7. On the Database Configuration page, select the database used by the deployment manager and specify the location of the JDBC driver classpath files.
8. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:
- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.
You have completed the steps to create the profile with default configuration settings.

The remaining steps in this topic are for the **Advanced profile creation**.
9. In the Profile Name and Location page, perform the following steps:
   a. In the **Profile name** field, specify a unique name or accept the default value.

   Each profile that you create must have a name. When you have more than one profile, you can tell them apart at their highest level by this name. If you elect not to use the default name, see Profile commands in a multiprofile environment for information about issues you must consider when naming the profile, such as restrictions on the length of the directory name.

   b. In the **Profile directory** field, enter the directory for the profile or use the **Browse. . .** button to go to the profile directory.

   The directory you specify will contain the files that define the runtime environment, such as commands, configuration files, and log files. The default directory is dependent on platform:

- *install_root*/profiles/*profile_name*
- *install_root*\profiles*profile_name*

where *profile_name* is the name you specified. An error message is displayed if:

- The *profile_name* you specify is not unique.
- The directory you specify is not empty.
- Your user ID does not have sufficient permissions for the directory.
- There is not sufficient space to create the profile.

c. Optional: Select the **Make this profile the default** check box if you wish to make the profile you are creating the default profile.

When a profile is made to be the default profile, commands work automatically with it.

Note: This check box appears only if you have an existing profile on your system.

The first profile that you create on a workstation is the default profile.

The default profile is the default target for commands that are issued from the `bin` directory in the product installation root. When only one profile exists on a workstation, every command operates on that profile. If more than one profile exists, certain commands require that you specify the profile to which the command applies. See Profile commands in a multiprofile environment for more information.

d. Click **Next**.

Note: If you click **Back** and change the name of the profile, you might have to manually change the name on this page when it is displayed again.

10. In the Node and Host Names page, perform the following actions for the profile you are creating:

- In the **Node name** field, enter a name for the node or accept the default value.

    Try to keep the node name as short as possible, but ensure that node names are unique within your deployment environment. See *Naming considerations for profiles, nodes, servers, hosts, and cells* for information about reserved terms and other issues you must consider when naming.

- In the **Host name** field, enter a name for the host or accept the default value.

Click **Next** to display the Federation page.

11. In the **Federation** page, choose to federate the node into the deployment manager now as part of the profile creation, or at a later time and apart from profile creation. If you choose to federate the node as part of the profile creation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if to be used to authenticate with the deployment manager.

Important:

Check **Federate this node later** if any one of the following situations is true:

- You plan to use this custom node as a migration target.
- Another profile is being federated. (Node federation must be serialized.)
- The deployment manager is not running or you are not sure if it is running.
- The deployment manager has the SOAP connector disabled
- The deployment manager has not yet been augmented into a IBM Business Process Manager deployment manager.
- The deployment manager is not at a release level the same or higher than the release level of the profile you are creating.
- The deployment manager does not have a JMX administrative port enabled.

- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration** > **Deployment manager** > **Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

    Processing associated with federating the node as part of custom profile creation:
- The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).
- If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

    Click **Next** to continue.

12. In the Security Certificate (Part 1) page, specify whether to create new certificates or import existing certificates.

    Perform the following actions:
- To create a new default personal certificate and a new root signing certificate, select the **Create a new default personal certificate** and the **Create a new root signing certificate** radio buttons then click **Next**.
- To import an existing certificates, select the **Import an existing default personal certificate** and the **Import an existing root signing personal certificate** radio buttons and provide the following information:
    - In the **Path** field, enter the directory path to the existing certificate.
    - In the **Password** field, enter the password for the certificate
    - In the **Keystore type** field, select the keystore type for the certificate you are importing.
    - In the **Keystore alias** field, select the keystore alias for the certificate you are importing.
    - Click **Next** to display the Security Certificate (Part 2) page

    When you import a personal certificate as the default personal certificate, import the root certificate that signed the personal certificate. Otherwise, the Profile Management Tool adds the signer of the personal certificate to the `trust.p12` file.

13. In the Security Certificate (Part 2) page, verify that the certificate information is correct, and click **Next** to display the Port Values Assignment page.

    If you create the certificates, you can use the default values or modify them to create new certificates. The default personal certificate is valid for one year by default and is signed by the root signing certificate. The root signing certificate is a self-signed certificate that is valid for 15 years by default. The default keystore password for the root signing certificate is `WebAS`. Change the password. The password cannot contain any double-byte character set (DBCS) characters because certain keystore types, including PKCS12, do not support these characters. The keystore types that are supported depend on the providers in the `java.security` file.

    When you create either or both certificates, or import either or both certificates, the keystore files that are created are:
- `key.p12`: Contains the default personal certificate.
- `trust.p12`: Contains the signer certificate from the default root certificate.
- `root-key.p12`: Contains the root signing certificate.
- `default-signers.p12`: Contains signer certificates that are added to any new keystore file that you create after the server is installed and running. By default, the default root certificate signer and a DataPower signer certificate are in this keystore file.
- `deleted.p12`: Holds certificates deleted with the deleteKeyStore task so that they can be recovered if needed.

- `ltpa.jceks`: Contains server default Lightweight Third-Party Authentication (LTPA) keys that the servers in your environment use to communicate with each other.

These files all have the same password when you create or import the certificates, which is either the default password, or a password that you specify.

An imported certificate is added to the `key.p12` file or the `root-key.p12` file.

If you import any certificates and the certificates do not contain the information that you want, click **Back** to import another certificate.

14. In the Database Configuration page, select a database product and the location of the JDBC drivers.
15. In the Profile Summary page, click **Create** to create the profile or **Back** to change the characteristics of the profile.

The progress of the configuration displays on the Profile Configuration Progress window.

When the profile creation is complete, the Profile Complete page is displayed with the message **The Profile Management tool created the profile successfully**.

Attention: If errors are detected during profile creation, other messages might appear in place of the success message, for example:

- **The Profile Management tool created the profile but errors occurred**, which indicates that profile creation completed but errors were generated.
- **The Profile Management tool cannot create the profile**, which indicates that profile creation failed completely.

The Profile Complete page identifies the log file to reference in order to troubleshoot the problems. To proceed to the First Steps Console, make sure the **Launch First Steps Console checkbox** checked and click **Finish**. Also, close the Profiles page, which is open in a separate window. Use the First steps console to access the product documentation.

16. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.

You have created your custom profile.

The node within the profile is empty until you federate the node to a deployment manager, then use the administrative console to customize it.

In a network deployment environment, you must create and configure databases, create other custom profiles and federate them to your deployment manager, create servers, create clusters if you want workload management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

Federating custom nodes to a deployment manager:

You can use the **addNode** command to federate a custom node into a deployment manager cell. The following instructions guide you through the process of federating and deploying custom nodes.

Before using this procedure, ensure that the following prerequisites are met:

- You have installed IBM Business Process Manager and created an IBM Business Process Manager or WebSphere ESB deployment manager and a custom profile. This procedure assumes you did *not* federate the custom profile during its creation or augmentation, either with the Profile Management Tool or with the **manageprofiles** command-line utility.
- The deployment manager is running. If it is not, start it either by selecting **Start the deployment manager** from its First steps console or by entering the following command, where *profile_root* represents the installation location of the deployment manager profile:

– `Linux` `UNIX` *profile_root*/bin/startManager.sh

– `Windows` *profile_root*\bin\startManager.bat

- The deployment manager has been augmented into an IBM Business Process Manager or WebSphere ESB deployment manager.
- The deployment manager is at the same release level or higher than the custom profile you created or augmented.
- The deployment manager has a JMX administrative port enabled. The default protocol is SOAP.
- You do not plan to use this custom node as a migration target.

Federate a custom node so that it can be managed by a deployment manager. Use the **addNode** command to federate a custom profile into a deployment manager cell.

1. Go to the `bin` directory of the custom profile you want to federate. Open a command window and go to one of the following directories (from a command line), depending on platform (where *profile_root* represents the installation location of the custom profile):

   - `Linux` `UNIX` *profile_root*/bin

   - `Windows` *profile_root*\bin

2. Issue the **addNode** command.

   Issue one of the following commands from the command line if security is not enabled:

   - `Linux` `UNIX` ./addNode.sh *deployment_manager_host deployment_manager_SOAP_port*

   - `Windows` addNode.bat *deployment_manager_host deployment_manager_SOAP_port*

   Issue one of the following commands from the command line if security is enabled:

   - `Linux` `UNIX` ./addNode.sh *deployment_manager_host deployment_manager_SOAP_port* -username *userID_for_authentication* -password *password_for_authentication*

   - `Windows` addNode.bat *deployment_manager_host deployment_manager_SOAP_port* -username *userID_for_authentication* -password *password_for_authentication*

   An output window opens. If you see a message similar to the following message, your custom profile was federated successfully:

   ADMU0003I: Node DMNDID2Node03 has been successfully federated.

The custom profile is federated into the deployment manager. For more information about the **addNode** command and its parameters, see the topic Using wsadmin scripting to run the addNode command in the WebSphere Application Server Network Deployment information center.

After federating the custom profile, go to the administrative console of the deployment manager to customize the empty node or to create a new server.

Deleting profiles using the manageprofiles command-line utility:

You can delete a profile from the command line using the **manageprofiles** command-line utility.

1. Open a command prompt and run one of the following commands, based on your operating system:

   - `Linux` `UNIX` manageprofiles.sh -delete -profileName *profile_name*

   - `Windows` manageprofiles.bat -delete -profileName *profile_name*

   The variable *profile_name* represents the name of the profile that you want to delete.

2. Confirm that the profile deletion has completed by checking the following log file:

   - `Linux` `UNIX` *install_root*/logs/manageprofiles/*profile_name*_delete.log

   - `Windows` *install_root*\logs\manageprofiles*profile_name*_delete.log

If you plan to recreate a deleted profile using both the same profile name and the same database names that are associated with the deleted profile, you must manually delete the associated database names before you attempt to recreate the profile name and the database names.

Configuring the environment using manageprofiles and wsadmin

You can achieve the same configurations that you set up using the Profile Management Tool and deployment environment wizard by using the **manageprofiles** command-line utility and the **wsadmin** command.

Related concepts:

"Creating stand-alone profiles after a custom installation" on page 137
After you perform a Custom installation, you can use the Profile Management Tool or the **manageprofiles** command-line utility to create a stand-alone server profile.

Creating profiles using the manageprofiles command-line utility:

You can create a profile from the command line using the **manageprofiles** command-line utility and a property file.

To find out more about the **manageprofiles** command-line utility, see manageprofiles command-line utility. Sample response files for the **manageprofiles** command-line utility are located in the *install_root*/BPM/samples/manageprofiles directory.

Before you run the **manageprofiles** command-line utility ensure that you have completed the following tasks:

- You have reviewed the full list of prerequisites for creating or augmenting a profile at Prerequisites for creating or augmenting profiles.
- You have reviewed example profile creation commands
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Security role required for this task: See Granting write permission of files and directories to nonroot users for profile creation.

To use the **manageprofiles** command-line utility to create a profile, perform the following steps.

1. Determine the kind of profile you want to create, which in turn determines the template to use for your new profile (using the **-templatePath** option).

    Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates. The following templates are available:
    - `default.procctr`: for an IBM Business Process Manager Process Center stand-alone profile, which defines a stand-alone server with capabilities and functionality specific to IBM BPM Standard or Express.
    - `dmgr.procctr`: for an IBM Business Process Manager Process Center deployment manager profile, which defines a deployment manager for IBM BPM Standard. A deployment manager provides one administrative interface to a logical group of servers on one or more workstations.
    - `managed.procctr`: for an IBM Business Process Manager Process Center custom profile for IBM BPM Standard, which, when federated to a deployment manager, defines a managed node. If you have decided that your solution requires a deployment environment, your runtime environment requires one or more managed nodes.

        A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node. Do not federate a node unless the deployment manager you are federating to is at a release level the same or higher than that of the custom profile you are creating.

- `default.procsvr`: for a Process Server stand-alone profile, which defines a stand-alone server with capabilities and functionality specific to IBM BPM Standard or Express.
- `dmgr.procsvr`: for an IBM Business Process Manager Process Server deployment manager profile, which defines a deployment manager for IBM BPM Standard. A deployment manager provides one administrative interface to a logical group of servers on one or more workstations.
- `managed.procsvr`: for an IBM Business Process Manager Process Server custom profile for IBM BPM Standard, which, when federated to a deployment manager, defines a managed node.

  If you have decided that your solution requires a network deployment configuration, your runtime environment requires one or more managed nodes. A custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node. Do not federate a node unless the deployment manager you are federating to is at a release level the same or higher than that of the custom profile you are creating.

2. Determine which parameters are required for your type of profile by reviewing the example profile creation commands in manageprofile examples.
3. Determine the values that you want to supply for the profile by reviewing the default values in the manageprofiles parameters topic to see if they are what you need for your profile.

   Note: If you create profiles in IBM Business Process Manager using the **manageprofiles** command-line utility without specifying the **samplesPassword** parameter, the **INSTCONFPARTIALSUCCESS** message is returned. This occurs when the following criteria are met:
   - You installed the samples during IBM Business Process Manager or WebSphere Application Server installation.
   - You use the **manageprofiles** command-line utility to create the profiles.
   - The **samplesPassword** parameter is not specified in the **manageprofiles** command-line utility.

4. Create a response file.

   Sample response files are available under the `install_root/BPM/samples/manageprofiles/` directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server) and customize the file for your environment.

   Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. See Creating a response file using the IBM Business Process Manager samples.

5. Run the **manageprofiles** command from the command line, using the **-response** parameter and the name of the response file that you created.

   - Linux UNIX **manageprofiles.sh -response** *myResponseFile*

   - Windows **manageprofiles.bat -response** *myResponseFile*

   The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

You can see that your profile creation completed successfully if you receive a `INSTCONFSUCCESS: Profile creation succeeded.` message, and you can check the following log file: Linux UNIX Windows

- Linux UNIX `install_root/logs/manageprofiles/profile_name_create.log`

- Windows `install_root\logs\manageprofiles\profile_name_create.log`

Run the Installation Verification Test (IVT) tool to verify that the profile was created successfully. To do this, run the following command:

- Linux UNIX `profile_root/bin/wbi_ivt.sh`

- Windows `profile_root\bin\wbi_ivt.bat`

Related concepts:

"JDBC drivers and locations" on page 58
The following tables list the supported JDBC drivers. The first table contains the names and locations of the JDBC drivers that are provided with the product. The second table contains the names of the JDBC drivers that are supported but not provided with the product.

Creating a response file using the IBM Business Process Manager samples:

IBM Business Process Manager is shipped with sample response files that you can use to create a response file that contains all the details of how you want to configure your system.

Install and configure WebSphere Application Server which is used as a base for your IBM Business Process Manager configuration.

The response file you create is based on one of the following eight sample response files:
- PCStd_StandAlone_MSSQLSERVER_Win.response

  You can create a response file to use when configuring IBM Business Process Manager Standard Process Center in a stand-alone configuration with a SQL Server database.
- PSStd_StandAlone_DB2_UNIVERSAL_Win.response

  You can create a response file to use when configuring IBM Business Process Manager Standard Process Server in a stand-alone configuration with a DB2 database.
- PCAdv_StandAlone_MSSQLSERVER_MICROSOFT_Win.response

  You can create a response file to use when configuring IBM Business Process Manager Standard Process Center in a stand-alone configuration with a SQL Server database.
- PSAdv_StandAlone_DB2_UNIVERSAL_Win.response

  You can create a response file to use when configuring IBM Business Process Manager Standard Process Server in a stand-alone configuration with a DB2 database.
- PCAdv_Dmgr_ORACLE_Unix.response

  You can create a response file to use when configuring IBM Business Process Manager Standard Process Center in a deployment manager configuration with an Oracle database.
- PSAdv_Dmgr_DB2_DATASERVER_Unix.response

  You can create a response file to use when configuring IBM Business Process Manager Standard Process Server in a deployment manager configuration with a DB2 Dataserver database.
- PCAdv_Custom_ORACLE_Unix.response

  You can create a response file to use when configuring IBM Business Process Manager Standard Process Center in a custom configuration with an Oracle database.
- PSAdv_Custom_DB2_DATASERVER_Unix.response

  You can create a response file to use when configuring IBM Business Process Manager Standard Process Server in a custom configuration with a DB2 Dataserver database.

1. From the `Install_root`\BPM\samples\manageprofiles directory, locate and copy the sample response file to your working directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server).

2. `UNIX`　`Linux` Use the **chmod** command to assign the appropriate permissions to the copy of the response file. For example:

   `chmod 644 myResponseFile`

3. Edit the parameters in the response file as appropriate to your configuration. Make sure the **templatePath** property path matches your specific installation directory.

   Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail.

4. Save the edited response file.

5. **z/OS only:** Increase the OMVS time limit to allow the product configuration script time to complete. In an MVS console enter the following command: **SETOMVS MAXCPUTIME=86400**.

6. **z/OS only:** Access the z/OS UNIX System Services command shell, then switch to the administrator user ID. For example: **su wsadmin**.

7. Run the `manageprofiles` command from the command line, using the `-response` parameter and the name of the response file that you created.

   - **Linux** **UNIX** **manageprofiles.sh -response** *myResponseFile*

   - **Windows** **manageprofiles.bat -response** *myResponseFile*

   The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

8. Start the server.

The following example shows a response file to create a stand-alone Process Center profile on Windows.

```
#*****************************************************************************
# Creates a stand-alone Process Center profile on a Windows platform.
# Call manageprofiles -response PCStd_StandAlone__DB2_UNIVERSAL_Win.response
#*****************************************************************************
create
templatePath=C:/Program Files/IBM/WebSphere/AppServer/profileTemplates/BPM/default.procctr
profileName=ProcCtr01
cellName=SampleCell
nodeName=SampleNode01
serverName=server1
hostName=SampleHost.com
winserviceCheck=false
enableAdminSecurity=true
adminUserName=tw_admin
adminPassword=tw_admin
applyPerfTuningSetting=standard
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbType=DB2_UNIVERSAL
dbUserId=bpmadmin
dbPassword=bpmadmin1
dbHostName=DB_hostname
dbServerPort=50000
dbCreateNew=true
dbDelayConfig=false
procSvrDbName=PCBPMDB<
dbProcSvrUserId=procadmin
dbProcSvrPassword=procpwd
perfDWDbName=PCPERFDB
dbPerfDWUserId=perfadmin
dbPerfDWPassword=perfpwd
samplesPassword=smppwd
```

The following example shows a response file to create a stand-alone Process Server profile on Windows.

```
#*****************************************************************************
# Creates a stand-alone Process Server profile on a Windows platform.
# Call manageprofiles -response PSStd_StandAlone_DB2_UNIVERSAL_Win.response
#*****************************************************************************

create
templatePath=C:/Program Files/IBM/WebSphere/AppServer/profileTemplates/BPM/default.procsvr
profileName=ProcSrv01
cellName=SampleCell
nodeName=SampleNode01
serverName=server1
hostName=SampleHost.com
```

```
environmentName=SampleEnv
winserviceCheck=false
enableAdminSecurity=true
adminUserName=tw_admin
adminPassword=tw_admin
processCenterURL=http://processcenter_hostname:9080
processCenterUserId=tw_admin
processCenterPassword=tw_admin
applyPerfTuningSetting=standard
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbType=DB2_UNIVERSAL
dbUserId=bpmadmin
dbPassword=bpmadmin1
dbHostName=DB_hostname
dbServerPort=50000
dbCreateNew=true
dbDelayConfig=false
procSvrDbName=PSBPMDB
perfDWDbName=PSPERFDB
samplesPassword=smppwd
```

Creating stand-alone profiles using manageprofiles after a Custom installation:

After performing a Custom installation you can create stand-alone server profiles using the Profile Management Tool or the manageprofiles command-line utility.

The information in this section describes how to use the manageprofiles command-line utility to create stand-alone profiles. It assumes that you have run the installer and have performed a Custom installation.

You can also use the Profile Management Tool to create stand-alone profiles after performing a Custom installation.

Creating Process Center stand-alone profiles using manageprofiles:

Use **manageprofiles** command-line utility to create Process Center stand-alone profiles for IBM Business Process Manager Standard .

Before you run the **manageprofiles** command-line utility ensure that you have completed the following tasks:
- You have reviewed the full list of prerequisites for creating or augmenting a profile
- You have reviewed example profile creation commands
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

This task describes how to use the **manageprofiles** command-line utility to create Process Center profiles for IBM Business Process Manager Standard.

To use the **manageprofiles** command-line utility to create a profile, perform the following steps.
1. Determine the kind of profile you want to create, which in turn determines the template to use for your new profile (using the **-templatePath** option).

   Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates.

   The following templates are available:
   - default.procctr: for Process Center stand-alone profiles, which define stand-alone servers with capabilities and functionality specific to IBM BPM Standard and Express configurations.
2. Determine which parameters are required for your type of profile by reviewing the example profile creation commands in manageprofile examples.

3. Determine the values that you want to supply for the profile by reviewing the default values in the manageprofiles parameters topic to see if they are what you need for your profile.

4. Create a response file.

   Sample response files are available under the `install_root/BPM/samples/manageprofiles/` directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server) and customize the file for your environment.

   Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. See Creating a response file using the IBM Business Process Manager samples.

5. Run the **manageprofiles** command from the command line, using the **-response** parameter and the name of the response file that you created.

   - Linux UNIX **manageprofiles.sh -response** *myResponseFile*

   - Windows **manageprofiles.bat -response** *myResponseFile*

   The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

Related concepts:

"Prerequisites for creating or augmenting profiles" on page 131
Before creating or augmenting a profile, you must ensure that a series of prerequisites have been met.

"**manageprofiles** examples" on page 502
The examples in this section show how to create stand-alone, deployment manager, and custom (managed node) profiles using the **manageprofiles** command-line utility.

Response file for a Standard Process Center:

Use the **manageprofiles** command-line utility and the response file to create a stand-alone profile.

Response file for creating a Standard Process Center stand-alone profile for DB2:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a DB2 database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# *************************************************************************
# END COPYRIGHT

#############################################################################
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
```

```
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
################################################################################

################################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
################################################################################
create

################################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/default.procctr

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
################################################################################
profileName=ProcCtr01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#     Set a performance tuning level appropriate for the profile you are
#     creating.
#     This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#     development
#     standard
#     production
# Default Values:
#     None
################################################################################
#applyPerfTuningSetting=standard

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
```

```
#     Varies
#
# Parameter:   serverName
#
# Description:
#       Specifies the name of the server.
##############################################################################
cellName=cell_name
nodeName=node_name
serverName=server1

##############################################################################
# Parameter:   enableAdminSecurity
#
# Description:
#       For IBM Business Process Manager omit this parameter. Administrative
#       security is always enabled for IBM Business Process Manager profiles.
#       You must also specify the parameters -adminUserName and -adminPassword
#       along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
##############################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

##############################################################################
# Parameter:   signingCertDN
#
# Description:
#       Specifies the distinguished name of the root signing certificate that you
#       create when you create the profile. Specify the distinguished name in
#       quotation marks. This default personal certificate is located in the
#       server keystore file. If you do not specifically create or import a root
#       signing certificate, one is created by default. See the
#       -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:   signingCertValidityPeriod
#
# Description:
#       An optional parameter that specifies the amount of time in years that the
#       root signing certificate is valid. If you do not specify this parameter
#       with the -signingCertDN parameter, the root signing certificate is valid
#       for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:   keyStorePassword
#
# Description:
#       Specifies the password to use on all keystore files created during profile
#       creation. Keystore files are created for the default personal certificate
#       and the root signing certificate.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
```

```
###########################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#      Specify the password for the specified user or the local account that is
#      to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#      The value can be either true or false. Specify true to create a Windows
#      service for the server process that is created within the profile. Specify
#      false to not create the Windows service. The default value for this
#      parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#      The following values for Windows service startup can be used:
#      manual
#      automatic
#      disabled
#      The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
###############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


###############################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#    None
##########################################################################
#bpmdbDesign=design_file

##########################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
##########################################################################
dbCreateNew=true

##########################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    DB2_DATASERVER
# Default Values:
#    DB2_DATASERVER
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbHostName
#
# Description:
#     The database server host name or IP address. The default value is the
#     local host name. Required when the -bpmdbDesign parameter was not
#     specified. (It is not required if the -bpmdbDesign parameter is
#     specified.)
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
```

```
# Parameter:  dbServerPort
#
# Description:
#     The database server port number. Depending on the database you are using,
#     you can specify a different port number instead of the default port
#     number.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

################################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
################################################################################
dbDelayConfig=false

################################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    CMNDB
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
```

```
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd


###########################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password


###########################################################################
# Parameter:  perfDWDbName
#
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
```

```
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
##############################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


##############################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#     Process Server Messaging Engine user name. Required for setting up the
#     Process Server Messaging Engine. If you enter a user-specified a user name
#     and password during profile creation and if you specified ORACLE for
#     dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password


##############################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#     Performance Data Warehouse Messaging Engine user name. Required for
#     setting up the Performance Data Warehouse Messaging Engine.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPerfDWMePassword
#
# Description:
```

202 Installing

```
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
###############################################################################
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password


###############################################################################
# Parameter:  cdbSchemaName
#
# Description:
#      The schema name for the Common database. If no value is specified, the
#      following default database schema names are used:
#      DB2 database
#      The default schema name is the value specified by the dbUserId parameter.
#      The -cdbSchemaName parameter is valid for all database types except for
#      Oracle and Microsoft SQL Server.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  dbCommonUserId
#
# Description:
#      The user id to authenticate with the Common database.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  dbCommonPassword
#
# Description:
#      The password for the user id of the dbCommonUserId parameter.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
###############################################################################
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd


###############################################################################
# Parameter:  environmentName
#
# Description:
#      Specifies the name of the environment you are configuring. This parameter
#      has no default value, and it must not be empty.
#
# Valid Values:
#      Varies
# Default Values:
#      None
```

```
###############################################################################
environmentName=environment_name

###############################################################################
# Parameter:  environmentType
#
# Description:
#      Specifies the environment type. The environment type refers to how IBM
#      Business Process Manager is used; for example, in a production, stage or
#      test environment. Load testing might be done on a test server, while a
#      stage environment type might be used as a temporary location to host
#      changes before putting those changes into production. You might specify
#      Stage as the Environment type if the server you are configuring will be
#      accessed and used to review content and new functionality.  Valid values
#      are as follows:
#      Test
#      Use Test if the server you are configuring is to be used as a testing
#      environment.
#      Stage
#      Use Stage if the server is to serve as a staging platform to be used as a
#      preproduction server.
#      Production
#      Use Production if the server is to serve in a production capacity.
#      The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
###############################################################################
environmentType=Test

###############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#      Specifies whether the server is intended for development purposes only.
#      This parameter is useful when creating profiles to test applications on a
#      non-production server prior to deploying the applications on their
#      production application servers.  If -isDeveloperServer is set when
#      creating an IBM Business Process Manager profile, then a preconfigured VMM
#      file repository is installed. This file repository contains a sample
#      organization that can be used to test Business Process Choreographer
#      people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
###############################################################################
#isDeveloperServer=false

###############################################################################
# Parameter:  samplesPassword
#
# Description:
#      Creates a password to be used for samples. The password is used to
#      restrict access to Web application samples installed during the
#      installation of the application server.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#    None
###########################################################################
samplesPassword=samplesPassword
```

Response file for creating a Standard Process Center stand-alone profile for DB2 for z/OS:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a DB2 for z/OS database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# *************************************************************************
# END COPYRIGHT

###########################################################################
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###########################################################################

###########################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###########################################################################
create

###########################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
```

```
#      When using the -templatePath parameter, specify the fully qualified file
#      path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/default.procctr

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=ProcCtr01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#      Set a performance tuning level appropriate for the profile you are
#      creating.
#      This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#    development
#    standard
#    production
# Default Values:
#     None
################################################################################
#applyPerfTuningSetting=standard

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
```

```
#          shortHostNameCellCellNumber
#       else
#          shortHostNameNodeNodeNumberCell
#       where CellNumber is a sequential number starting at 01 and NodeNumber is
#       the node number that you used to define the node name.
#       The value for this parameter must not contain spaces or any characters
#       that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
################################################################################
cellName=cell_name
nodeName=node_name
serverName=server1


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
```

```
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


###############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
```

```
#       assignments to avoid port conflicts.
#       Do not use this parameter with the -defaultPorts or -portsFile parameters.
#       During profile creation, the manageprofiles command-line utility uses an
#       automatically generated set of recommended ports if you do not specify the
#       -startingPort parameter, the -defaultPorts parameter or the -portsFile
#       parameter. The recommended port values can be different than the default
#       port values based on the availability of the default ports.
#
# Valid Values:
#       Varies
# Default Values:
#       None
#
# Parameter:  portsFile
#
# Description:
#       An optional parameter that specifies the path to a file that defines port
#       settings for the new profile.
#       Do not use this parameter when using the -startingPort or -defaultPorts
#       parameter.
#       During profile creation, the manageprofiles command-line utility uses an
#       automatically generated set of recommended ports if you do not specify the
#       -startingPort parameter, the -defaultPorts parameter or the -portsFile
#       parameter. The recommended port values can be different than the default
#       port values based on the availability of the default ports.
#
# Valid Values:
#       Varies
# Default Values:
#       None
################################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

################################################################################
# Parameter:  enableService
#
# Description:
#       Enables the creation of a Linux service. Valid values include true or
#       false. The default value for this parameter is false.  When the
#       manageprofiles command-line utility is run with the -enableService option
#       set to true , the Linux service is created with the profile when the
#       command is run by the root user. When a nonroot user runs the
#       manageprofiles command-line utility, the profile is created, but the Linux
#       service is not. The Linux service is not created because the nonroot user
#       does not have sufficient permission to set up the service. An
#       INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#       creation and the profile creation log
#       install_root/logs/manageprofiles/profile_name_create.log contains a
#       message indicating the current user does not have sufficient permission to
#       set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#       false
#       true
# Default Values:
#       None
#
# Parameter:  serviceUserName
#
# Description:
#       Specifies the user ID that is used during the creation of the Linux
#       service so that the Linux service will run under this user ID. The Linux
#       service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
```

```
#       The following values for Windows service startup can be used:
#       manual
#       automatic
#       disabled
#       The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
###############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


###############################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#bpmdbDesign=design_file


###############################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
###############################################################################
dbCreateNew=false


###############################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
```

```
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      DB2UDBOS390
# Default Values:
#      DB2UDBOS390
################################################################################
dbType=DB2UDBOS390

################################################################################
# Parameter:  dbDelayConfig
#
# Description:
#      Indicates whether to postpone table creation until after the profile is
#      created. Valid values are true or false. The parameter is set to false by
#      default. Setting this parameter to true delays the execution of the
#      scripts that create the database tables. No. Specify false if you want the
#      database to be set up during profile creation.
#
# Valid Values:
#      false
#      true
# Default Values:
#      false
################################################################################
dbDelayConfig=true

################################################################################
# Parameter:  dbName
#
# Description:
#      The name of the database. By default, the value is set to orcl for Oracle
#      databases and to CMNDB for all other supported databases.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      CMNDB
#
# Parameter:  dbUserId
#
# Description:
#      The user ID for all database types. Specifies the user ID that has
#      privileges to create and drop the databases. The WebSphere data source
#      uses this ID to authenticate the database connection.
#      Important: The -dbUserId parameter value must be a valid database
#      authorization ID.  Required if the -bpmdbDesign file is not set.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  dbPassword
#
# Description:
#      The password required for database authentication. Required when the
#      -bpmdbDesign file is not set.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
```

```
# Default Values:
#    None
##########################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

##########################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##########################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

##########################################################################
# Parameter:  perfDWDbName
#
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
```

```
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
################################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


################################################################################
# Parameter:  dbStorageGroup
#
# Description:
#     The storage group name for DB2 z/OS databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbConnectionLocation
#
# Description:
#     The location of DB2 for z/OS database.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location


################################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
environmentName=environment_name


################################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
```

```
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
################################################################################
environmentType=Test

################################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
################################################################################
#isDeveloperServer=false

################################################################################
# Parameter:  samplesPassword
#
# Description:
#     Creates a password to be used for samples. The password is used to
#     restrict access to Web application samples installed during the
#     installation of the application server.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
samplesPassword=samplesPassword
```

Response file for creating a Standard Process Center stand-alone profile for Oracle:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for an Oracle database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
```

```
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# *************************************************************************
# END COPYRIGHT

###########################################################################
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###########################################################################

###########################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###########################################################################
create

###########################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
templatePath=BPM/default.procctr

###########################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
```

```
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=ProcCtr01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#     Set a performance tuning level appropriate for the profile you are
#     creating.
#     This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#     development
#     standard
#     production
# Default Values:
#     None
################################################################################
#applyPerfTuningSetting=standard

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
```

```
#      This parameter is required for profile creation only with the
#      dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#      procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#      The default value for this parameter is based on the short host name,
#      profile type, and a trailing number, for example:
#      if (DMgr)
#               shortHostNameCellManagerNodeNumber
#      else
#               shortHostNameNodeNodeNumber
#      where NodeNumber is a sequential number starting at 01.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  serverName
#
# Description:
#      Specifies the name of the server.
################################################################################
cellName=cell_name
nodeName=node_name
serverName=server1


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#      Specifies the distinguished name of the root signing certificate that you
#      create when you create the profile. Specify the distinguished name in
#      quotation marks. This default personal certificate is located in the
#      server keystore file. If you do not specifically create or import a root
#      signing certificate, one is created by default. See the
#      -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#      An optional parameter that specifies the amount of time in years that the
#      root signing certificate is valid. If you do not specify this parameter
#      with the -signingCertDN parameter, the root signing certificate is valid
```

```
#      for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
```

```
#      parameter.
#      During profile creation, the manageprofiles command-line utility uses an
#      automatically generated set of recommended ports if you do not specify the
#      -startingPort parameter, the -defaultPorts parameter or the -portsFile
#      parameter. The recommended port values can be different than the default
#      port values based on the availability of the default ports.
#
# Valid Values:
#      Varies
# Default Values:
#      None
###############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


###############################################################################
# Parameter:  enableService
#
# Description:
#      Enables the creation of a Linux service. Valid values include true or
#      false. The default value for this parameter is false.  When the
#      manageprofiles command-line utility is run with the -enableService option
#      set to true , the Linux service is created with the profile when the
#      command is run by the root user. When a nonroot user runs the
#      manageprofiles command-line utility, the profile is created, but the Linux
#      service is not. The Linux service is not created because the nonroot user
#      does not have sufficient permission to set up the service. An
#      INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#      creation and the profile creation log
#      install_root/logs/manageprofiles/profile_name_create.log contains a
#      message indicating the current user does not have sufficient permission to
#      set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#      false
#      true
# Default Values:
#      None
#
# Parameter:  serviceUserName
#
# Description:
#      Specifies the user ID that is used during the creation of the Linux
#      service so that the Linux service will run under this user ID. The Linux
#      service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  winserviceAccountType
#
# Description:
#      The type of the owner account of the Windows service created for the
#      profile. Valid values include specifieduser or localsystem. The
#      localsystem value runs the Windows service under the local account of the
#      user who creates the profile. The default value for this parameter is
#      localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#      localsystem
#      specifieduser
# Default Values:
#      localsystem
#
```

```
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
```

```
#winserviceStartupType=manual

###############################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#bpmdbDesign=design_file

###############################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
###############################################################################
dbCreateNew=false

###############################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     ORACLE
# Default Values:
#     ORACLE
#
# Parameter:  dbDriverType
#
# Description:
#     The database driver type. For an Oracle database, specify ORACLE. For
#     databases other than Oracle, the value is automatically set based on the
#     server operating system. Server installations on all operating systems use
#     type 4.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#    None
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##############################################################################
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

##############################################################################
# Parameter:  dbLocation
#
# Description:
#     The directory of the ORACLE_HOME directory. This parameter is required
#     when the parameter dbDelayConfig is set to false. No, this parameter is
#     only valid if you are using Oracle databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##############################################################################
dbLocation=db_location

##############################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
##############################################################################
dbDelayConfig=false

##############################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    orcl
```

```
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd

#############################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName can be the same.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
```

```
# Valid Values:
#    Varies
# Default Values:
#    None
############################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password


############################################################################
# Parameter:  perfDWDbName
#
# Description:
#    Database name for Performance Data Warehouse.  The value specified for
#    -perfDWDbName and -procSvrDbName can be the same.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#    Performance Data Warehouse user name. Required for setting up the
#    Performance Data Warehouse.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#    Performance Data Warehouse password. Required for setting up the
#    Performance Data Warehouse.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
############################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


############################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#    Process Server Messaging Engine user name. Required for setting up the
#    Process Server Messaging Engine. If you enter a user-specified a user name
#    and password during profile creation and if you specified ORACLE for
#    dbType, this parameter is required.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
############################################################################
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password
```

```
##########################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#      Performance Data Warehouse Messaging Engine user name. Required for
#      setting up the Performance Data Warehouse Messaging Engine.  This
#      parameter is required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPerfDWMePassword
#
# Valid Values:
#     Varies
# Default Values:
#     None
##########################################################################
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password


##########################################################################
# Parameter:  dbCommonUserId
#
# Description:
#     The user id to authenticate with the Common database. For Oracle, the
#     default user name is the first three characters of the Oracle database
#     service name. For example: If the database service name is ORCL,
#     dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#     Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#     profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#     and no bpmdbDesign file was specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbCommonPassword
#
# Description:
#      The password for the user id of the dbCommonUserId parameter. Required for
#      all stand-alone IBM Business Process Manager profiles and advanced
#      deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#      to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##########################################################################
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd


##########################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
```

```
# Valid Values:
#    Varies
# Default Values:
#    None
###############################################################################
environmentName=environment_name

###############################################################################
# Parameter:  environmentType
#
# Description:
#    Specifies the environment type. The environment type refers to how IBM
#    Business Process Manager is used; for example, in a production, stage or
#    test environment. Load testing might be done on a test server, while a
#    stage environment type might be used as a temporary location to host
#    changes before putting those changes into production. You might specify
#    Stage as the Environment type if the server you are configuring will be
#    accessed and used to review content and new functionality.  Valid values
#    are as follows:
#    Test
#    Use Test if the server you are configuring is to be used as a testing
#    environment.
#    Stage
#    Use Stage if the server is to serve as a staging platform to be used as a
#    preproduction server.
#    Production
#    Use Production if the server is to serve in a production capacity.
#    The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
###############################################################################
environmentType=Test

###############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#    Specifies whether the server is intended for development purposes only.
#    This parameter is useful when creating profiles to test applications on a
#    non-production server prior to deploying the applications on their
#    production application servers.  If -isDeveloperServer is set when
#    creating an IBM Business Process Manager profile, then a preconfigured VMM
#    file repository is installed. This file repository contains a sample
#    organization that can be used to test Business Process Choreographer
#    people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
###############################################################################
#isDeveloperServer=false

###############################################################################
# Parameter:  samplesPassword
#
# Description:
#    Creates a password to be used for samples. The password is used to
#    restrict access to Web application samples installed during the
#    installation of the application server.
#
```

```
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
samplesPassword=samplesPassword
```

Response file for creating a Standard Process Center stand-alone profile for SQL:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a SQL database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

###########################################################################
# These response file has the applicable parameters for creating a
# Standalone PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###########################################################################

###########################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###########################################################################
create

###########################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
```

```
#      a valid template directory.
#
#      When using the -templatePath parameter, specify the fully qualified file
#      path for the parameter.
#
# Valid Values:
#      Varies
# Default Values:
#      None
###############################################################################
templatePath=BPM/default.procctr

###############################################################################
# Parameter:  profileName
#
# Description:
#      Specifies the name of the profile. Use a unique value when creating a
#      profile.
#      Each profile that shares the same set of product binaries must have a
#      unique name. The default profile name is based on the profile type and a
#      trailing number, for example:
#      profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#      Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#      unique profile name. The value for this parameter must not contain spaces
#      or characters that are not valid such as the following: *, ?, ", <, >, ,,
#      /, \, and |. The profile name that you choose must not be in use. On
#      Windows platforms: If the fully qualified path contains spaces, enclose
#      the value in quotation marks.  The default value is based on the
#      install_root directory, the profiles subdirectory, and the name of the
#      file. For example, the default for profile creation is:
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#      file in the install_root/properties directory.  The value for this
#      parameter must be a valid path for the target system and must not be
#      currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#      Varies
# Default Values:
#      None
###############################################################################
profileName=ProcCtr01

###############################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#      Set a performance tuning level appropriate for the profile you are
#      creating.
#      This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#      development
#      standard
#      production
# Default Values:
#      None
###############################################################################
#applyPerfTuningSetting=standard

###############################################################################
# Parameter:  cellName
#
# Description:
#      Specifies the cell name of the profile. Use a unique cell name for each
#      profile.
#      The default value for this parameter is based on a combination of the
```

```
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
################################################################################
cellName=cell_name
nodeName=node_name
serverName=server1

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

################################################################################
```

```
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


################################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
```

```
#        omitting those ports that are already in use. The system recognizes and
#        resolves ports that are currently in use and determines the port
#        assignments to avoid port conflicts.
#        Do not use this parameter with the -defaultPorts or -portsFile parameters.
#        During profile creation, the manageprofiles command-line utility uses an
#        automatically generated set of recommended ports if you do not specify the
#        -startingPort parameter, the -defaultPorts parameter or the -portsFile
#        parameter. The recommended port values can be different than the default
#        port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#        An optional parameter that specifies the path to a file that defines port
#        settings for the new profile.
#        Do not use this parameter when using the -startingPort or -defaultPorts
#        parameter.
#        During profile creation, the manageprofiles command-line utility uses an
#        automatically generated set of recommended ports if you do not specify the
#        -startingPort parameter, the -defaultPorts parameter or the -portsFile
#        parameter. The recommended port values can be different than the default
#        port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


################################################################################
# Parameter:  enableService
#
# Description:
#        Enables the creation of a Linux service. Valid values include true or
#        false. The default value for this parameter is false.  When the
#        manageprofiles command-line utility is run with the -enableService option
#        set to true , the Linux service is created with the profile when the
#        command is run by the root user. When a nonroot user runs the
#        manageprofiles command-line utility, the profile is created, but the Linux
#        service is not. The Linux service is not created because the nonroot user
#        does not have sufficient permission to set up the service. An
#        INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#        creation and the profile creation log
#        install_root/logs/manageprofiles/profile_name_create.log contains a
#        message indicating the current user does not have sufficient permission to
#        set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#        Specifies the user ID that is used during the creation of the Linux
#        service so that the Linux service will run under this user ID. The Linux
#        service runs whenever the user ID is logged on. Not required.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
```

```
#
# Description:
#      The following values for Windows service startup can be used:
#      manual
#      automatic
#      disabled
#      The default value for this parameter is manual.
#
# Valid Values:
#      manual
#      automatic
#      disabled
# Default Values:
#      manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  bpmdbDesign
#
# Description:
#      Specifies the file that holds the database configuration design for all of
#      the IBM Business Process Manager components .
#      This parameter replaces the -wbidbDesign that was used in previous
#      versions of WebSphere Process Server.
#
#      Note: If a bpmdbDesign file parameter is specified, none of the database
#      related properties should be specified.
#
# Valid Values:
#      Varies
# Default Values:
#      None
################################################################################
#bpmdbDesign=design_file


################################################################################
# Parameter:  dbCreateNew
#
# Description:
#      Indicates a new database is created or if an existing database is to be
#      reused. Valid values are true or false. This value cannot be true for an
#      Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#      a DB2 database.
#
# Valid Values:
#      false
# Default Values:
#      false
################################################################################
dbCreateNew=false


################################################################################
# Parameter:  dbType
#
# Description:
#      The database type. Set one of the following values for the type of
#      database product you are using with IBM Business Process Manager.
#
#      Required only for one of the following:
#      A stand-alone profile when -bpmdbDesign was not specified.
```

```
#      An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      MSSQLSERVER_MICROSOFT
# Default Values:
#      MSSQLSERVER_MICROSOFT
#
# Parameter:  dbJDBCClasspath
#
# Description:
#      The directory path location of JDBC driver files.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
################################################################################
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

################################################################################
# Parameter:  dbDelayConfig
#
# Description:
#      Indicates whether to postpone table creation until after the profile is
#      created. Valid values are true or false. The parameter is set to false by
#      default. Setting this parameter to true delays the execution of the
#      scripts that create the database tables. No. Specify false if you want the
#      database to be set up during profile creation.
#
# Valid Values:
#      false
#      true
# Default Values:
#      false
################################################################################
dbDelayConfig=false

################################################################################
# Parameter:  dbName
#
# Description:
#      The name of the database. By default, the value is set to orcl for Oracle
#      databases and to CMNDB for all other supported databases.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      CMNDB
#
# Parameter:  dbUserId
#
# Description:
#      The user ID for all database types. Specifies the user ID that has
#      privileges to create and drop the databases. The WebSphere data source
#      uses this ID to authenticate the database connection.
#      Important: The -dbUserId parameter value must be a valid database
#      authorization ID.  Required if the -bpmdbDesign file is not set.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
```

```
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd


###########################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password


###########################################################################
# Parameter:  perfDWDbName
#
```

```
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
##############################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


##############################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#     Process Server Messaging Engine user name. Required for setting up the
#     Process Server Messaging Engine. If you enter a user-specified a user name
#     and password during profile creation and if you specified ORACLE for
#     dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password


##############################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#     Performance Data Warehouse Messaging Engine user name. Required for
#     setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
```

```
# Parameter:  dbPerfDWMePassword
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password


###########################################################################
# Parameter:  dbCommonUserId
#
# Description:
#     The user id to authenticate with the Common database. For Oracle, the
#     default user name is the first three characters of the Oracle database
#     service name. For example: If the database service name is ORCL,
#     dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#     Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#     profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#     and no bpmdbDesign file was specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbCommonPassword
#
# Description:
#     The password for the user id of the dbCommonUserId parameter. Required for
#     all stand-alone IBM Business Process Manager profiles and advanced
#     deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#     to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd


###########################################################################
# Parameter:  dbWinAuth
#
# Description:
#     An optional parameter that is valid in all default profile templates.
#     Specify true to indicate that you are running SQL Server using integrated
#     windows authentication.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
###########################################################################
#dbWinAuth=true


###########################################################################
# Parameter:  environmentName
```

```
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
environmentName=environment_name

################################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
################################################################################
environmentType=Test

################################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
################################################################################
#isDeveloperServer=false

################################################################################
# Parameter:  samplesPassword
#
```

```
# Description:
#     Creates a password to be used for samples. The password is used to
#     restrict access to Web application samples installed during the
#     installation of the application server.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
samplesPassword=samplesPassword
```

Creating Process Server stand-alone profiles using manageprofiles:

Use the **manageprofiles** command-line utility to create Process Server stand-alone profiles for IBM Business Process Manager Standard .

Before you run the **manageprofiles** command-line utility ensure that you have completed the following tasks:
- You have reviewed the full list of prerequisites for creating or augmenting a profile at Prerequisites for creating or augmenting profiles
- You have reviewed example profile creation commands.
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

This task describes how to use the **manageprofiles** command-line utility to create Process Server stand-alone profiles for IBM Business Process Manager Standard.

To use the **manageprofiles** command-line utility to create a profile, perform the following steps.
1. Determine the kind of profile you want to create, which in turn determines the template to use for your new profile (using the **-templatePath** option).

    Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates.

    The following templates are available:
    - `default.procsvr`: for Process Server stand-alone profiles, which define stand-alone servers with capabilities and functionality specific to IBM BPM Standard and Express configurations.
2. Determine which parameters are required for your type of profile by reviewing the example profile creation commands in manageprofile examples.
3. Determine the values that you want to supply for the profile by reviewing the default values in the manageprofiles parameters topic to see if they are what you need for your profile.
4. Create a response file.

    Sample response files are available under the install_root/BPM/samples/manageprofiles/ directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server) and customize the file for your environment.

    Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. See Creating a response file using the IBM Business Process Manager samples.
5. Run the **manageprofiles** command from the command line, using the **-response** parameter and the name of the response file that you created.

    - `Linux` `UNIX` **manageprofiles.sh -response** *myResponseFile*

    - `Windows` **manageprofiles.bat -response** *myResponseFile*

The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

Related concepts:

"Creating stand-alone profiles after a custom installation" on page 137
After you perform a Custom installation, you can use the Profile Management Tool or the **manageprofiles** command-line utility to create a stand-alone server profile.

"Prerequisites for creating or augmenting profiles" on page 131
Before creating or augmenting a profile, you must ensure that a series of prerequisites have been met.

"**manageprofiles** examples" on page 502
The examples in this section show how to create stand-alone, deployment manager, and custom (managed node) profiles using the **manageprofiles** command-line utility.

Response file for a Standard Process Server:

Use the **manageprofiles** command-line utility and the response file to create a stand-alone profile.

Response file for creating a Standard Process Server stand-alone profile for DB2:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a DB2 database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

#############################################################################
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#############################################################################

#############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
```

```
#      If you are creating a profile, this parameter is required.
################################################################################
create

################################################################################
# Parameter:  templatePath
#
# Description:
#      Specifies the directory path to the template files in the installation
#      root directory. Within the profileTemplates directory are various
#      directories that correspond to different profile types and that vary with
#      the type of product installed. The profile directories are the paths that
#      you indicate while using the -templatePath option.
#      Use absolute paths. This parameter must exist as a directory and point to
#      a valid template directory.
#
#      When using the -templatePath parameter, specify the fully qualified file
#      path for the parameter.
#
# Valid Values:
#      Varies
# Default Values:
#      None
################################################################################
templatePath=BPM/default.procsvr

################################################################################
# Parameter:  profileName
#
# Description:
#      Specifies the name of the profile. Use a unique value when creating a
#      profile.
#      Each profile that shares the same set of product binaries must have a
#      unique name. The default profile name is based on the profile type and a
#      trailing number, for example:
#      profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#      Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#      unique profile name. The value for this parameter must not contain spaces
#      or characters that are not valid such as the following: *, ?, ", <, >, ,,
#      /, \, and |. The profile name that you choose must not be in use. On
#      Windows platforms: If the fully qualified path contains spaces, enclose
#      the value in quotation marks.  The default value is based on the
#      install_root directory, the profiles subdirectory, and the name of the
#      file. For example, the default for profile creation is:
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#      file in the install_root/properties directory.  The value for this
#      parameter must be a valid path for the target system and must not be
#      currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#      Varies
# Default Values:
#      None
################################################################################
profileName=ProcSrv01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#      Set a performance tuning level appropriate for the profile you are
#      creating.
#      This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#      development
```

```
#     standard
#     production
# Default Values:
#     None
##############################################################################
#applyPerfTuningSetting=standard

##############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
##############################################################################
cellName=cell_name
nodeName=node_name
serverName=server1

##############################################################################
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
```

```
#      to. If this parameter is set, the protocol must be either http or https.
#      Note: No corresponding useServerOffline parameter exists to use the server
#      offline.  To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  processCenterUserId
#
# Description:
#      Specifies the user ID to be used to connect to the Process Center. This
#      parameter is valid for stand-alone Process Server profiles. This parameter
#      is required if the Process Server is online (connected to a Process
#      Center). It is required for a stand-alone process server (standard or
#      advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  processCenterPassword
#
# Description:
#      Specifies the user password to be used to connect to the Process Center.
#      This parameter is valid for stand-alone Process Server profiles. This
#      parameter is required if the Process Server is online (connected to a
#      Process Center). It is required for a stand-alone process server (standard
#      or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#      Specifies the distinguished name of the root signing certificate that you
#      create when you create the profile. Specify the distinguished name in
#      quotation marks. This default personal certificate is located in the
#      server keystore file. If you do not specifically create or import a root
```

```
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


################################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
```

```
#       parameter. The recommended port values can be different than the default
#       port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#       An optional parameter that specifies the path to a file that defines port
#       settings for the new profile.
#       Do not use this parameter when using the -startingPort or -defaultPorts
#       parameter.
#       During profile creation, the manageprofiles command-line utility uses an
#       automatically generated set of recommended ports if you do not specify the
#       -startingPort parameter, the -defaultPorts parameter or the -portsFile
#       parameter. The recommended port values can be different than the default
#       port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

################################################################################
# Parameter:  enableService
#
# Description:
#       Enables the creation of a Linux service. Valid values include true or
#       false. The default value for this parameter is false.  When the
#       manageprofiles command-line utility is run with the -enableService option
#       set to true , the Linux service is created with the profile when the
#       command is run by the root user. When a nonroot user runs the
#       manageprofiles command-line utility, the profile is created, but the Linux
#       service is not. The Linux service is not created because the nonroot user
#       does not have sufficient permission to set up the service. An
#       INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#       creation and the profile creation log
#       install_root/logs/manageprofiles/profile_name_create.log contains a
#       message indicating the current user does not have sufficient permission to
#       set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#       Specifies the user ID that is used during the creation of the Linux
#       service so that the Linux service will run under this user ID. The Linux
#       service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
```

```
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
```

```
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#bpmdbDesign=design_file

################################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
################################################################################
dbCreateNew=true

################################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    DB2_DATASERVER
```

```
# Default Values:
#    DB2_DATASERVER
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbHostName
#
# Description:
#     The database server host name or IP address. The default value is the
#     local host name. Required when the -bpmdbDesign parameter was not
#     specified. (It is not required if the -bpmdbDesign parameter is
#     specified.)
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbServerPort
#
# Description:
#     The database server port number. Depending on the database you are using,
#     you can specify a different port number instead of the default port
#     number.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

################################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
################################################################################
dbDelayConfig=false
```

```
############################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    CMNDB
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
############################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

############################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password


################################################################################
# Parameter:  perfDWDbName
#
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
################################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


################################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#     Process Server Messaging Engine user name. Required for setting up the
#     Process Server Messaging Engine. If you enter a user-specified a user name
#     and password during profile creation and if you specified ORACLE for
#     dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
```

```
# Parameter:  dbProcSvrMePassword
#
# Description:
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password


###############################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#      Performance Data Warehouse Messaging Engine user name. Required for
#      setting up the Performance Data Warehouse Messaging Engine.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPerfDWMePassword
#
# Description:
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password


###############################################################################
# Parameter:  cdbSchemaName
#
# Description:
#     The schema name for the Common database. If no value is specified, the
#     following default database schema names are used:
#     DB2 database
#     The default schema name is the value specified by the dbUserId parameter.
#     The -cdbSchemaName parameter is valid for all database types except for
#     Oracle and Microsoft SQL Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbCommonUserId
#
# Description:
#     The user id to authenticate with the Common database.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#    None
#
# Parameter:  dbCommonPassword
#
# Description:
#     The password for the user id of the dbCommonUserId parameter.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd


################################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
environmentName=environment_name


################################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
################################################################################
environmentType=Test


################################################################################
# Parameter:  isDeveloperServer
#
```

```
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##########################################################################
#isDeveloperServer=false


##########################################################################
# Parameter:  samplesPassword
#
# Description:
#     Creates a password to be used for samples. The password is used to
#     restrict access to Web application samples installed during the
#     installation of the application server.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##########################################################################
samplesPassword=samplesPassword
```

Response file for creating a Standard Process Server stand-alone profile for DB2 for z/OS:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a DB2 for z/OS database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ************************************************************************
# END COPYRIGHT

##########################################################################
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_DB2zOS.response
```

```
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###############################################################################

###############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###############################################################################
create

###############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
templatePath=BPM/default.procsvr

###############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
```

```
profileName=ProcSrv01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#      Set a performance tuning level appropriate for the profile you are
#      creating.
#      This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#    development
#    standard
#    production
# Default Values:
#    None
################################################################################
#applyPerfTuningSetting=standard

################################################################################
# Parameter:  cellName
#
# Description:
#      Specifies the cell name of the profile. Use a unique cell name for each
#      profile.
#      The default value for this parameter is based on a combination of the
#      short host name, the constant Cell, and a trailing number, for example:
#      if (DMgr)
#          shortHostNameCellCellNumber
#      else
#          shortHostNameNodeNodeNumberCell
#      where CellNumber is a sequential number starting at 01 and NodeNumber is
#      the node number that you used to define the node name.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#      Specifies the node name for the node that is created with the new profile.
#      Use a unique value within the cell or on the workstation. Each profile
#      that shares the same set of product binaries must have a unique node name.
#      This parameter is required for profile creation only with the
#      dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#      procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#      The default value for this parameter is based on the short host name,
#      profile type, and a trailing number, for example:
#      if (DMgr)
#              shortHostNameCellManagerNodeNumber
#      else
#              shortHostNameNodeNodeNumber
#      where NodeNumber is a sequential number starting at 01.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  serverName
#
```

```
# Description:
#     Specifies the name of the server.
##########################################################################
cellName=cell_name
nodeName=node_name
serverName=server1


##########################################################################
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline.  To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##########################################################################
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password


##########################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
##########################################################################
```

```
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

##########################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##########################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

##########################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
```

```
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##########################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


##########################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
```

```
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
```

```
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
###############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

###############################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#bpmdbDesign=design_file

###############################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
###############################################################################
dbCreateNew=false

###############################################################################
# Parameter:  dbType
#
# Description:
```

```
#      The database type. Set one of the following values for the type of
#      database product you are using with IBM Business Process Manager.
#
#      Required only for one of the following:
#      A stand-alone profile when -bpmdbDesign was not specified.
#      An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      DB2UDBOS390
# Default Values:
#      DB2UDBOS390
############################################################################
dbType=DB2UDBOS390

############################################################################
# Parameter:  dbDelayConfig
#
# Description:
#      Indicates whether to postpone table creation until after the profile is
#      created. Valid values are true or false. The parameter is set to false by
#      default. Setting this parameter to true delays the execution of the
#      scripts that create the database tables. No. Specify false if you want the
#      database to be set up during profile creation.
#
# Valid Values:
#      false
#      true
# Default Values:
#      false
############################################################################
dbDelayConfig=true

############################################################################
# Parameter:  dbName
#
# Description:
#      The name of the database. By default, the value is set to orcl for Oracle
#      databases and to CMNDB for all other supported databases.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      CMNDB
#
# Parameter:  dbUserId
#
# Description:
#      The user ID for all database types. Specifies the user ID that has
#      privileges to create and drop the databases. The WebSphere data source
#      uses this ID to authenticate the database connection.
#      Important: The -dbUserId parameter value must be a valid database
#      authorization ID.  Required if the -bpmdbDesign file is not set.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  dbPassword
#
# Description:
```

```
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
############################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd


############################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
############################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password


############################################################################
# Parameter:  perfDWDbName
#
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
```

```
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
##############################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


##############################################################################
# Parameter:  dbStorageGroup
#
# Description:
#     The storage group name for DB2 z/OS databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbConnectionLocation
#
# Description:
#     The location of DB2 for z/OS database.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##############################################################################
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location


##############################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##############################################################################
environmentName=environment_name


##############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
```

```
#      stage environment type might be used as a temporary location to host
#      changes before putting those changes into production. You might specify
#      Stage as the Environment type if the server you are configuring will be
#      accessed and used to review content and new functionality.  Valid values
#      are as follows:
#      Test
#      Use Test if the server you are configuring is to be used as a testing
#      environment.
#      Stage
#      Use Stage if the server is to serve as a staging platform to be used as a
#      preproduction server.
#      Production
#      Use Production if the server is to serve in a production capacity.
#      The default value is Test.
#
# Valid Values:
#      Test
#      Production
#      Stage
# Default Values:
#      Test
###########################################################################
environmentType=Test

###########################################################################
# Parameter:  isDeveloperServer
#
# Description:
#      Specifies whether the server is intended for development purposes only.
#      This parameter is useful when creating profiles to test applications on a
#      non-production server prior to deploying the applications on their
#      production application servers.  If -isDeveloperServer is set when
#      creating an IBM Business Process Manager profile, then a preconfigured VMM
#      file repository is installed. This file repository contains a sample
#      organization that can be used to test Business Process Choreographer
#      people resolution, ready for you to use as is.
#
# Valid Values:
#      false
#      true
# Default Values:
#      false
###########################################################################
#isDeveloperServer=false

###########################################################################
# Parameter:  samplesPassword
#
# Description:
#      Creates a password to be used for samples. The password is used to
#      restrict access to Web application samples installed during the
#      installation of the application server.
#
# Valid Values:
#      Varies
# Default Values:
#      None
###########################################################################
samplesPassword=samplesPassword
```

Response file for creating a Standard Process Server stand-alone profile for Oracle:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for an Oracle database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

#############################################################################
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#############################################################################

#############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
#############################################################################
create

#############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
templatePath=BPM/default.procsvr

#############################################################################
```

```
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=ProcSrv01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#     Set a performance tuning level appropriate for the profile you are
#     creating.
#     This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#     development
#     standard
#     production
# Default Values:
#     None
################################################################################
#applyPerfTuningSetting=standard

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
```

```
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
##############################################################################
cellName=cell_name
nodeName=node_name
serverName=server1


##############################################################################
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline.  To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
```

```
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password


###############################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
###############################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


###############################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


###########################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
#defaultPorts=default_ports
```

```
#startingPort=startingPort
#portsFile=file_path

##############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
```

```
#      You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#bpmdbDesign=design_file

################################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
################################################################################
dbCreateNew=false

################################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     ORACLE
# Default Values:
#     ORACLE
#
# Parameter:  dbDriverType
#
# Description:
#     The database driver type. For an Oracle database, specify ORACLE. For
#     databases other than Oracle, the value is automatically set based on the
#     server operating system. Server installations on all operating systems use
#     type 4.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
```

```
#     None
##############################################################################
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle


##############################################################################
# Parameter:  dbLocation
#
# Description:
#     The directory of the ORACLE_HOME directory. This parameter is required
#     when the parameter dbDelayConfig is set to false. No, this parameter is
#     only valid if you are using Oracle databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
dbLocation=db_location


##############################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##############################################################################
dbDelayConfig=false


##############################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     orcl
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
```

```
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPassword
#
# Description:
#    The password required for database authentication. Required when the
#    -bpmdbDesign file is not set.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd


###########################################################################
# Parameter:  procSvrDbName
#
# Description:
#    Database name for Process Server database.  The value specified for
#    -procSvrDbName and -perfDWDbName can be the same.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#    The Process Server user name. Required for setting up the Process Server.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#    The Process Server password. Required for setting up the Process Server.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password


###########################################################################
# Parameter:  perfDWDbName
#
# Description:
```

```
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName can be the same.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
################################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


################################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#     Process Server Messaging Engine user name. Required for setting up the
#     Process Server Messaging Engine. If you enter a user-specified a user name
#     and password during profile creation and if you specified ORACLE for
#     dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password


################################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#     Performance Data Warehouse Messaging Engine user name. Required for
#     setting up the Performance Data Warehouse Messaging Engine.  This
#     parameter is required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
```

```
# Parameter:  dbPerfDWMePassword
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password


################################################################################
# Parameter:  dbCommonUserId
#
# Description:
#    The user id to authenticate with the Common database. For Oracle, the
#    default user name is the first three characters of the Oracle database
#    service name. For example: If the database service name is ORCL,
#    dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#    Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#    profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#    and no bpmdbDesign file was specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCommonPassword
#
# Description:
#    The password for the user id of the dbCommonUserId parameter. Required for
#    all stand-alone IBM Business Process Manager profiles and advanced
#    deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#    to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd


################################################################################
# Parameter:  environmentName
#
# Description:
#    Specifies the name of the environment you are configuring. This parameter
#    has no default value, and it must not be empty.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
environmentName=environment_name


################################################################################
# Parameter:  environmentType
#
# Description:
#    Specifies the environment type. The environment type refers to how IBM
#    Business Process Manager is used; for example, in a production, stage or
```

```
#      test environment. Load testing might be done on a test server, while a
#      stage environment type might be used as a temporary location to host
#      changes before putting those changes into production. You might specify
#      Stage as the Environment type if the server you are configuring will be
#      accessed and used to review content and new functionality.  Valid values
#      are as follows:
#      Test
#      Use Test if the server you are configuring is to be used as a testing
#      environment.
#      Stage
#      Use Stage if the server is to serve as a staging platform to be used as a
#      preproduction server.
#      Production
#      Use Production if the server is to serve in a production capacity.
#      The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
##############################################################################
environmentType=Test

##############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#      Specifies whether the server is intended for development purposes only.
#      This parameter is useful when creating profiles to test applications on a
#      non-production server prior to deploying the applications on their
#      production application servers.  If -isDeveloperServer is set when
#      creating an IBM Business Process Manager profile, then a preconfigured VMM
#      file repository is installed. This file repository contains a sample
#      organization that can be used to test Business Process Choreographer
#      people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##############################################################################
#isDeveloperServer=false

##############################################################################
# Parameter:  samplesPassword
#
# Description:
#      Creates a password to be used for samples. The password is used to
#      restrict access to Web application samples installed during the
#      installation of the application server.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
samplesPassword=samplesPassword
```

Response file for creating a Standard Process Server stand-alone profile for SQL:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a SQL database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

#############################################################################
# These response file has the applicable parameters for creating a
# Standalone PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#############################################################################

#############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
#############################################################################
create

#############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
templatePath=BPM/default.procsvr

#############################################################################
```

```
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
profileName=ProcSrv01

##############################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#     Set a performance tuning level appropriate for the profile you are
#     creating.
#     This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#     development
#     standard
#     production
# Default Values:
#     None
##############################################################################
#applyPerfTuningSetting=standard

##############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
```

```
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
##############################################################################
cellName=cell_name
nodeName=node_name
serverName=server1


##############################################################################
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline.  To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
```

```
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


#############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
#defaultPorts=default_ports
```

```
#startingPort=startingPort
#portsFile=file_path

##############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
```

```
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#bpmdbDesign=design_file

##############################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
##############################################################################
dbCreateNew=false

##############################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     MSSQLSERVER_MICROSOFT
# Default Values:
#     MSSQLSERVER_MICROSOFT
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

##############################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
```

```
# Valid Values:
#    false
#    true
# Default Values:
#    false
#############################################################################
dbDelayConfig=false

#############################################################################
# Parameter:  dbName
#
# Description:
#      The name of the database. By default, the value is set to orcl for Oracle
#      databases and to CMNDB for all other supported databases.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    CMNDB
#
# Parameter:  dbUserId
#
# Description:
#      The user ID for all database types. Specifies the user ID that has
#      privileges to create and drop the databases. The WebSphere data source
#      uses this ID to authenticate the database connection.
#      Important: The -dbUserId parameter value must be a valid database
#      authorization ID.  Required if the -bpmdbDesign file is not set.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPassword
#
# Description:
#      The password required for database authentication. Required when the
#      -bpmdbDesign file is not set.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#############################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

#############################################################################
# Parameter:  procSvrDbName
#
# Description:
#      Database name for Process Server database.  The value specified for
#      -procSvrDbName and -perfDWDbName must be different.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
```

```
#     None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#      The Process Server user name. Required for setting up the Process Server.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#      The Process Server password. Required for setting up the Process Server.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password


###############################################################################
# Parameter:  perfDWDbName
#
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
###############################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


###############################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#     Process Server Messaging Engine user name. Required for setting up the
#     Process Server Messaging Engine. If you enter a user-specified a user name
#     and password during profile creation and if you specified ORACLE for
#     dbType, this parameter is required.
```

```
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password


###############################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#     Performance Data Warehouse Messaging Engine user name. Required for
#     setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPerfDWMePassword
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password


###############################################################################
# Parameter:  dbCommonUserId
#
# Description:
#     The user id to authenticate with the Common database. For Oracle, the
#     default user name is the first three characters of the Oracle database
#     service name. For example: If the database service name is ORCL,
#     dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#     Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#     profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#     and no bpmdbDesign file was specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbCommonPassword
#
# Description:
#     The password for the user id of the dbCommonUserId parameter. Required for
#     all stand-alone IBM Business Process Manager profiles and advanced
```

```
#     deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#     to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd


###############################################################################
# Parameter:  dbWinAuth
#
# Description:
#     An optional parameter that is valid in all default profile templates.
#     Specify true to indicate that you are running SQL Server using integrated
#     windows authentication.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
###############################################################################
#dbWinAuth=true


###############################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
environmentName=environment_name


###############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
```

```
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
############################################################################
environmentType=Test

############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#    Specifies whether the server is intended for development purposes only.
#    This parameter is useful when creating profiles to test applications on a
#    non-production server prior to deploying the applications on their
#    production application servers.  If -isDeveloperServer is set when
#    creating an IBM Business Process Manager profile, then a preconfigured VMM
#    file repository is installed. This file repository contains a sample
#    organization that can be used to test Business Process Choreographer
#    people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
############################################################################
#isDeveloperServer=false

############################################################################
# Parameter:  samplesPassword
#
# Description:
#    Creates a password to be used for samples. The password is used to
#    restrict access to Web application samples installed during the
#    installation of the application server.
#
# Valid Values:
#    Varies
# Default Values:
#    None
############################################################################
samplesPassword=samplesPassword
```

Response file for an Advanced Process Server:

Use the **manageprofiles** command-line utility and the response file to create a stand-alone profile.

Response file for creating an Advanced Process Server stand-alone profile for DB2:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a DB2 database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# **************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
```

```
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ****************************************************************************
# END COPYRIGHT

############################################################################
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
############################################################################

############################################################################
# Parameter:  create
#
# Description:
#      Creates the profile.
#      If you are creating a profile, this parameter is required.
############################################################################
create

############################################################################
# Parameter:  templatePath
#
# Description:
#      Specifies the directory path to the template files in the installation
#      root directory. Within the profileTemplates directory are various
#      directories that correspond to different profile types and that vary with
#      the type of product installed. The profile directories are the paths that
#      you indicate while using the -templatePath option.
#      Use absolute paths. This parameter must exist as a directory and point to
#      a valid template directory.
#
#      When using the -templatePath parameter, specify the fully qualified file
#      path for the parameter.
#
# Valid Values:
#      Varies
# Default Values:
#      None
############################################################################
templatePath=BPM/default.procsvr.adv

############################################################################
# Parameter:  profileName
#
# Description:
#      Specifies the name of the profile. Use a unique value when creating a
#      profile.
#      Each profile that shares the same set of product binaries must have a
#      unique name. The default profile name is based on the profile type and a
#      trailing number, for example:
#      profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#      Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#      unique profile name. The value for this parameter must not contain spaces
#      or characters that are not valid such as the following: *, ?, ", <, >, ,,
```

```
#      /, \, and |. The profile name that you choose must not be in use. On
#      Windows platforms: If the fully qualified path contains spaces, enclose
#      the value in quotation marks.  The default value is based on the
#      install_root directory, the profiles subdirectory, and the name of the
#      file. For example, the default for profile creation is:
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#      file in the install_root/properties directory.  The value for this
#      parameter must be a valid path for the target system and must not be
#      currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
profileName=ProcSrv01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#     Set a performance tuning level appropriate for the profile you are
#     creating.
#     This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#    development
#    standard
#    production
# Default Values:
#    None
################################################################################
#applyPerfTuningSetting=standard

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
```

```
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
##############################################################################
cellName=cell_name
nodeName=node_name
serverName=server1


##############################################################################
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline.  To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
```

```
processCenterPassword=process_Center_Password

###########################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
###########################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

###########################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

###########################################################################
# Parameter:  defaultPorts
```

```
#
# Description:
#       Assigns the default or base port values to the profile.d o not use this
#       parameter when using the -startingPort or -portsFile parameter.
#       During profile creation, the manageprofiles command-line utility uses an
#       automatically generated set of recommended ports if you do not specify the
#       -startingPort parameter, the -defaultPorts parameter or the -portsFile
#       parameter. The recommended port values can be different than the default
#       port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#       Specifies the starting port number for generating and assigning all ports
#       for the profile.
#       Port values are assigned sequentially from the -startingPort value,
#       omitting those ports that are already in use. The system recognizes and
#       resolves ports that are currently in use and determines the port
#       assignments to avoid port conflicts.
#       Do not use this parameter with the -defaultPorts or -portsFile parameters.
#       During profile creation, the manageprofiles command-line utility uses an
#       automatically generated set of recommended ports if you do not specify the
#       -startingPort parameter, the -defaultPorts parameter or the -portsFile
#       parameter. The recommended port values can be different than the default
#       port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#       An optional parameter that specifies the path to a file that defines port
#       settings for the new profile.
#       Do not use this parameter when using the -startingPort or -defaultPorts
#       parameter.
#       During profile creation, the manageprofiles command-line utility uses an
#       automatically generated set of recommended ports if you do not specify the
#       -startingPort parameter, the -defaultPorts parameter or the -portsFile
#       parameter. The recommended port values can be different than the default
#       port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


##############################################################################
# Parameter:  enableService
#
# Description:
#       Enables the creation of a Linux service. Valid values include true or
#       false. The default value for this parameter is false.  When the
#       manageprofiles command-line utility is run with the -enableService option
#       set to true , the Linux service is created with the profile when the
#       command is run by the root user. When a nonroot user runs the
```

```
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#bpmdbDesign=design_file

################################################################################
# Parameter:  dbCreateNew
#
# Description:
```

```
#      Indicates a new database is created or if an existing database is to be
#      reused. Valid values are true or false. This value cannot be true for an
#      Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#      a DB2 database.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
################################################################################
dbCreateNew=true

################################################################################
# Parameter:  dbType
#
# Description:
#      The database type. Set one of the following values for the type of
#      database product you are using with IBM Business Process Manager.
#
#      Required only for one of the following:
#      A stand-alone profile when -bpmdbDesign was not specified.
#      An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     DB2_DATASERVER
# Default Values:
#     DB2_DATASERVER
#
# Parameter:  dbJDBCClasspath
#
# Description:
#      The directory path location of JDBC driver files.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbHostName
#
# Description:
#      The database server host name or IP address. The default value is the
#      local host name. Required when the -bpmdbDesign parameter was not
#      specified. (It is not required if the -bpmdbDesign parameter is
#      specified.)
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbServerPort
#
# Description:
#      The database server port number. Depending on the database you are using,
#      you can specify a different port number instead of the default port
#      number.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
```

```
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

##############################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##############################################################################
dbDelayConfig=false

##############################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     CMNDB
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
############################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

############################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
############################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

############################################################################
# Parameter:  perfDWDbName
#
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
```

```
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
################################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


################################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#     Process Server Messaging Engine user name. Required for setting up the
#     Process Server Messaging Engine. If you enter a user-specified a user name
#     and password during profile creation and if you specified ORACLE for
#     dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#dbProcSvrMeUserId=db_ProcSvr_ME_UserID
#dbProcSvrMePassword=db_ProcSvr_Me_Password


################################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#     Performance Data Warehouse Messaging Engine user name. Required for
#     setting up the Performance Data Warehouse Messaging Engine.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPerfDWMePassword
#
# Description:
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#dbPerfDWMeUserId=db_PerfDW_Me_UserId
#dbPerfDWMePassword=db_PerfDW_ME_Password
```

```
#############################################################################
# Parameter:  cdbSchemaName
#
# Description:
#      The schema name for the Common database. If no value is specified, the
#      following default database schema names are used:
#      DB2 database
#      The default schema name is the value specified by the dbUserId parameter.
#      The -cdbSchemaName parameter is valid for all database types except for
#      Oracle and Microsoft SQL Server.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  dbCommonUserId
#
# Description:
#      The user id to authenticate with the Common database.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  dbCommonPassword
#
# Description:
#      The password for the user id of the dbCommonUserId parameter.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#############################################################################
#cdbSchemaName=db_schema_name
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

#############################################################################
# Parameter:  dbAppMeUserId
#
# Description:
#      User name for application messaging engine.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  dbAppMePassword
#
# Description:
#      The password for the user id of the dbAppMeUserId parameter.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
```

```
# Valid Values:
#     Varies
# Default Values:
#     None
############################################################################
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd


############################################################################
# Parameter:  dbCeiMeUserId
#
# Description:
#     User name for CEI messaging engine.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbCeiMePassword
#
# Description:
#     The password for the user id of the dbCeiMeUserId parameter. If you enter
#     a user-specified a user name and password during profile creation and if
#     you specified ORACLE for -dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
############################################################################
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd


############################################################################
# Parameter:  dbSysMeUserId
#
# Description:
#     The default user name for the SCA system bus messaging engine is the first
#     three characters of the Oracle database service name. For example: If the
#     database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbSysMePassword
#
# Description:
#     The default password for the SCA system bus messaging engine is
#     dbPassword.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
############################################################################
#dbSysMeUserId=sys_me_userID
```

```
#dbSysMePassword=sys_me_pwd

###############################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
environmentName=environment_name

###############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
###############################################################################
environmentType=Test

###############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
###############################################################################
#isDeveloperServer=false
```

```
###########################################################################
# Parameter:  samplesPassword
#
# Description:
#     Creates a password to be used for samples. The password is used to
#     restrict access to Web application samples installed during the
#     installation of the application server.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
samplesPassword=samplesPassword

###########################################################################
# Parameter:  webFormConfig
#
# Description:
#     Indicates if Business Space is configured to use IBM Forms Server to work
#     with Human Task Management widgets. The default value for this parameter
#     is false. Indicate true to configure Business Space to use IBM Forms
#     Server. Both the webFormConfig and webFormInstallRoot parameters are
#     required to configure IBM Forms Server. This parameter is valid for
#     stand-alone server profiles only.
#     Note: IBM Forms Server configuration using these parameters is only valid
#     for local IBM Forms Server installations.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  webFormHTTP
#
# Description:
#     Specifies the location of the IBM Forms Server Translator. The default URL
#     for the location is http://localhost:8085/translator. No. This parameter
#     is valid only if the -webFormConfig parameter is set to true. It is valid
#     for stand-alone server profiles only.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  webFormInstallRoot
#
# Description:
#     Specifies the full path where IBM Forms Server is installed. For example,
#     the IBM Forms Server install root might be
#     C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#     if the -webFormConfig parameter is set to true. It is valid for
#     stand-alone server profiles only.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

###########################################################################
```

```
# Parameter:  configureBRM
#
# Description:
#     Configures the business process rules manager. The default value is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#########################################################################
configureBRM=false


#########################################################################
# Parameter:  configureBPC
#
# Description:
#     Determines whether the Business Process Choreographer sample configuration
#     is created.
#     The following conditions apply:
#     Only DB2 databases are supported.
#     The default is set to true for DB2.
#     It uses its own schema within the common database (CMNDB) and the schema
#     name is always BPEDB.
#     Business Process Choreographer database parameters set in the Database
#     Design Tool and generated to the database design file are ignored by
#     Business Process Choreographer at profile creation time.
#     See -bpmdbDesign for information about the database design file.
#     Note: If these conditions are too restrictive, consider setting
#     -configureBPC to false and using other documented methods to configure
#     Business Process Choreographer.
#
# Valid Values:
#     false
#     true
# Default Values:
#     true
#########################################################################
configureBPC=true
```

Response file for creating an Advanced Process Server stand-alone profile for DB2 for z/OS:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a DB2 for z/OS database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# *************************************************************************
# END COPYRIGHT

#########################################################################
# These response file has the applicable parameters for creating a
```

```
# Standalone PS Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
################################################################################

################################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
################################################################################
create

################################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/default.procsvr.adv

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
```

```
#      currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=ProcSrv01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#     Set a performance tuning level appropriate for the profile you are
#     creating.
#     This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#    development
#    standard
#    production
# Default Values:
#     None
################################################################################
#applyPerfTuningSetting=standard

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
```

```
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
################################################################################
cellName=cell_name
nodeName=node_name
serverName=server1


################################################################################
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline.  To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
```

```
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


################################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

################################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
```

```
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
```

```
#       The value can be either true or false. Specify true to create a Windows
#       service for the server process that is created within the profile. Specify
#       false to not create the Windows service. The default value for this
#       parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#       The following values for Windows service startup can be used:
#       manual
#       automatic
#       disabled
#       The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  bpmdbDesign
#
# Description:
#       Specifies the file that holds the database configuration design for all of
#       the IBM Business Process Manager components .
#       This parameter replaces the -wbidbDesign that was used in previous
#       versions of WebSphere Process Server.
#
#       Note: If a bpmdbDesign file parameter is specified, none of the database
#       related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#bpmdbDesign=design_file

################################################################################
# Parameter:  dbCreateNew
#
# Description:
#       Indicates a new database is created or if an existing database is to be
#       reused. Valid values are true or false. This value cannot be true for an
#       Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#       a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
```

```
##########################################################################
dbCreateNew=false

##########################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     DB2UDBOS390
# Default Values:
#     DB2UDBOS390
##########################################################################
dbType=DB2UDBOS390

##########################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##########################################################################
dbDelayConfig=true

##########################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     CMNDB
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
```

```
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPassword
#
# Description:
#    The password required for database authentication. Required when the
#    -bpmdbDesign file is not set.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##############################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

##############################################################################
# Parameter:  procSvrDbName
#
# Description:
#    Database name for Process Server database.  The value specified for
#    -procSvrDbName and -perfDWDbName must be different.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#    The Process Server user name. Required for setting up the Process Server.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#    The Process Server password. Required for setting up the Process Server.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##############################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password

##############################################################################
# Parameter:  perfDWDbName
#
# Description:
```

```
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
###############################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


###############################################################################
# Parameter:  dbStorageGroup
#
# Description:
#     The storage group name for DB2 z/OS databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbConnectionLocation
#
# Description:
#     The location of DB2 for z/OS database.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
dbStorageGroup=db_stg_group
dbConnectionLocation=db2_location


###############################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
environmentName=environment_name
```

```
################################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
################################################################################
environmentType=Test

################################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
################################################################################
#isDeveloperServer=false

################################################################################
# Parameter:  samplesPassword
#
# Description:
#     Creates a password to be used for samples. The password is used to
#     restrict access to Web application samples installed during the
#     installation of the application server.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
samplesPassword=samplesPassword
```

```
###########################################################################
# Parameter:  webFormConfig
#
# Description:
#     Indicates if Business Space is configured to use IBM Forms Server to work
#     with Human Task Management widgets. The default value for this parameter
#     is false. Indicate true to configure Business Space to use IBM Forms
#     Server. Both the webFormConfig and webFormInstallRoot parameters are
#     required to configure IBM Forms Server. This parameter is valid for
#     stand-alone server profiles only.
#     Note: IBM Forms Server configuration using these parameters is only valid
#     for local IBM Forms Server installations.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  webFormHTTP
#
# Description:
#     Specifies the location of the IBM Forms Server Translator. The default URL
#     for the location is http://localhost:8085/translator. No. This parameter
#     is valid only if the -webFormConfig parameter is set to true. It is valid
#     for stand-alone server profiles only.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  webFormInstallRoot
#
# Description:
#     Specifies the full path where IBM Forms Server is installed. For example,
#     the IBM Forms Server install root might be
#     C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#     if the -webFormConfig parameter is set to true. It is valid for
#     stand-alone server profiles only.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

###########################################################################
# Parameter:  configureBRM
#
# Description:
#     Configures the business process rules manager. The default value is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
###########################################################################
configureBRM=false
```

Response file for creating an Advanced Process Server stand-alone profile for Oracle:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for an Oracle database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

############################################################################
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
############################################################################

############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
############################################################################
create

############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
###########################################################################
templatePath=BPM/default.procsvr.adv

###########################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
profileName=ProcSrv01

###########################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#     Set a performance tuning level appropriate for the profile you are
#     creating.
#     This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#     development
#     standard
#     production
# Default Values:
#     None
###########################################################################
#applyPerfTuningSetting=standard

###########################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
```

```
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
################################################################################
cellName=cell_name
nodeName=node_name
serverName=server1


################################################################################
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline.  To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#    Varies
# Default Values:
```

```
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
#     or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
```

```
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


###############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
```

```
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

##############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
```

```
#      Act as part of the operating system
#      Log on as a service
#      The default value for this parameter is the current user name. The value
#      for this parameter must not contain spaces or characters that are not
#      valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#      you specify must have the proper permissions to create a Windows service.
#      You must specify the correct password for the user name that you choose.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  winservicePassword
#
# Description:
#      Specify the password for the specified user or the local account that is
#      to own the Windows service.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  winserviceCheck
#
# Description:
#      The value can be either true or false. Specify true to create a Windows
#      service for the server process that is created within the profile. Specify
#      false to not create the Windows service. The default value for this
#      parameter is false.
#
# Valid Values:
#      false
#      true
# Default Values:
#      false
#
# Parameter:  winserviceStartupType
#
# Description:
#      The following values for Windows service startup can be used:
#      manual
#      automatic
#      disabled
#      The default value for this parameter is manual.
#
# Valid Values:
#      manual
#      automatic
#      disabled
# Default Values:
#      manual
############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

############################################################################
# Parameter:  bpmdbDesign
#
# Description:
#      Specifies the file that holds the database configuration design for all of
```

```
#      the IBM Business Process Manager components .
#      This parameter replaces the -wbidbDesign that was used in previous
#      versions of WebSphere Process Server.
#
#      Note: If a bpmdbDesign file parameter is specified, none of the database
#      related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#bpmdbDesign=design_file

##############################################################################
# Parameter:   dbCreateNew
#
# Description:
#      Indicates a new database is created or if an existing database is to be
#      reused. Valid values are true or false. This value cannot be true for an
#      Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#      a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
##############################################################################
dbCreateNew=false

##############################################################################
# Parameter:   dbType
#
# Description:
#      The database type. Set one of the following values for the type of
#      database product you are using with IBM Business Process Manager.
#
#      Required only for one of the following:
#      A stand-alone profile when -bpmdbDesign was not specified.
#      An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     ORACLE
# Default Values:
#     ORACLE
#
# Parameter:   dbDriverType
#
# Description:
#      The database driver type. For an Oracle database, specify ORACLE. For
#      databases other than Oracle, the value is automatically set based on the
#      server operating system. Server installations on all operating systems use
#      type 4.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:   dbJDBCClasspath
#
# Description:
#      The directory path location of JDBC driver files.
```

```
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

################################################################################
# Parameter:  dbLocation
#
# Description:
#     The directory of the ORACLE_HOME directory. This parameter is required
#     when the parameter dbDelayConfig is set to false. No, this parameter is
#     only valid if you are using Oracle databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
dbLocation=db_location

################################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
################################################################################
dbDelayConfig=false

################################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     orcl
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
```

```
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPassword
#
# Description:
#    The password required for database authentication. Required when the
#    -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
############################################################################
dbName=orcl
dbUserId=db_userid
dbPassword=db_pswd


############################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName can be the same.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrUserId
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
############################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
```

```
dbProcSvrPassword=db_ProcSvr_Password

###########################################################################
# Parameter:  perfDWDbName
#
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName can be the same.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
###########################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password

###########################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#     Process Server Messaging Engine user name. Required for setting up the
#     Process Server Messaging Engine. If you enter a user-specified a user name
#     and password during profile creation and if you specified ORACLE for
#     dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password

###########################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#     Performance Data Warehouse Messaging Engine user name. Required for
#     setting up the Performance Data Warehouse Messaging Engine.  This
#     parameter is required.
```

```
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPerfDWMePassword
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password


################################################################################
# Parameter:  dbCommonUserId
#
# Description:
#    The user id to authenticate with the Common database. For Oracle, the
#    default user name is the first three characters of the Oracle database
#    service name. For example: If the database service name is ORCL,
#    dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#    Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#    profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#    and no bpmdbDesign file was specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCommonPassword
#
# Description:
#    The password for the user id of the dbCommonUserId parameter. Required for
#    all stand-alone IBM Business Process Manager profiles and advanced
#    deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#    to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd


################################################################################
# Parameter:  dbAppMeUserId
#
# Description:
#    The default user name is the first three characters of the Oracle database
#    service name. For example: If the database service name is ORCL,
#    dbAppMeUserId is ORCSA00. If you enter a user-specified a user name and
#    password during profile creation and if you specified ORACLE for dbType,
#    this parameter is required.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
```

```
# Default Values:
#     None
#
# Parameter:  dbAppMePassword
#
# Description:
#     The password for the user id of the dbAppMeUserId parameter. If you enter
#     a user-specified a user name and password during profile creation and if
#     you specified ORACLE for dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
dbAppMeUserId=app_me_userID
dbAppMePassword=app_me_pwd


################################################################################
# Parameter:  dbCeiMeUserId
#
# Description:
#     The default user name is the first three characters of the Oracle database
#     service name. For example: If the database service name is ORCL,
#     dbCeiMeUserId is ORCCM00. If you enter a user-specified a user name and
#     password during profile creation and if you specified ORACLE for dbType,
#     this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbCeiMePassword
#
# Description:
#     The password for the user id of the dbCeiMeUserId parameter. If you enter
#     a user-specified a user name and password during profile creation and if
#     you specified ORACLE for -dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
dbCeiMeUserId=cei_me_userID
dbCeiMePassword=cei_me_pwd


################################################################################
# Parameter:  dbSysMeUserId
#
# Description:
#     The default user name for the SCA system bus messaging engine is the first
#     three characters of the Oracle database service name. For example: If the
#     database service name is ORCL, dbSysMeUserId is ORCSS00. If you enter a
#     user-specified a user name and password during profile creation, this
#     parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
#
# Parameter:  dbSysMePassword
#
# Description:
#     The default password for the SCA system bus messaging engine is
#     dbPassword. If you enter a user-specified a user name and password during
#     profile creation, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
dbSysMeUserId=sys_me_userID
dbSysMePassword=sys_me_pwd


#############################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
environmentName=environment_name


#############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
#############################################################################
environmentType=Test


#############################################################################
```

```
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##############################################################################
#isDeveloperServer=false

##############################################################################
# Parameter:  samplesPassword
#
# Description:
#     Creates a password to be used for samples. The password is used to
#     restrict access to Web application samples installed during the
#     installation of the application server.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
samplesPassword=samplesPassword

##############################################################################
# Parameter:  webFormConfig
#
# Description:
#     Indicates if Business Space is configured to use IBM Forms Server to work
#     with Human Task Management widgets. The default value for this parameter
#     is false. Indicate true to configure Business Space to use IBM Forms
#     Server. Both the webFormConfig and webFormInstallRoot parameters are
#     required to configure IBM Forms Server. This parameter is valid for
#     stand-alone server profiles only.
#     Note: IBM Forms Server configuration using these parameters is only valid
#     for local IBM Forms Server installations.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  webFormHTTP
#
# Description:
#     Specifies the location of the IBM Forms Server Translator. The default URL
#     for the location is http://localhost:8085/translator. No. This parameter
#     is valid only if the -webFormConfig parameter is set to true. It is valid
#     for stand-alone server profiles only.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
```

334 Installing

```
# Parameter:  webFormInstallRoot
#
# Description:
#     Specifies the full path where IBM Forms Server is installed. For example,
#     the IBM Forms Server install root might be
#     C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#     if the -webFormConfig parameter is set to true. It is valid for
#     stand-alone server profiles only.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

################################################################################
# Parameter:  configureBRM
#
# Description:
#     Configures the business process rules manager. The default value is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
################################################################################
configureBRM=false

################################################################################
# Parameter:  configureBPC
#
# Description:
#     Determines whether the Business Process Choreographer sample configuration
#     is created.
#     The following conditions apply:
#     Only DB2 databases are supported.
#     The default is set to true for DB2.
#     It uses its own schema within the common database (CMNDB) and the schema
#     name is always BPEDB.
#     Business Process Choreographer database parameters set in the Database
#     Design Tool and generated to the database design file are ignored by
#     Business Process Choreographer at profile creation time.
#     See -bpmdbDesign for information about the database design file.
#     Note: If these conditions are too restrictive, consider setting
#     -configureBPC to false and using other documented methods to configure
#     Business Process Choreographer.
#
# Valid Values:
#     false
# Default Values:
#     false
################################################################################
configureBPC=false
```

Response file for creating an Advanced Process Server stand-alone profile for SQL:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a stand-alone profile for a SQL database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# *************************************************************************
# END COPYRIGHT

###############################################################################
# These response file has the applicable parameters for creating a
# Standalone PS Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_Standalone_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###############################################################################

###############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###############################################################################
create

###############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
templatePath=BPM/default.procsvr.adv

###############################################################################
# Parameter:  profileName
#
# Description:
```

```
#      Specifies the name of the profile. Use a unique value when creating a
#      profile.
#      Each profile that shares the same set of product binaries must have a
#      unique name. The default profile name is based on the profile type and a
#      trailing number, for example:
#      profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#      Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#      unique profile name. The value for this parameter must not contain spaces
#      or characters that are not valid such as the following: *, ?, ", <, >, ,,
#      /, \, and |. The profile name that you choose must not be in use. On
#      Windows platforms: If the fully qualified path contains spaces, enclose
#      the value in quotation marks.  The default value is based on the
#      install_root directory, the profiles subdirectory, and the name of the
#      file. For example, the default for profile creation is:
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#      file in the install_root/properties directory.  The value for this
#      parameter must be a valid path for the target system and must not be
#      currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
profileName=ProcSrv01

################################################################################
# Parameter:  applyPerfTuningSetting
#
# Description:
#     Set a performance tuning level appropriate for the profile you are
#     creating.
#     This parameter is a WebSphere Application Server parameter.
#
# Valid Values:
#    development
#    standard
#    production
# Default Values:
#    None
################################################################################
#applyPerfTuningSetting=standard

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#        shortHostNameCellCellNumber
#     else
#        shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
```

```
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  serverName
#
# Description:
#     Specifies the name of the server.
################################################################################
cellName=cell_name
nodeName=node_name
serverName=server1

################################################################################
# Parameter:  processCenterURL
#
# Description:
#     Specifies the URL of the Process Center that the Process Server connects
#     to. If this parameter is set, the protocol must be either http or https.
#     Note: No corresponding useServerOffline parameter exists to use the server
#     offline.  To use the server offline, omit the processCenterURL parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterUserId
#
# Description:
#     Specifies the user ID to be used to connect to the Process Center. This
#     parameter is valid for stand-alone Process Server profiles. This parameter
#     is required if the Process Server is online (connected to a Process
#     Center). It is required for a stand-alone process server (standard or
#     advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  processCenterPassword
#
# Description:
#     Specifies the user password to be used to connect to the Process Center.
#     This parameter is valid for stand-alone Process Server profiles. This
#     parameter is required if the Process Server is online (connected to a
#     Process Center). It is required for a stand-alone process server (standard
```

```
#      or advanced) profile if the processCenterURL parameter is specified.
#
# Valid Values:
#      Varies
# Default Values:
#      None
################################################################################
processCenterURL=process_Center_URL
processCenterUserId=process_Center_UserId
processCenterPassword=process_Center_Password

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#      true
# Default Values:
#      true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

################################################################################
# Parameter:  signingCertDN
#
# Description:
#      Specifies the distinguished name of the root signing certificate that you
#      create when you create the profile. Specify the distinguished name in
#      quotation marks. This default personal certificate is located in the
#      server keystore file. If you do not specifically create or import a root
#      signing certificate, one is created by default. See the
#      -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#      An optional parameter that specifies the amount of time in years that the
#      root signing certificate is valid. If you do not specify this parameter
#      with the -signingCertDN parameter, the root signing certificate is valid
#      for 20 years.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  keyStorePassword
#
# Description:
#      Specifies the password to use on all keystore files created during profile
#      creation. Keystore files are created for the default personal certificate
#      and the root signing certificate.
#
# Valid Values:
#      Varies
```

```
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
```

```
#############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
```

```
#       Varies
# Default Values:
#       None
#
# Parameter:  winservicePassword
#
# Description:
#       Specify the password for the specified user or the local account that is
#       to own the Windows service.
#
# Valid Values:
#       Varies
# Default Values:
#       None
#
# Parameter:  winserviceCheck
#
# Description:
#       The value can be either true or false. Specify true to create a Windows
#       service for the server process that is created within the profile. Specify
#       false to not create the Windows service. The default value for this
#       parameter is false.
#
# Valid Values:
#       false
#       true
# Default Values:
#       false
#
# Parameter:  winserviceStartupType
#
# Description:
#       The following values for Windows service startup can be used:
#       manual
#       automatic
#       disabled
#       The default value for this parameter is manual.
#
# Valid Values:
#       manual
#       automatic
#       disabled
# Default Values:
#       manual
###############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


###############################################################################
# Parameter:  bpmdbDesign
#
# Description:
#       Specifies the file that holds the database configuration design for all of
#       the IBM Business Process Manager components .
#       This parameter replaces the -wbidbDesign that was used in previous
#       versions of WebSphere Process Server.
#
#       Note: If a bpmdbDesign file parameter is specified, none of the database
#       related properties should be specified.
#
# Valid Values:
#       Varies
```

```
# Default Values:
#     None
##############################################################################
#bpmdbDesign=design_file

##############################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
##############################################################################
dbCreateNew=false

##############################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     MSSQLSERVER_MICROSOFT
# Default Values:
#     MSSQLSERVER_MICROSOFT
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer

##############################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#     false
#     true
```

```
# Default Values:
#     false
##############################################################################
dbDelayConfig=false

##############################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     CMNDB
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

##############################################################################
# Parameter:  procSvrDbName
#
# Description:
#     Database name for Process Server database.  The value specified for
#     -procSvrDbName and -perfDWDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbProcSvrUserId
```

344 Installing

```
#
# Description:
#     The Process Server user name. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrPassword
#
# Description:
#     The Process Server password. Required for setting up the Process Server.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
procSvrDbName=procSvr_Db_Name
dbProcSvrUserId=db_ProcSvr_UserId_name
dbProcSvrPassword=db_ProcSvr_Password


################################################################################
# Parameter:  perfDWDbName
#
# Description:
#     Database name for Performance Data Warehouse.  The value specified for
#     -perfDWDbName and -procSvrDbName must be different.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWUserId
#
# Description:
#     Performance Data Warehouse user name. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Parameter:  dbPerfDWPassword
#
# Description:
#     Performance Data Warehouse password. Required for setting up the
#     Performance Data Warehouse.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
################################################################################
perfDWDbName=perfDW_Db_Name
dbPerfDWUserId=db_PerfDW_UserId
dbPerfDWPassword=db_PerfDW_Password


################################################################################
# Parameter:  dbProcSvrMeUserId
#
# Description:
#     Process Server Messaging Engine user name. Required for setting up the
#     Process Server Messaging Engine. If you enter a user-specified a user name
#     and password during profile creation and if you specified ORACLE for
#     dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
```

```
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbProcSvrMePassword
#
# Description:
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
dbProcSvrMeUserId=db_ProcSvr_ME_UserID
dbProcSvrMePassword=db_ProcSvr_Me_Password


###########################################################################
# Parameter:  dbPerfDWMeUserId
#
# Description:
#     Performance Data Warehouse Messaging Engine user name. Required for
#     setting up the Performance Data Warehouse Messaging Engine.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPerfDWMePassword
#
# Valid Values:
#    Varies
# Default Values:
#    None
###########################################################################
dbPerfDWMeUserId=db_PerfDW_Me_UserId
dbPerfDWMePassword=db_PerfDW_ME_Password


###########################################################################
# Parameter:  dbCommonUserId
#
# Description:
#     The user id to authenticate with the Common database. For Oracle, the
#     default user name is the first three characters of the Oracle database
#     service name. For example: If the database service name is ORCL,
#     dbCommonUserId is ORCCOMM. Required for all stand-alone IBM Business
#     Process Manager profiles and advanced deployment manager (dmgr.*.adv)
#     profiles, if the -dbType parameter is set to ORACLE or MICROSOFT_SQLSERVER
#     and no bpmdbDesign file was specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCommonPassword
#
# Description:
#     The password for the user id of the dbCommonUserId parameter. Required for
#     all stand-alone IBM Business Process Manager profiles and advanced
#     deployment manager (dmgr.*.adv) profiles, if the -dbType parameter is set
#     to ORACLE or MICROSOFT_SQLSERVER and no bpmdbDesign file was specified.
#
```

```
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd

################################################################################
# Parameter:  dbAppMeUserId
#
# Description:
#     User name for application messaging engine.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbAppMePassword
#
# Description:
#     The password for the user id of the dbAppMeUserId parameter.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#dbAppMeUserId=app_me_userID
#dbAppMePassword=app_me_pwd

################################################################################
# Parameter:  dbCeiMeUserId
#
# Description:
#     User name for CEI messaging engine.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCeiMePassword
#
# Description:
#     The password for the user id of the dbCeiMeUserId parameter. If you enter
#     a user-specified a user name and password during profile creation and if
#     you specified ORACLE for -dbType, this parameter is required.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#dbCeiMeUserId=cei_me_userID
#dbCeiMePassword=cei_me_pwd
```

```
###############################################################################
# Parameter:  dbSysMeUserId
#
# Description:
#     The default user name for the SCA system bus messaging engine is the first
#     three characters of the Oracle database service name. For example: If the
#     database service name is ORCL, dbSysMeUserId is ORCSS00.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbSysMePassword
#
# Description:
#     The default password for the SCA system bus messaging engine is
#     dbPassword.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###############################################################################
#dbSysMeUserId=sys_me_userID
#dbSysMePassword=sys_me_pwd

###############################################################################
# Parameter:  dbWinAuth
#
# Description:
#     An optional parameter that is valid in all default profile templates.
#     Specify true to indicate that you are running SQL Server using integrated
#     windows authentication.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
###############################################################################
#dbWinAuth=true

###############################################################################
# Parameter:  environmentName
#
# Description:
#     Specifies the name of the environment you are configuring. This parameter
#     has no default value, and it must not be empty.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###############################################################################
environmentName=environment_name

###############################################################################
# Parameter:  environmentType
#
```

```
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
###############################################################################
environmentType=Test

###############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
###############################################################################
#isDeveloperServer=false

###############################################################################
# Parameter:  samplesPassword
#
# Description:
#     Creates a password to be used for samples. The password is used to
#     restrict access to Web application samples installed during the
#     installation of the application server.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
samplesPassword=samplesPassword

###############################################################################
# Parameter:  webFormConfig
#
```

```
# Description:
#     Indicates if Business Space is configured to use IBM Forms Server to work
#     with Human Task Management widgets. The default value for this parameter
#     is false. Indicate true to configure Business Space to use IBM Forms
#     Server. Both the webFormConfig and webFormInstallRoot parameters are
#     required to configure IBM Forms Server. This parameter is valid for
#     stand-alone server profiles only.
#     Note: IBM Forms Server configuration using these parameters is only valid
#     for local IBM Forms Server installations.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  webFormHTTP
#
# Description:
#     Specifies the location of the IBM Forms Server Translator. The default URL
#     for the location is http://localhost:8085/translator. No. This parameter
#     is valid only if the -webFormConfig parameter is set to true. It is valid
#     for stand-alone server profiles only.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  webFormInstallRoot
#
# Description:
#     Specifies the full path where IBM Forms Server is installed. For example,
#     the IBM Forms Server install root might be
#     C:/IBM/IBMFormsServer/4.0/WebFormServer. No. This parameter is valid only
#     if the -webFormConfig parameter is set to true. It is valid for
#     stand-alone server profiles only.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#webFormConfig=false
#webFormHTTP=URL
#webFormInstallRoot=Webform_Server_install_root

################################################################################
# Parameter:  configureBRM
#
# Description:
#     Configures the business process rules manager. The default value is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
################################################################################
configureBRM=false

################################################################################
# Parameter:  configureBPC
#
# Description:
#     Determines whether the Business Process Choreographer sample configuration
#     is created.
```

```
#     The following conditions apply:
#     Only DB2 databases are supported.
#     The default is set to true for DB2.
#     It uses its own schema within the common database (CMNDB) and the schema
#     name is always BPEDB.
#     Business Process Choreographer database parameters set in the Database
#     Design Tool and generated to the database design file are ignored by
#     Business Process Choreographer at profile creation time.
#     See -bpmdbDesign for information about the database design file.
#     Note: If these conditions are too restrictive, consider setting
#     -configureBPC to false and using other documented methods to configure
#     Business Process Choreographer.
#
# Valid Values:
#     false
# Default Values:
#     false
################################################################################
configureBPC=false
```

Creating deployment manager and custom profiles using manageprofiles after a Custom installation:

After performing a Custom installation you can create deployment manager and custom (managed node) profiles using the Profile Management Tool or the manageprofiles command-line utility.

The information in this section describes how to use the manageprofiles command-line utility to create deployment manager and custom (managed node) profiles for a network deployment configuration. It assumes that you have run the installer and have performed a Custom installation.

For information about using the Profile Management Tool to create deployment manager and custom (managed node) profiles after performing a Custom installation, see Creating custom profiles (managed nodes) by using the Profile Management Tool.

Creating Process Center deployment manager and custom profiles using manageprofiles:

Use the **manageprofiles** command-line utility to create Process Center deployment manager and custom profiles for a network deployment configuration .

Before you run the **manageprofiles** command-line utility ensure that you have completed the following tasks:
- You have reviewed the full list of prerequisites for creating or augmenting a profile
- You have reviewed example profile creation commands
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

This task describes how to use the **manageprofiles** command-line utility to create Process Center deployment manager and custom profiles for a network deployment configuration.

To use the **manageprofiles** command-line utility to create a profile, perform the following steps.
1. Determine the kind of profile you want to create, which in turn determines the template to use for your new profile (using the **-templatePath** option).

   Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates.

   The following templates are available:
   - dmgr.procctr: for a Process Center deployment manager profile, which defines a deployment manager. A *deployment manager* provides one administrative interface to a logical group of servers on one or more workstations.

- managed.procctr: for a Process Center custom profile, which, when federated to a deployment manager, defines a managed node. If you have decided that your solution requires a deployment environment, your runtime environment requires one or more managed nodes. A *custom profile* contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node. Do not federate a node unless the deployment manager you are federating to is at a release level the same or higher than that of the custom profile you are creating.

2. Determine which parameters are required for your type of profile by reviewing the example profile creation commands in manageprofile examples.

3. Determine the values that you want to supply for the profile by reviewing the default values in the manageprofiles parameters topic to see if they are what you need for your profile.

4. Create a response file.

   Sample response files are available under the `install_root/BPM/samples/manageprofiles/` directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server) and customize the file for your environment.

   Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. See Creating a response file using the IBM Business Process Manager samples.

5. Run the **manageprofiles** command from the command line, using the **-response** parameter and the name of the response file that you created.

   - `Linux` `UNIX` **manageprofiles.sh -response** *myResponseFile*

   - `Windows` **manageprofiles.bat -response** *myResponseFile*

   The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

Response file for a Standard Process Center:

Use the **manageprofiles** command-line utility and the response file to create a deployment manager profile.

Response file for creating a Standard Process Center deployment manager profile for DB2:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a DB2 database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

###########################################################################
```

```
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMgr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
################################################################################

################################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
################################################################################
create

################################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/dmgr.procctr

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
```

```
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=Dmgr01

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
################################################################################
cellName=cell_name
nodeName=node_name

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
```

```
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


################################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
```

```
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  startingPort
#
# Description:
#    Specifies the starting port number for generating and assigning all ports
#    for the profile.
#    Port values are assigned sequentially from the -startingPort value,
#    omitting those ports that are already in use. The system recognizes and
#    resolves ports that are currently in use and determines the port
#    assignments to avoid port conflicts.
#    Do not use this parameter with the -defaultPorts or -portsFile parameters.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  portsFile
#
# Description:
#    An optional parameter that specifies the path to a file that defines port
#    settings for the new profile.
#    Do not use this parameter when using the -startingPort or -defaultPorts
#    parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

################################################################################
# Parameter:  enableService
#
# Description:
#    Enables the creation of a Linux service. Valid values include true or
#    false. The default value for this parameter is false.  When the
#    manageprofiles command-line utility is run with the -enableService option
#    set to true , the Linux service is created with the profile when the
#    command is run by the root user. When a nonroot user runs the
#    manageprofiles command-line utility, the profile is created, but the Linux
#    service is not. The Linux service is not created because the nonroot user
#    does not have sufficient permission to set up the service. An
#    INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#    creation and the profile creation log
#    install_root/logs/manageprofiles/profile_name_create.log contains a
#    message indicating the current user does not have sufficient permission to
#    set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
```

```
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#      Specifies the user ID that is used during the creation of the Linux
#      service so that the Linux service will run under this user ID. The Linux
#      service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#      The type of the owner account of the Windows service created for the
#      profile. Valid values include specifieduser or localsystem. The
#      localsystem value runs the Windows service under the local account of the
#      user who creates the profile. The default value for this parameter is
#      localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#      Specify your user ID so that the Windows operating system can verify you
#      as an ID that is capable of creating a Windows service. Your user ID must
#      belong to the Administrator group and have the following advanced user
#      rights:
#      Act as part of the operating system
#      Log on as a service
#      The default value for this parameter is the current user name. The value
#      for this parameter must not contain spaces or characters that are not
#      valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#      you specify must have the proper permissions to create a Windows service.
#      You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#      Specify the password for the specified user or the local account that is
#      to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#      The value can be either true or false. Specify true to create a Windows
```

```
#       service for the server process that is created within the profile. Specify
#       false to not create the Windows service. The default value for this
#       parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#       The following values for Windows service startup can be used:
#       manual
#       automatic
#       disabled
#       The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

##############################################################################
# Parameter:  environmentType
#
# Description:
#       Specifies the environment type. The environment type refers to how IBM
#       Business Process Manager is used; for example, in a production, stage or
#       test environment. Load testing might be done on a test server, while a
#       stage environment type might be used as a temporary location to host
#       changes before putting those changes into production. You might specify
#       Stage as the Environment type if the server you are configuring will be
#       accessed and used to review content and new functionality.  Valid values
#       are as follows:
#       Test
#       Use Test if the server you are configuring is to be used as a testing
#       environment.
#       Stage
#       Use Stage if the server is to serve as a staging platform to be used as a
#       preproduction server.
#       Production
#       Use Production if the server is to serve in a production capacity.
#       The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
##############################################################################
environmentType=Test

##############################################################################
# Parameter:  isDeveloperServer
```

```
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
###############################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Center deployment manager profile for DB2 for z/OS:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a DB2 for z/OS database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

###############################################################################
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMgr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###############################################################################

###############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###############################################################################
```

```
create


##############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
templatePath=BPM/dmgr.procctr


##############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
profileName=Dmgr01


##############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
```

```
#      where CellNumber is a sequential number starting at 01 and NodeNumber is
#      the node number that you used to define the node name.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#      Specifies the node name for the node that is created with the new profile.
#      Use a unique value within the cell or on the workstation. Each profile
#      that shares the same set of product binaries must have a unique node name.
#      This parameter is required for profile creation only with the
#      dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#      procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#      The default value for this parameter is based on the short host name,
#      profile type, and a trailing number, for example:
#      if (DMgr)
#              shortHostNameCellManagerNodeNumber
#      else
#              shortHostNameNodeNodeNumber
#      where NodeNumber is a sequential number starting at 01.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
################################################################################
cellName=cell_name
nodeName=node_name


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#      Specifies the distinguished name of the root signing certificate that you
#      create when you create the profile. Specify the distinguished name in
#      quotation marks. This default personal certificate is located in the
#      server keystore file. If you do not specifically create or import a root
#      signing certificate, one is created by default. See the
#      -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

###############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
```

```
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
```

```
#    disabled
# Default Values:
#    manual
###########################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


###########################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
###########################################################################
environmentType=Test

###########################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
###########################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Center deployment manager profile for Oracle:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for an Oracle database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

#############################################################################
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMgr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#############################################################################

#############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
#############################################################################
create

#############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
############################################################################
templatePath=BPM/dmgr.procctr

############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
############################################################################
profileName=Dmgr01

############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
```

```
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
################################################################################
cellName=cell_name
nodeName=node_name


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
```

368   Installing

```
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter: portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
###########################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


###########################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
```

370 Installing

```
#    The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  dbDriverType
#
# Description:
#     The database driver type. For an Oracle database, specify ORACLE. For
#     databases other than Oracle, the value is automatically set based on the
#     server operating system. Server installations on all operating systems use
```

```
#      type 4.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#      Varies
# Default Values:
#      None
###############################################################################
dbDriverType=ORACLE_THIN

###############################################################################
# Parameter:  environmentType
#
# Description:
#      Specifies the environment type. The environment type refers to how IBM
#      Business Process Manager is used; for example, in a production, stage or
#      test environment. Load testing might be done on a test server, while a
#      stage environment type might be used as a temporary location to host
#      changes before putting those changes into production. You might specify
#      Stage as the Environment type if the server you are configuring will be
#      accessed and used to review content and new functionality.  Valid values
#      are as follows:
#      Test
#      Use Test if the server you are configuring is to be used as a testing
#      environment.
#      Stage
#      Use Stage if the server is to serve as a staging platform to be used as a
#      preproduction server.
#      Production
#      Use Production if the server is to serve in a production capacity.
#      The default value is Test.
#
# Valid Values:
#      Test
#      Production
#      Stage
# Default Values:
#      Test
###############################################################################
environmentType=Test

###############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#      Specifies whether the server is intended for development purposes only.
#      This parameter is useful when creating profiles to test applications on a
#      non-production server prior to deploying the applications on their
#      production application servers.  If -isDeveloperServer is set when
#      creating an IBM Business Process Manager profile, then a preconfigured VMM
#      file repository is installed. This file repository contains a sample
#      organization that can be used to test Business Process Choreographer
#      people resolution, ready for you to use as is.
#
# Valid Values:
#      false
#      true
# Default Values:
#      false
###############################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Center deployment manager profile for SQL:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a SQL database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

###########################################################################
# These response file has the applicable parameters for creating a
# DMgr PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###########################################################################

###########################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###########################################################################
create

###########################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
#############################################################################
templatePath=BPM/dmgr.procctr

#############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
profileName=Dmgr01

#############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
```

```
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
################################################################################
cellName=cell_name
nodeName=node_name


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
```

```
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#    None
##############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path

##############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
```

```
#      The default value for this parameter is the current user name. The value
#      for this parameter must not contain spaces or characters that are not
#      valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#      you specify must have the proper permissions to create a Windows service.
#      You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#      Specify the password for the specified user or the local account that is
#      to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#      The value can be either true or false. Specify true to create a Windows
#      service for the server process that is created within the profile. Specify
#      false to not create the Windows service. The default value for this
#      parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#      The following values for Windows service startup can be used:
#      manual
#      automatic
#      disabled
#      The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
###########################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

###########################################################################
# Parameter:  dbWinAuth
#
# Description:
#      An optional parameter that is valid in all default profile templates.
#      Specify true to indicate that you are running SQL Server using integrated
#      windows authentication.
```

378   Installing

```
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
################################################################################
#dbWinAuth=true

################################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
################################################################################
environmentType=Test

################################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
################################################################################
#isDeveloperServer=false
```

Response file for a Standard Process Center:

Use the **manageprofiles** command-line utility and the response file to create a custom profile.

Response file for creating a Standard Process Center custom profile for DB2:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a custom profile for a DB2 database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

#############################################################################
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#############################################################################

#############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
#############################################################################
create

#############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
##########################################################################
templatePath=BPM/managed.procctr

##########################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##########################################################################
profileName=Custom01

##########################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
```

```
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
################################################################################
cellName=cell_name
nodeName=node_name

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
```

```
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


###############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
```

```
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
###############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual
```

```
##############################################################################
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.   If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#    false
#    true
# Default Values:
#    true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.   The
#     host name can be the long or short DNS name or the IP address of the
#     deployment manager workstation. Specifying this optional parameter directs
#     the manageprofiles command-line utility to attempt to federate the custom
#     node into the deployment manager cell as it creates the custom profile.
#     This parameter is ignored when creating a deployment manager profile or
#     stand-alone server profile. If you federate a custom node when the
#     deployment manager is not running, the installation indicator in the logs
#     is INSTCONFFAILED to indicate a complete failure. The resulting custom
#     profile is unusable. You must move the custom profile directory out of the
#     profile repository (the profile's installation root directory) before
#     creating another custom profile with the same profile name. If you have
#     changed the default JMX connector type, you cannot federate with the
#     manageprofiles command-line utility. Use the addNode command later after
#     the node is created instead. The default value for this parameter is
#     localhost. The value for this parameter must be a properly formed host
#     name and must not contain spaces or characters that are not valid such as
#     the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#     deployment manager must also be available in conjunction with the dmgrPort
#     parameter.  Required if you are creating a managed profile with the
#     -federateLaterBPM parameter set to false and the deployment manager is not
#     on the local machine.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dmgrPort
#
# Description:
#     Identifies the SOAP port of the deployment manager. Specify this parameter
#     and the dmgrHost parameter to federate a custom profile as it is created
#     or augmented. The deployment manager must be running and accessible.  If
#     you have changed the default JMX connector type, you cannot federate with
#     the manageprofiles command-line utility. Use the addNode command later
#     after the node is created instead. The default value for this parameter is
#     8879. The port that you indicate must be a positive integer and a
#     connection to the deployment manager must be available in conjunction with
#     the dmgrHost parameter. Required if you are creating a managed profile,
#     the -federateLaterBPM parameter is set to false, and the deployment
#     manager was configured with a port that was not a default port.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password


################################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
################################################################################
environmentType=Test


################################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
################################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Center custom profile for DB2 for z/OS:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a custom profile for a DB2 for z/OS database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

###########################################################################
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###########################################################################

###########################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###########################################################################
create

###########################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
############################################################################
templatePath=BPM/managed.procctr

############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
############################################################################
profileName=Custom01

############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
```

```
#      The default value for this parameter is based on the short host name,
#      profile type, and a trailing number, for example:
#      if (DMgr)
#              shortHostNameCellManagerNodeNumber
#      else
#              shortHostNameNodeNodeNumber
#      where NodeNumber is a sequential number starting at 01.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
################################################################################
cellName=cell_name
nodeName=node_name


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#      Specifies the distinguished name of the root signing certificate that you
#      create when you create the profile. Specify the distinguished name in
#      quotation marks. This default personal certificate is located in the
#      server keystore file. If you do not specifically create or import a root
#      signing certificate, one is created by default. See the
#      -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#      An optional parameter that specifies the amount of time in years that the
#      root signing certificate is valid. If you do not specify this parameter
#      with the -signingCertDN parameter, the root signing certificate is valid
#      for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
```

```
# Description:
#      Specifies the password to use on all keystore files created during profile
#      creation. Keystore files are created for the default personal certificate
#      and the root signing certificate.
#
# Valid Values:
#      Varies
# Default Values:
#      None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


################################################################################
# Parameter:  enableService
#
# Description:
#      Enables the creation of a Linux service. Valid values include true or
#      false. The default value for this parameter is false.  When the
#      manageprofiles command-line utility is run with the -enableService option
#      set to true , the Linux service is created with the profile when the
#      command is run by the root user. When a nonroot user runs the
#      manageprofiles command-line utility, the profile is created, but the Linux
#      service is not. The Linux service is not created because the nonroot user
#      does not have sufficient permission to set up the service. An
#      INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#      creation and the profile creation log
#      install_root/logs/manageprofiles/profile_name_create.log contains a
#      message indicating the current user does not have sufficient permission to
#      set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#      false
#      true
# Default Values:
#      None
#
# Parameter:  serviceUserName
#
# Description:
#      Specifies the user ID that is used during the creation of the Linux
#      service so that the Linux service will run under this user ID. The Linux
#      service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  winserviceAccountType
#
# Description:
#      The type of the owner account of the Windows service created for the
#      profile. Valid values include specifieduser or localsystem. The
#      localsystem value runs the Windows service under the local account of the
#      user who creates the profile. The default value for this parameter is
#      localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#      localsystem
#      specifieduser
# Default Values:
#      localsystem
#
# Parameter:  winserviceUserName
#
```

```
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual
```

```
#############################################################################
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.   If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#     false
#     true
# Default Values:
#     true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.  The
#     host name can be the long or short DNS name or the IP address of the
#     deployment manager workstation. Specifying this optional parameter directs
#     the manageprofiles command-line utility to attempt to federate the custom
#     node into the deployment manager cell as it creates the custom profile.
#     This parameter is ignored when creating a deployment manager profile or
#     stand-alone server profile. If you federate a custom node when the
#     deployment manager is not running, the installation indicator in the logs
#     is INSTCONFFAILED to indicate a complete failure. The resulting custom
#     profile is unusable. You must move the custom profile directory out of the
#     profile repository (the profile's installation root directory) before
#     creating another custom profile with the same profile name. If you have
#     changed the default JMX connector type, you cannot federate with the
#     manageprofiles command-line utility. Use the addNode command later after
#     the node is created instead. The default value for this parameter is
#     localhost. The value for this parameter must be a properly formed host
#     name and must not contain spaces or characters that are not valid such as
#     the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#     deployment manager must also be available in conjunction with the dmgrPort
#     parameter.  Required if you are creating a managed profile with the
#     -federateLaterBPM parameter set to false and the deployment manager is not
#     on the local machine.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dmgrPort
#
# Description:
#     Identifies the SOAP port of the deployment manager. Specify this parameter
#     and the dmgrHost parameter to federate a custom profile as it is created
#     or augmented. The deployment manager must be running and accessible.  If
#     you have changed the default JMX connector type, you cannot federate with
#     the manageprofiles command-line utility. Use the addNode command later
#     after the node is created instead. The default value for this parameter is
#     8879. The port that you indicate must be a positive integer and a
#     connection to the deployment manager must be available in conjunction with
#     the dmgrHost parameter. Required if you are creating a managed profile,
#     the -federateLaterBPM parameter is set to false, and the deployment
#     manager was configured with a port that was not a default port.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password


################################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
################################################################################
environmentType=Test


################################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
################################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Center custom profile for Oracle:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a custom profile for an Oracle database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

###########################################################################
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###########################################################################

###########################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###########################################################################
create

###########################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
###########################################################################
templatePath=BPM/managed.procctr

###########################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
profileName=Custom01

###########################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
```

```
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
################################################################################
cellName=cell_name
nodeName=node_name


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
```

```
# Description:
#      Specifies the password to use on all keystore files created during profile
#      creation. Keystore files are created for the default personal certificate
#      and the root signing certificate.
#
# Valid Values:
#      Varies
# Default Values:
#      None
############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

############################################################################
# Parameter:  enableService
#
# Description:
#      Enables the creation of a Linux service. Valid values include true or
#      false. The default value for this parameter is false.  When the
#      manageprofiles command-line utility is run with the -enableService option
#      set to true , the Linux service is created with the profile when the
#      command is run by the root user. When a nonroot user runs the
#      manageprofiles command-line utility, the profile is created, but the Linux
#      service is not. The Linux service is not created because the nonroot user
#      does not have sufficient permission to set up the service. An
#      INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#      creation and the profile creation log
#      install_root/logs/manageprofiles/profile_name_create.log contains a
#      message indicating the current user does not have sufficient permission to
#      set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#      false
#      true
# Default Values:
#      None
#
# Parameter:  serviceUserName
#
# Description:
#      Specifies the user ID that is used during the creation of the Linux
#      service so that the Linux service will run under this user ID. The Linux
#      service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  winserviceAccountType
#
# Description:
#      The type of the owner account of the Windows service created for the
#      profile. Valid values include specifieduser or localsystem. The
#      localsystem value runs the Windows service under the local account of the
#      user who creates the profile. The default value for this parameter is
#      localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#      localsystem
#      specifieduser
# Default Values:
#      localsystem
#
# Parameter:  winserviceUserName
#
```

```
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
###############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual
```

```
#############################################################################
# Parameter:  dbDriverType
#
# Description:
#     The database driver type. For an Oracle database, specify ORACLE. For
#     databases other than Oracle, the value is automatically set based on the
#     server operating system. Server installations on all operating systems use
#     type 4.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
dbDriverType=ORACLE_THIN

#############################################################################
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.    If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#     false
#     true
# Default Values:
#     true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.  The
#     host name can be the long or short DNS name or the IP address of the
#     deployment manager workstation. Specifying this optional parameter directs
#     the manageprofiles command-line utility to attempt to federate the custom
#     node into the deployment manager cell as it creates the custom profile.
#     This parameter is ignored when creating a deployment manager profile or
#     stand-alone server profile. If you federate a custom node when the
#     deployment manager is not running, the installation indicator in the logs
#     is INSTCONFFAILED to indicate a complete failure. The resulting custom
#     profile is unusable. You must move the custom profile directory out of the
#     profile repository (the profile's installation root directory) before
#     creating another custom profile with the same profile name. If you have
#     changed the default JMX connector type, you cannot federate with the
#     manageprofiles command-line utility. Use the addNode command later after
#     the node is created instead. The default value for this parameter is
#     localhost. The value for this parameter must be a properly formed host
#     name and must not contain spaces or characters that are not valid such as
#     the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#     deployment manager must also be available in conjunction with the dmgrPort
#     parameter.  Required if you are creating a managed profile with the
#     -federateLaterBPM parameter set to false and the deployment manager is not
#     on the local machine.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
#
# Parameter:  dmgrPort
#
# Description:
#     Identifies the SOAP port of the deployment manager. Specify this parameter
#     and the dmgrHost parameter to federate a custom profile as it is created
#     or augmented. The deployment manager must be running and accessible.  If
#     you have changed the default JMX connector type, you cannot federate with
#     the manageprofiles command-line utility. Use the addNode command later
#     after the node is created instead. The default value for this parameter is
#     8879. The port that you indicate must be a positive integer and a
#     connection to the deployment manager must be available in conjunction with
#     the dmgrHost parameter. Required if you are creating a managed profile,
#     the -federateLaterBPM parameter is set to false, and the deployment
#     manager was configured with a port that was not a default port.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password


###############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
###############################################################################
environmentType=Test


###############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
```

```
#      non-production server prior to deploying the applications on their
#      production application servers.  If -isDeveloperServer is set when
#      creating an IBM Business Process Manager profile, then a preconfigured VMM
#      file repository is installed. This file repository contains a sample
#      organization that can be used to test Business Process Choreographer
#      people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
##############################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Center custom profile for SQL:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a custom profile for a SQL database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# **************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# **************************************************************************
# END COPYRIGHT

##############################################################################
# These response file has the applicable parameters for creating a
# Managed PC Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PC_Std_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
##############################################################################

##############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
##############################################################################
create

##############################################################################
# Parameter:  templatePath
```

```
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/managed.procctr

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=Custom01

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
##############################################################################
cellName=cell_name
nodeName=node_name

##############################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
##############################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

##############################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
```

```
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


#############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
```

```
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
```

```
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
###############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


###############################################################################
# Parameter:  dbWinAuth
#
# Description:
#     An optional parameter that is valid in all default profile templates.
#     Specify true to indicate that you are running SQL Server using integrated
#     windows authentication.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
###############################################################################
#dbWinAuth=true


###############################################################################
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.   If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#     false
#     true
# Default Values:
#     true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.   The
#     host name can be the long or short DNS name or the IP address of the
#     deployment manager workstation. Specifying this optional parameter directs
#     the manageprofiles command-line utility to attempt to federate the custom
#     node into the deployment manager cell as it creates the custom profile.
#     This parameter is ignored when creating a deployment manager profile or
#     stand-alone server profile. If you federate a custom node when the
#     deployment manager is not running, the installation indicator in the logs
#     is INSTCONFFAILED to indicate a complete failure. The resulting custom
```

```
#       profile is unusable. You must move the custom profile directory out of the
#       profile repository (the profile's installation root directory) before
#       creating another custom profile with the same profile name. If you have
#       changed the default JMX connector type, you cannot federate with the
#       manageprofiles command-line utility. Use the addNode command later after
#       the node is created instead. The default value for this parameter is
#       localhost. The value for this parameter must be a properly formed host
#       name and must not contain spaces or characters that are not valid such as
#       the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#       deployment manager must also be available in conjunction with the dmgrPort
#       parameter.  Required if you are creating a managed profile with the
#       -federateLaterBPM parameter set to false and the deployment manager is not
#       on the local machine.
#
# Valid Values:
#       Varies
# Default Values:
#       None
#
# Parameter:  dmgrPort
#
# Description:
#       Identifies the SOAP port of the deployment manager. Specify this parameter
#       and the dmgrHost parameter to federate a custom profile as it is created
#       or augmented. The deployment manager must be running and accessible.  If
#       you have changed the default JMX connector type, you cannot federate with
#       the manageprofiles command-line utility. Use the addNode command later
#       after the node is created instead. The default value for this parameter is
#       8879. The port that you indicate must be a positive integer and a
#       connection to the deployment manager must be available in conjunction with
#       the dmgrHost parameter. Required if you are creating a managed profile,
#       the -federateLaterBPM parameter is set to false, and the deployment
#       manager was configured with a port that was not a default port.
#
# Valid Values:
#       Varies
# Default Values:
#       None
################################################################################
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password


################################################################################
# Parameter:  environmentType
#
# Description:
#       Specifies the environment type. The environment type refers to how IBM
#       Business Process Manager is used; for example, in a production, stage or
#       test environment. Load testing might be done on a test server, while a
#       stage environment type might be used as a temporary location to host
#       changes before putting those changes into production. You might specify
#       Stage as the Environment type if the server you are configuring will be
#       accessed and used to review content and new functionality.  Valid values
#       are as follows:
#       Test
#       Use Test if the server you are configuring is to be used as a testing
#       environment.
#       Stage
#       Use Stage if the server is to serve as a staging platform to be used as a
#       preproduction server.
#       Production
#       Use Production if the server is to serve in a production capacity.
#       The default value is Test.
#
```

```
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
#############################################################################
environmentType=Test

#############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#############################################################################
#isDeveloperServer=false
```

Creating Process Server deployment manager and custom profiles using manageprofiles:

Use the **manageprofiles** command-line utility to create Process Server deployment manager and custom profiles for a network deployment configuration.

Before you run the **manageprofiles** command-line utility ensure that you have completed the following tasks:

- You have reviewed the full list of prerequisites for creating or augmenting a profile
- You have reviewed example profile creation commands
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

This task describes how to use the **manageprofiles** command-line utility to create deployment manager and custom profiles for a network deployment configuration. Use this task for Process Server deployment manager and custom profiles for a network deployment configuration.

To use the **manageprofiles** command-line utility to create the profiles, perform the following steps.
1. Determine the kind of profile you want to create, which in turn determines the template to use for your new profile (using the **-templatePath** option).

   Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates.

   The following templates are available:
   - dmgr.procsvr: for a deployment manager profile, which defines a deployment manager. A *deployment manager* provides one administrative interface to a logical group of servers on one or more workstations.
   - managed.procsvr: for a Process Server custom profile, which, when federated to a deployment manager, defines a managed node. If you have decided that your solution requires a network deployment configuration, your runtime environment requires one or more managed nodes. A

custom profile contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node. Do not federate a node unless the deployment manager you are federating to is at a release level the same or higher than that of the custom profile you are creating. Also, : Process Server profiles cannot use a WebSphere Enterprise Service Bus deployment manager, but WebSphere Enterprise Service Bus profiles can use a : Process Server deployment manager.

- `managed.procsvr.adv`: for Process Server custom profiles which, when federated to a deployment manager, define a managed node with capabilities and functionality specific to an IBM BPM Advanced configuration.
- `dmgr.esbserver`: for a WebSphere Enterprise Service Bus deployment manager profile, which defines a deployment manager.
- `managed.esbserver`: for a WebSphere Enterprise Service Bus custom profile, which, when federated to a deployment manager, defines a managed node. Do not federate a node unless the deployment manager you are federating to is at a release level the same or higher than that of the custom profile you are creating. WebSphere Enterprise Service Bus profiles can use a Enterprise Service Bus or IBM Business Process Manager Process Server deployment manager.

2. Determine which parameters are required for your type of profile by reviewing the example profile creation commands in manageprofile examples.

3. Determine the values that you want to supply for the profile by reviewing the default values in the manageprofiles parameters topic to see if they are what you need for your profile.

4. Create a response file.

   Sample response files are available under the `install_root/BPM/samples/manageprofiles/` directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server) and customize the file for your environment.

   Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. See Creating a response file using the IBM Business Process Manager samples.

5. Run the **manageprofiles** command from the command line, using the **-response** parameter and the name of the response file that you created.

   - Linux   *UNIX*   **manageprofiles.sh -response** *myResponseFile*

   - Windows   **manageprofiles.bat -response** *myResponseFile*

   The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

Related concepts:

"Prerequisites for creating or augmenting profiles" on page 131
Before creating or augmenting a profile, you must ensure that a series of prerequisites have been met.

"**manageprofiles** examples" on page 502
The examples in this section show how to create stand-alone, deployment manager, and custom (managed node) profiles using the **manageprofiles** command-line utility.

Response file for a Standard Process Server:

Use the **manageprofiles** command-line utility and the response file to create a deployment manager profile.

Response file for creating a Standard Process Server deployment manager profile for DB2:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a DB2 database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

##############################################################################
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMgr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
##############################################################################

##############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
##############################################################################
create

##############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
templatePath=BPM/dmgr.procsvr

##############################################################################
```

```
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=Dmgr01

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
```

```
#      where NodeNumber is a sequential number starting at 01.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
################################################################################
cellName=cell_name
nodeName=node_name

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path
```

```
################################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
```

```
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


##############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
```

```
#      Use Test if the server you are configuring is to be used as a testing
#      environment.
#      Stage
#      Use Stage if the server is to serve as a staging platform to be used as a
#      preproduction server.
#      Production
#      Use Production if the server is to serve in a production capacity.
#      The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
##############################################################################
environmentType=Test

##############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#      Specifies whether the server is intended for development purposes only.
#      This parameter is useful when creating profiles to test applications on a
#      non-production server prior to deploying the applications on their
#      production application servers.  If -isDeveloperServer is set when
#      creating an IBM Business Process Manager profile, then a preconfigured VMM
#      file repository is installed. This file repository contains a sample
#      organization that can be used to test Business Process Choreographer
#      people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##############################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Server deployment manager profile for DB2 for z/OS:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a DB2 for z/OS database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# **************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# **************************************************************************
# END COPYRIGHT

##############################################################################
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses DB2zOS.
```

```
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMgr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
################################################################################

################################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
################################################################################
create

################################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/dmgr.procsvr

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
```

```
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
profileName=Dmgr01

###############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
###############################################################################
cellName=cell_name
nodeName=node_name

###############################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
```

```
#     true
# Default Values:
#     true
##############################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

##############################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
```

```
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


################################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
```

```
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
```

```
#      parameter is false.
#
# Valid Values:
#      false
#      true
# Default Values:
#      false
#
# Parameter:  winserviceStartupType
#
# Description:
#      The following values for Windows service startup can be used:
#      manual
#      automatic
#      disabled
#      The default value for this parameter is manual.
#
# Valid Values:
#      manual
#      automatic
#      disabled
# Default Values:
#      manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

##############################################################################
# Parameter:  environmentType
#
# Description:
#      Specifies the environment type. The environment type refers to how IBM
#      Business Process Manager is used; for example, in a production, stage or
#      test environment. Load testing might be done on a test server, while a
#      stage environment type might be used as a temporary location to host
#      changes before putting those changes into production. You might specify
#      Stage as the Environment type if the server you are configuring will be
#      accessed and used to review content and new functionality.  Valid values
#      are as follows:
#      Test
#      Use Test if the server you are configuring is to be used as a testing
#      environment.
#      Stage
#      Use Stage if the server is to serve as a staging platform to be used as a
#      preproduction server.
#      Production
#      Use Production if the server is to serve in a production capacity.
#      The default value is Test.
#
# Valid Values:
#      Test
#      Production
#      Stage
# Default Values:
#      Test
##############################################################################
environmentType=Test

##############################################################################
# Parameter:  isDeveloperServer
#
# Description:
```

```
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
################################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Server deployment manager profile for Oracle:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for an Oracle database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ****************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ****************************************************************************
# END COPYRIGHT

################################################################################
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMgr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
################################################################################


################################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
################################################################################
create
```

```
################################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/dmgr.procsvr

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=Dmgr01

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
```

```
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#      Varies
# Default Values:
#      Varies
#
# Parameter:  nodeName
#
# Description:
#      Specifies the node name for the node that is created with the new profile.
#      Use a unique value within the cell or on the workstation. Each profile
#      that shares the same set of product binaries must have a unique node name.
#      This parameter is required for profile creation only with the
#      dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#      procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#      The default value for this parameter is based on the short host name,
#      profile type, and a trailing number, for example:
#      if (DMgr)
#              shortHostNameCellManagerNodeNumber
#      else
#              shortHostNameNodeNodeNumber
#      where NodeNumber is a sequential number starting at 01.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#      Varies
# Default Values:
#      Varies
################################################################################
cellName=cell_name
nodeName=node_name

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#      true
# Default Values:
#      true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

################################################################################
# Parameter:  signingCertDN
#
# Description:
#      Specifies the distinguished name of the root signing certificate that you
#      create when you create the profile. Specify the distinguished name in
#      quotation marks. This default personal certificate is located in the
#      server keystore file. If you do not specifically create or import a root
#      signing certificate, one is created by default. See the
#      -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#      Varies
# Default Values:
```

```
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


################################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
```

```
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


################################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
```

```
#      localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
```

```
#    manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

##############################################################################
# Parameter:  dbDriverType
#
# Description:
#     The database driver type. For an Oracle database, specify ORACLE. For
#     databases other than Oracle, the value is automatically set based on the
#     server operating system. Server installations on all operating systems use
#     type 4.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
dbDriverType=ORACLE_THIN

##############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
##############################################################################
environmentType=Test

##############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
```

```
#       creating an IBM Business Process Manager profile, then a preconfigured VMM
#       file repository is installed. This file repository contains a sample
#       organization that can be used to test Business Process Choreographer
#       people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##########################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Server deployment manager profile for SQL:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a SQL database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# **************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# **************************************************************************
# END COPYRIGHT

##########################################################################
# These response file has the applicable parameters for creating a
# DMgr PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
##########################################################################

##########################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
##########################################################################
create

##########################################################################
# Parameter:  templatePath
#
# Description:
```

```
#      Specifies the directory path to the template files in the installation
#      root directory. Within the profileTemplates directory are various
#      directories that correspond to different profile types and that vary with
#      the type of product installed. The profile directories are the paths that
#      you indicate while using the -templatePath option.
#      Use absolute paths. This parameter must exist as a directory and point to
#      a valid template directory.
#
#      When using the -templatePath parameter, specify the fully qualified file
#      path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/dmgr.procsvr

################################################################################
# Parameter:   profileName
#
# Description:
#      Specifies the name of the profile. Use a unique value when creating a
#      profile.
#      Each profile that shares the same set of product binaries must have a
#      unique name. The default profile name is based on the profile type and a
#      trailing number, for example:
#      profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#      Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#      unique profile name. The value for this parameter must not contain spaces
#      or characters that are not valid such as the following: *, ?, ", <, >, ,,
#      /, \, and |. The profile name that you choose must not be in use. On
#      Windows platforms: If the fully qualified path contains spaces, enclose
#      the value in quotation marks.  The default value is based on the
#      install_root directory, the profiles subdirectory, and the name of the
#      file. For example, the default for profile creation is:
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#      file in the install_root/properties directory.  The value for this
#      parameter must be a valid path for the target system and must not be
#      currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=Dmgr01

################################################################################
# Parameter:   cellName
#
# Description:
#      Specifies the cell name of the profile. Use a unique cell name for each
#      profile.
#      The default value for this parameter is based on a combination of the
#      short host name, the constant Cell, and a trailing number, for example:
#      if (DMgr)
#         shortHostNameCellCellNumber
#      else
#         shortHostNameNodeNodeNumberCell
#      where CellNumber is a sequential number starting at 01 and NodeNumber is
#      the node number that you used to define the node name.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
################################################################################
cellName=cell_name
nodeName=node_name

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
```

```
# Description:
#    An optional parameter that specifies the amount of time in years that the
#    root signing certificate is valid. If you do not specify this parameter
#    with the -signingCertDN parameter, the root signing certificate is valid
#    for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#    Specifies the password to use on all keystore files created during profile
#    creation. Keystore files are created for the default personal certificate
#    and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


################################################################################
# Parameter:  defaultPorts
#
# Description:
#    Assigns the default or base port values to the profile.d o not use this
#    parameter when using the -startingPort or -portsFile parameter.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  startingPort
#
# Description:
#    Specifies the starting port number for generating and assigning all ports
#    for the profile.
#    Port values are assigned sequentially from the -startingPort value,
#    omitting those ports that are already in use. The system recognizes and
#    resolves ports that are currently in use and determines the port
#    assignments to avoid port conflicts.
#    Do not use this parameter with the -defaultPorts or -portsFile parameters.
#    During profile creation, the manageprofiles command-line utility uses an
#    automatically generated set of recommended ports if you do not specify the
#    -startingPort parameter, the -defaultPorts parameter or the -portsFile
#    parameter. The recommended port values can be different than the default
#    port values based on the availability of the default ports.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  portsFile
#
```

```
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


##############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
```

434 Installing

```
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
```

```
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


##############################################################################
# Parameter:  dbWinAuth
#
# Description:
#     An optional parameter that is valid in all default profile templates.
#     Specify true to indicate that you are running SQL Server using integrated
#     windows authentication.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
##############################################################################
#dbWinAuth=true

##############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
##############################################################################
environmentType=Test

##############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
```

```
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
###########################################################################
#isDeveloperServer=false
```

Response file for an Advanced Process Server:

Use the **manageprofiles** command-line utility and the response file to create a deployment manager profile.

Response file for creating an Advanced Process Server deployment manager profile for DB2:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a DB2 database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# *************************************************************************
# END COPYRIGHT

###########################################################################
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###########################################################################

###########################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
###########################################################################
create

###########################################################################
# Parameter:  templatePath
```

```
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/dmgr.procsvr.adv

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=Dmgr01

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
```

```
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#    Specifies the node name for the node that is created with the new profile.
#    Use a unique value within the cell or on the workstation. Each profile
#    that shares the same set of product binaries must have a unique node name.
#    This parameter is required for profile creation only with the
#    dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#    procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#    The default value for this parameter is based on the short host name,
#    profile type, and a trailing number, for example:
#    if (DMgr)
#            shortHostNameCellManagerNodeNumber
#    else
#            shortHostNameNodeNodeNumber
#    where NodeNumber is a sequential number starting at 01.
#    The value for this parameter must not contain spaces or any characters
#    that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
###############################################################################
cellName=cell_name
nodeName=node_name

###############################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#    For IBM Business Process Manager omit this parameter. Administrative
#    security is always enabled for IBM Business Process Manager profiles.
#    You must also specify the parameters -adminUserName and -adminPassword
#    along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
###############################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

###############################################################################
# Parameter:  signingCertDN
#
# Description:
#    Specifies the distinguished name of the root signing certificate that you
#    create when you create the profile. Specify the distinguished name in
#    quotation marks. This default personal certificate is located in the
#    server keystore file. If you do not specifically create or import a root
#    signing certificate, one is created by default. See the
#    -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
```

```
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
```

```
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


################################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
```

```
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
################################################################################
```

```
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

################################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#bpmdbDesign=design_file

################################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
################################################################################
dbCreateNew=true

################################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    DB2_DATASERVER
# Default Values:
#    DB2_DATASERVER
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
```

```
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbHostName
#
# Description:
#      The database server host name or IP address. The default value is the
#      local host name. Required when the -bpmdbDesign parameter was not
#      specified. (It is not required if the -bpmdbDesign parameter is
#      specified.)
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbServerPort
#
# Description:
#      The database server port number. Depending on the database you are using,
#      you can specify a different port number instead of the default port
#      number.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
dbType=DB2_DATASERVER
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2
dbHostName=db_host_name
dbServerPort=db_port_number

###############################################################################
# Parameter:  dbDelayConfig
#
# Description:
#      Indicates whether to postpone table creation until after the profile is
#      created. Valid values are true or false. The parameter is set to false by
#      default. Setting this parameter to true delays the execution of the
#      scripts that create the database tables. No. Specify false if you want the
#      database to be set up during profile creation.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
###############################################################################
dbDelayConfig=false

###############################################################################
# Parameter:  dbName
#
# Description:
#      The name of the database. By default, the value is set to orcl for Oracle
#      databases and to CMNDB for all other supported databases.
#
#      Note: Do not use this parameter if you are using -bpmdbDesign.
```

```
#
# Valid Values:
#    Varies
# Default Values:
#    CMNDB
#
# Parameter:  dbUserId
#
# Description:
#    The user ID for all database types. Specifies the user ID that has
#    privileges to create and drop the databases. The WebSphere data source
#    uses this ID to authenticate the database connection.
#    Important: The -dbUserId parameter value must be a valid database
#    authorization ID.  Required if the -bpmdbDesign file is not set.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPassword
#
# Description:
#    The password required for database authentication. Required when the
#    -bpmdbDesign file is not set.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd


################################################################################
# Parameter:  environmentType
#
# Description:
#    Specifies the environment type. The environment type refers to how IBM
#    Business Process Manager is used; for example, in a production, stage or
#    test environment. Load testing might be done on a test server, while a
#    stage environment type might be used as a temporary location to host
#    changes before putting those changes into production. You might specify
#    Stage as the Environment type if the server you are configuring will be
#    accessed and used to review content and new functionality.  Valid values
#    are as follows:
#    Test
#    Use Test if the server you are configuring is to be used as a testing
#    environment.
#    Stage
#    Use Stage if the server is to serve as a staging platform to be used as a
#    preproduction server.
#    Production
#    Use Production if the server is to serve in a production capacity.
#    The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
```

```
###########################################################################
environmentType=Test

###########################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
###########################################################################
#isDeveloperServer=false
```

Response file for creating an Advanced Process Server deployment manager profile for DB2 for z/OS:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a DB2 for z/OS database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# **************************************************************************
#
#   Licensed Materials - Property of IBM
#   5725-C94
#   (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#   This sample program is provided AS IS and may be used, executed, copied
#   and modified without royalty payment by customer (a) for its own
#   instruction and study, (b) in order to develop applications designed to
#   run with an IBM WebSphere product, either for customer's own internal
#   use or for redistribution by customer, as part of such an application,
#   in customer's own products.
#
# **************************************************************************
# END COPYRIGHT

###########################################################################
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
###########################################################################

###########################################################################
# Parameter:  create
```

```
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
#############################################################################
create

#############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
templatePath=BPM/dmgr.procsvr.adv

#############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
profileName=Dmgr01

#############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
```

```
#      short host name, the constant Cell, and a trailing number, for example:
#      if (DMgr)
#          shortHostNameCellCellNumber
#      else
#          shortHostNameNodeNodeNumberCell
#      where CellNumber is a sequential number starting at 01 and NodeNumber is
#      the node number that you used to define the node name.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#      Specifies the node name for the node that is created with the new profile.
#      Use a unique value within the cell or on the workstation. Each profile
#      that shares the same set of product binaries must have a unique node name.
#      This parameter is required for profile creation only with the
#      dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#      procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#      The default value for this parameter is based on the short host name,
#      profile type, and a trailing number, for example:
#      if (DMgr)
#              shortHostNameCellManagerNodeNumber
#      else
#              shortHostNameNodeNodeNumber
#      where NodeNumber is a sequential number starting at 01.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
################################################################################
cellName=cell_name
nodeName=node_name


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#      Specifies the distinguished name of the root signing certificate that you
#      create when you create the profile. Specify the distinguished name in
#      quotation marks. This default personal certificate is located in the
```

```
#       server keystore file. If you do not specifically create or import a root
#       signing certificate, one is created by default. See the
#       -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:   signingCertValidityPeriod
#
# Description:
#       An optional parameter that specifies the amount of time in years that the
#       root signing certificate is valid. If you do not specify this parameter
#       with the -signingCertDN parameter, the root signing certificate is valid
#       for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:   keyStorePassword
#
# Description:
#       Specifies the password to use on all keystore files created during profile
#       creation. Keystore files are created for the default personal certificate
#       and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


################################################################################
# Parameter:   defaultPorts
#
# Description:
#       Assigns the default or base port values to the profile.d o not use this
#       parameter when using the -startingPort or -portsFile parameter.
#       During profile creation, the manageprofiles command-line utility uses an
#       automatically generated set of recommended ports if you do not specify the
#       -startingPort parameter, the -defaultPorts parameter or the -portsFile
#       parameter. The recommended port values can be different than the default
#       port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:   startingPort
#
# Description:
#       Specifies the starting port number for generating and assigning all ports
#       for the profile.
#       Port values are assigned sequentially from the -startingPort value,
#       omitting those ports that are already in use. The system recognizes and
#       resolves ports that are currently in use and determines the port
#       assignments to avoid port conflicts.
#       Do not use this parameter with the -defaultPorts or -portsFile parameters.
#       During profile creation, the manageprofiles command-line utility uses an
#       automatically generated set of recommended ports if you do not specify the
```

```
#      -startingPort parameter, the -defaultPorts parameter or the -portsFile
#      parameter. The recommended port values can be different than the default
#      port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#      An optional parameter that specifies the path to a file that defines port
#      settings for the new profile.
#      Do not use this parameter when using the -startingPort or -defaultPorts
#      parameter.
#      During profile creation, the manageprofiles command-line utility uses an
#      automatically generated set of recommended ports if you do not specify the
#      -startingPort parameter, the -defaultPorts parameter or the -portsFile
#      parameter. The recommended port values can be different than the default
#      port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


###############################################################################
# Parameter:  enableService
#
# Description:
#      Enables the creation of a Linux service. Valid values include true or
#      false. The default value for this parameter is false.  When the
#      manageprofiles command-line utility is run with the -enableService option
#      set to true , the Linux service is created with the profile when the
#      command is run by the root user. When a nonroot user runs the
#      manageprofiles command-line utility, the profile is created, but the Linux
#      service is not. The Linux service is not created because the nonroot user
#      does not have sufficient permission to set up the service. An
#      INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#      creation and the profile creation log
#      install_root/logs/manageprofiles/profile_name_create.log contains a
#      message indicating the current user does not have sufficient permission to
#      set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#      Specifies the user ID that is used during the creation of the Linux
#      service so that the Linux service will run under this user ID. The Linux
#      service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
```

```
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
```

```
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#bpmdbDesign=design_file

################################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
################################################################################
dbCreateNew=false

################################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     DB2UDBOS390
```

452 Installing

```
# Default Values:
#     DB2UDBOS390
############################################################################
dbType=DB2UDBOS390

############################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
############################################################################
dbDelayConfig=true

############################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     CMNDB
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
############################################################################
#dbName=CMNDB
```

```
    dbUserId=db_userid
    dbPassword=db_pswd

    ############################################################################
    # Parameter:  dbStorageGroup
    #
    # Description:
    #     The storage group name for DB2 z/OS databases.
    #
    #     Note: Do not use this parameter if you are using -bpmdbDesign.
    #
    # Valid Values:
    #     Varies
    # Default Values:
    #     None
    #
    # Parameter:  dbConnectionLocation
    #
    # Description:
    #     The location of DB2 for z/OS database.
    #
    #     Note: Do not use this parameter if you are using -bpmdbDesign.
    #
    # Valid Values:
    #     Varies
    # Default Values:
    #     None
    ############################################################################
    dbStorageGroup=db_stg_group
    dbConnectionLocation=db2_location

    ############################################################################
    # Parameter:  environmentType
    #
    # Description:
    #     Specifies the environment type. The environment type refers to how IBM
    #     Business Process Manager is used; for example, in a production, stage or
    #     test environment. Load testing might be done on a test server, while a
    #     stage environment type might be used as a temporary location to host
    #     changes before putting those changes into production. You might specify
    #     Stage as the Environment type if the server you are configuring will be
    #     accessed and used to review content and new functionality.  Valid values
    #     are as follows:
    #     Test
    #     Use Test if the server you are configuring is to be used as a testing
    #     environment.
    #     Stage
    #     Use Stage if the server is to serve as a staging platform to be used as a
    #     preproduction server.
    #     Production
    #     Use Production if the server is to serve in a production capacity.
    #     The default value is Test.
    #
    # Valid Values:
    #     Test
    #     Production
    #     Stage
    # Default Values:
    #     Test
    ############################################################################
    environmentType=Test

    ############################################################################
    # Parameter:  isDeveloperServer
    #
    # Description:
    #     Specifies whether the server is intended for development purposes only.
```

```
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
##############################################################################
#isDeveloperServer=false
```

Response file for creating an Advanced Process Server deployment manager profile for Oracle:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for an Oracle database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

##############################################################################
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
##############################################################################

##############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
##############################################################################
create

##############################################################################
```

```
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
templatePath=BPM/dmgr.procsvr.adv

###############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
profileName=Dmgr01

###############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
```

```
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
################################################################################
cellName=cell_name
nodeName=node_name

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
```

```
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


##############################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
```

```
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
##############################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


##############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
```

```
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#    Specify your user ID so that the Windows operating system can verify you
#    as an ID that is capable of creating a Windows service. Your user ID must
#    belong to the Administrator group and have the following advanced user
#    rights:
#    Act as part of the operating system
#    Log on as a service
#    The default value for this parameter is the current user name. The value
#    for this parameter must not contain spaces or characters that are not
#    valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#    you specify must have the proper permissions to create a Windows service.
#    You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#    Specify the password for the specified user or the local account that is
#    to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#    The value can be either true or false. Specify true to create a Windows
#    service for the server process that is created within the profile. Specify
#    false to not create the Windows service. The default value for this
#    parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#    The following values for Windows service startup can be used:
#    manual
#    automatic
#    disabled
#    The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
```

```
#############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

#############################################################################
# Parameter:  bpmdbDesign
#
# Description:
#     Specifies the file that holds the database configuration design for all of
#     the IBM Business Process Manager components .
#     This parameter replaces the -wbidbDesign that was used in previous
#     versions of WebSphere Process Server.
#
#     Note: If a bpmdbDesign file parameter is specified, none of the database
#     related properties should be specified.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
#bpmdbDesign=design_file

#############################################################################
# Parameter:  dbCreateNew
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#     false
# Default Values:
#     false
#############################################################################
dbCreateNew=false

#############################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     ORACLE
# Default Values:
#     ORACLE
#
# Parameter:  dbDriverType
#
# Description:
#     The database driver type. For an Oracle database, specify ORACLE. For
#     databases other than Oracle, the value is automatically set based on the
```

```
#     server operating system. Server installations on all operating systems use
#     type 4.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbType=ORACLE
dbDriverType=ORACLE_THIN
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/Oracle

################################################################################
# Parameter:  dbLocation
#
# Description:
#     The directory of the ORACLE_HOME directory. This parameter is required
#     when the parameter dbDelayConfig is set to false. No, this parameter is
#     only valid if you are using Oracle databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbLocation=db_location

################################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
################################################################################
dbDelayConfig=false

################################################################################
# Parameter:  dbName
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
```

```
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    CMNDB
#
# Parameter:  dbUserId
#
# Description:
#    The user ID for all database types. Specifies the user ID that has
#    privileges to create and drop the databases. The WebSphere data source
#    uses this ID to authenticate the database connection.
#    Important: The -dbUserId parameter value must be a valid database
#    authorization ID.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPassword
#
# Description:
#    The password required for database authentication.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###############################################################################
#dbName=orcl
#dbUserId=db_userid
#dbPassword=db_pswd


###############################################################################
# Parameter:  dbCommonUserId
#
# Description:
#    The user id to authenticate with the Common database.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCommonPassword
#
# Description:
#    The password for the user id of the dbCommonUserId parameter.
#
#    Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###############################################################################
dbCommonUserId=common_db_userid
dbCommonPassword=common_db_pswd
```

```
###############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
###############################################################################
environmentType=Test

###############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
###############################################################################
#isDeveloperServer=false
```

Response file for creating an Advanced Process Server deployment manager profile for SQL:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a deployment manager profile for a SQL database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# *************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
```

```
#   (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#   This sample program is provided AS IS and may be used, executed, copied
#   and modified without royalty payment by customer (a) for its own
#   instruction and study, (b) in order to develop applications designed to
#   run with an IBM WebSphere product, either for customer's own internal
#   use or for redistribution by customer, as part of such an application,
#   in customer's own products.
#
# *************************************************************************
# END COPYRIGHT

#############################################################################
# These response file has the applicable parameters for creating a
# DMgr PS Adv profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Adv_DMgr_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#############################################################################

#############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
#############################################################################
create

#############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#############################################################################
templatePath=BPM/dmgr.procsvr.adv

#############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
```

```
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=Dmgr01

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
```

466   Installing

```
################################################################################
cellName=cell_name
nodeName=node_name

################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password
```

```
###########################################################################
# Parameter:  defaultPorts
#
# Description:
#     Assigns the default or base port values to the profile.d o not use this
#     parameter when using the -startingPort or -portsFile parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  startingPort
#
# Description:
#     Specifies the starting port number for generating and assigning all ports
#     for the profile.
#     Port values are assigned sequentially from the -startingPort value,
#     omitting those ports that are already in use. The system recognizes and
#     resolves ports that are currently in use and determines the port
#     assignments to avoid port conflicts.
#     Do not use this parameter with the -defaultPorts or -portsFile parameters.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  portsFile
#
# Description:
#     An optional parameter that specifies the path to a file that defines port
#     settings for the new profile.
#     Do not use this parameter when using the -startingPort or -defaultPorts
#     parameter.
#     During profile creation, the manageprofiles command-line utility uses an
#     automatically generated set of recommended ports if you do not specify the
#     -startingPort parameter, the -defaultPorts parameter or the -portsFile
#     parameter. The recommended port values can be different than the default
#     port values based on the availability of the default ports.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###########################################################################
#defaultPorts=default_ports
#startingPort=startingPort
#portsFile=file_path


###########################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
```

```
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
```

```
#      Specify the password for the specified user or the local account that is
#      to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#      The value can be either true or false. Specify true to create a Windows
#      service for the server process that is created within the profile. Specify
#      false to not create the Windows service. The default value for this
#      parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#      The following values for Windows service startup can be used:
#      manual
#      automatic
#      disabled
#      The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
#    manual
###############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


###############################################################################
# Parameter:  bpmdbDesign
#
# Description:
#      Specifies the file that holds the database configuration design for all of
#      the IBM Business Process Manager components .
#      This parameter replaces the -wbidbDesign that was used in previous
#      versions of WebSphere Process Server.
#
#      Note: If a bpmdbDesign file parameter is specified, none of the database
#      related properties should be specified.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###############################################################################
#bpmdbDesign=design_file


###############################################################################
# Parameter:  dbCreateNew
```

```
#
# Description:
#     Indicates a new database is created or if an existing database is to be
#     reused. Valid values are true or false. This value cannot be true for an
#     Oracle, SQL Server or a DB2 z/OS database. This value can only be true for
#     a DB2 database.
#
# Valid Values:
#    false
# Default Values:
#    false
################################################################################
dbCreateNew=false


################################################################################
# Parameter:  dbType
#
# Description:
#     The database type. Set one of the following values for the type of
#     database product you are using with IBM Business Process Manager.
#
#     Required only for one of the following:
#     A stand-alone profile when -bpmdbDesign was not specified.
#     An advanced dmgr or managed profiles when -bpmdbDesign was not specified.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    MSSQLSERVER_MICROSOFT
# Default Values:
#    MSSQLSERVER_MICROSOFT
#
# Parameter:  dbJDBCClasspath
#
# Description:
#     The directory path location of JDBC driver files.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
dbType=MSSQLSERVER_MICROSOFT
dbJDBCClasspath=${WAS_INSTALL_ROOT}/jdbcdrivers/SQLServer


################################################################################
# Parameter:  dbDelayConfig
#
# Description:
#     Indicates whether to postpone table creation until after the profile is
#     created. Valid values are true or false. The parameter is set to false by
#     default. Setting this parameter to true delays the execution of the
#     scripts that create the database tables. No. Specify false if you want the
#     database to be set up during profile creation.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
################################################################################
dbDelayConfig=false


################################################################################
# Parameter:  dbName
```

```
#
# Description:
#     The name of the database. By default, the value is set to orcl for Oracle
#     databases and to CMNDB for all other supported databases.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    CMNDB
#
# Parameter:  dbUserId
#
# Description:
#     The user ID for all database types. Specifies the user ID that has
#     privileges to create and drop the databases. The WebSphere data source
#     uses this ID to authenticate the database connection.
#     Important: The -dbUserId parameter value must be a valid database
#     authorization ID.  Required if the -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbPassword
#
# Description:
#     The password required for database authentication. Required when the
#     -bpmdbDesign file is not set.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###############################################################################
#dbName=CMNDB
dbUserId=db_userid
dbPassword=db_pswd

###############################################################################
# Parameter:  dbCommonUserId
#
# Description:
#     The user id to authenticate with the Common database.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dbCommonPassword
#
# Description:
#     The password for the user id of the dbCommonUserId parameter.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#    Varies
```

```
# Default Values:
#     None
##############################################################################
#dbCommonUserId=common_db_userid
#dbCommonPassword=common_db_pswd

##############################################################################
# Parameter:  dbWinAuth
#
# Description:
#     An optional parameter that is valid in all default profile templates.
#     Specify true to indicate that you are running SQL Server using integrated
#     windows authentication.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
##############################################################################
#dbWinAuth=true

##############################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
##############################################################################
environmentType=Test

##############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
```

```
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
################################################################################
#isDeveloperServer=false
```

Response file for a Standard Process Server:

Use the **manageprofiles** command-line utility and the response file to create a custom profile.

Response file for creating a Standard Process Server custom profile for DB2:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a custom profile for a DB2 database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ****************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ****************************************************************************
# END COPYRIGHT

################################################################################
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses DB2.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
################################################################################

################################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
################################################################################
create

################################################################################
# Parameter:  templatePath
#
# Description:
```

```
#       Specifies the directory path to the template files in the installation
#       root directory. Within the profileTemplates directory are various
#       directories that correspond to different profile types and that vary with
#       the type of product installed. The profile directories are the paths that
#       you indicate while using the -templatePath option.
#       Use absolute paths. This parameter must exist as a directory and point to
#       a valid template directory.
#
#       When using the -templatePath parameter, specify the fully qualified file
#       path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
templatePath=BPM/managed.procsvr

################################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
profileName=Custom01

################################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
```

```
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
###############################################################################
cellName=cell_name
nodeName=node_name

###############################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
###############################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword

###############################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
```

```
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

################################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
```

```
#      profile. Valid values include specifieduser or localsystem. The
#      localsystem value runs the Windows service under the local account of the
#      user who creates the profile. The default value for this parameter is
#      localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#    manual
```

```
#     automatic
#     disabled
# Default Values:
#     manual
################################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


################################################################################
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.   If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#     false
#     true
# Default Values:
#     true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.   The
#     host name can be the long or short DNS name or the IP address of the
#     deployment manager workstation. Specifying this optional parameter directs
#     the manageprofiles command-line utility to attempt to federate the custom
#     node into the deployment manager cell as it creates the custom profile.
#     This parameter is ignored when creating a deployment manager profile or
#     stand-alone server profile. If you federate a custom node when the
#     deployment manager is not running, the installation indicator in the logs
#     is INSTCONFFAILED to indicate a complete failure. The resulting custom
#     profile is unusable. You must move the custom profile directory out of the
#     profile repository (the profile's installation root directory) before
#     creating another custom profile with the same profile name. If you have
#     changed the default JMX connector type, you cannot federate with the
#     manageprofiles command-line utility. Use the addNode command later after
#     the node is created instead. The default value for this parameter is
#     localhost. The value for this parameter must be a properly formed host
#     name and must not contain spaces or characters that are not valid such as
#     the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#     deployment manager must also be available in conjunction with the dmgrPort
#     parameter.  Required if you are creating a managed profile with the
#     -federateLaterBPM parameter set to false and the deployment manager is not
#     on the local machine.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  dmgrPort
```

```
#
# Description:
#     Identifies the SOAP port of the deployment manager. Specify this parameter
#     and the dmgrHost parameter to federate a custom profile as it is created
#     or augmented. The deployment manager must be running and accessible.  If
#     you have changed the default JMX connector type, you cannot federate with
#     the manageprofiles command-line utility. Use the addNode command later
#     after the node is created instead. The default value for this parameter is
#     8879. The port that you indicate must be a positive integer and a
#     connection to the deployment manager must be available in conjunction with
#     the dmgrHost parameter. Required if you are creating a managed profile,
#     the -federateLaterBPM parameter is set to false, and the deployment
#     manager was configured with a port that was not a default port.
#
# Valid Values:
#     Varies
# Default Values:
#     None
################################################################################
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password


################################################################################
# Parameter:  environmentType
#
# Description:
#     Specifies the environment type. The environment type refers to how IBM
#     Business Process Manager is used; for example, in a production, stage or
#     test environment. Load testing might be done on a test server, while a
#     stage environment type might be used as a temporary location to host
#     changes before putting those changes into production. You might specify
#     Stage as the Environment type if the server you are configuring will be
#     accessed and used to review content and new functionality.  Valid values
#     are as follows:
#     Test
#     Use Test if the server you are configuring is to be used as a testing
#     environment.
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#     Test
#     Production
#     Stage
# Default Values:
#     Test
################################################################################
environmentType=Test

################################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
```

```
#      people resolution, ready for you to use as is.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
##############################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Server custom profile for DB2 for z/OS:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a custom profile for a DB2 for z/OS database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ****************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ****************************************************************************
# END COPYRIGHT

##############################################################################
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses DB2zOS.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_DB2zOS.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
##############################################################################

##############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
##############################################################################
create

##############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
```

```
#      the type of product installed. The profile directories are the paths that
#      you indicate while using the -templatePath option.
#      Use absolute paths. This parameter must exist as a directory and point to
#      a valid template directory.
#
#      When using the -templatePath parameter, specify the fully qualified file
#      path for the parameter.
#
# Valid Values:
#      Varies
# Default Values:
#      None
##############################################################################
templatePath=BPM/managed.procsvr


##############################################################################
# Parameter:  profileName
#
# Description:
#      Specifies the name of the profile. Use a unique value when creating a
#      profile.
#      Each profile that shares the same set of product binaries must have a
#      unique name. The default profile name is based on the profile type and a
#      trailing number, for example:
#      profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#      Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#      unique profile name. The value for this parameter must not contain spaces
#      or characters that are not valid such as the following: *, ?, ", <, >, ,,
#      /, \, and |. The profile name that you choose must not be in use. On
#      Windows platforms: If the fully qualified path contains spaces, enclose
#      the value in quotation marks.  The default value is based on the
#      install_root directory, the profiles subdirectory, and the name of the
#      file. For example, the default for profile creation is:
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#      file in the install_root/properties directory.  The value for this
#      parameter must be a valid path for the target system and must not be
#      currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#      Varies
# Default Values:
#      None
##############################################################################
profileName=Custom01


##############################################################################
# Parameter:  cellName
#
# Description:
#      Specifies the cell name of the profile. Use a unique cell name for each
#      profile.
#      The default value for this parameter is based on a combination of the
#      short host name, the constant Cell, and a trailing number, for example:
#      if (DMgr)
#          shortHostNameCellCellNumber
#      else
#          shortHostNameNodeNodeNumberCell
#      where CellNumber is a sequential number starting at 01 and NodeNumber is
#      the node number that you used to define the node name.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#      Varies
# Default Values:
#      Varies
```

482 Installing

```
#
# Parameter:  nodeName
#
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
###############################################################################
cellName=cell_name
nodeName=node_name


###############################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
###############################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


###############################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
```

```
#      with the -signingCertDN parameter, the root signing certificate is valid
#      for 20 years.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#      Specifies the password to use on all keystore files created during profile
#      creation. Keystore files are created for the default personal certificate
#      and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
###############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password


###############################################################################
# Parameter:  enableService
#
# Description:
#      Enables the creation of a Linux service. Valid values include true or
#      false. The default value for this parameter is false.  When the
#      manageprofiles command-line utility is run with the -enableService option
#      set to true , the Linux service is created with the profile when the
#      command is run by the root user. When a nonroot user runs the
#      manageprofiles command-line utility, the profile is created, but the Linux
#      service is not. The Linux service is not created because the nonroot user
#      does not have sufficient permission to set up the service. An
#      INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#      creation and the profile creation log
#      install_root/logs/manageprofiles/profile_name_create.log contains a
#      message indicating the current user does not have sufficient permission to
#      set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#      Specifies the user ID that is used during the creation of the Linux
#      service so that the Linux service will run under this user ID. The Linux
#      service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#      The type of the owner account of the Windows service created for the
#      profile. Valid values include specifieduser or localsystem. The
#      localsystem value runs the Windows service under the local account of the
#      user who creates the profile. The default value for this parameter is
```

```
#      localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#    localsystem
#    specifieduser
# Default Values:
#    localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#    manual
#    automatic
#    disabled
# Default Values:
```

```
#     manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


##############################################################################
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.   If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#    false
#    true
# Default Values:
#    true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.  The
#     host name can be the long or short DNS name or the IP address of the
#     deployment manager workstation. Specifying this optional parameter directs
#     the manageprofiles command-line utility to attempt to federate the custom
#     node into the deployment manager cell as it creates the custom profile.
#     This parameter is ignored when creating a deployment manager profile or
#     stand-alone server profile. If you federate a custom node when the
#     deployment manager is not running, the installation indicator in the logs
#     is INSTCONFFAILED to indicate a complete failure. The resulting custom
#     profile is unusable. You must move the custom profile directory out of the
#     profile repository (the profile's installation root directory) before
#     creating another custom profile with the same profile name. If you have
#     changed the default JMX connector type, you cannot federate with the
#     manageprofiles command-line utility. Use the addNode command later after
#     the node is created instead. The default value for this parameter is
#     localhost. The value for this parameter must be a properly formed host
#     name and must not contain spaces or characters that are not valid such as
#     the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#     deployment manager must also be available in conjunction with the dmgrPort
#     parameter.  Required if you are creating a managed profile with the
#     -federateLaterBPM parameter set to false and the deployment manager is not
#     on the local machine.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dmgrPort
#
# Description:
#     Identifies the SOAP port of the deployment manager. Specify this parameter
```

```
#      and the dmgrHost parameter to federate a custom profile as it is created
#      or augmented. The deployment manager must be running and accessible.  If
#      you have changed the default JMX connector type, you cannot federate with
#      the manageprofiles command-line utility. Use the addNode command later
#      after the node is created instead. The default value for this parameter is
#      8879. The port that you indicate must be a positive integer and a
#      connection to the deployment manager must be available in conjunction with
#      the dmgrHost parameter. Required if you are creating a managed profile,
#      the -federateLaterBPM parameter is set to false, and the deployment
#      manager was configured with a port that was not a default port.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##############################################################################
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password


##############################################################################
# Parameter:  environmentType
#
# Description:
#    Specifies the environment type. The environment type refers to how IBM
#    Business Process Manager is used; for example, in a production, stage or
#    test environment. Load testing might be done on a test server, while a
#    stage environment type might be used as a temporary location to host
#    changes before putting those changes into production. You might specify
#    Stage as the Environment type if the server you are configuring will be
#    accessed and used to review content and new functionality.  Valid values
#    are as follows:
#    Test
#    Use Test if the server you are configuring is to be used as a testing
#    environment.
#    Stage
#    Use Stage if the server is to serve as a staging platform to be used as a
#    preproduction server.
#    Production
#    Use Production if the server is to serve in a production capacity.
#    The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
##############################################################################
environmentType=Test

##############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#    Specifies whether the server is intended for development purposes only.
#    This parameter is useful when creating profiles to test applications on a
#    non-production server prior to deploying the applications on their
#    production application servers.  If -isDeveloperServer is set when
#    creating an IBM Business Process Manager profile, then a preconfigured VMM
#    file repository is installed. This file repository contains a sample
#    organization that can be used to test Business Process Choreographer
#    people resolution, ready for you to use as is.
#
# Valid Values:
```

```
#     false
#     true
# Default Values:
#     false
############################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Server custom profile for Oracle:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a custom profile for an Oracle database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ****************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ****************************************************************************
# END COPYRIGHT

############################################################################
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses Oracle.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_Oracle.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
############################################################################

############################################################################
# Parameter:  create
#
# Description:
#     Creates the profile.
#     If you are creating a profile, this parameter is required.
############################################################################
create

############################################################################
# Parameter:  templatePath
#
# Description:
#     Specifies the directory path to the template files in the installation
#     root directory. Within the profileTemplates directory are various
#     directories that correspond to different profile types and that vary with
#     the type of product installed. The profile directories are the paths that
#     you indicate while using the -templatePath option.
#     Use absolute paths. This parameter must exist as a directory and point to
```

```
#     a valid template directory.
#
#     When using the -templatePath parameter, specify the fully qualified file
#     path for the parameter.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
templatePath=BPM/managed.procsvr

###############################################################################
# Parameter:  profileName
#
# Description:
#     Specifies the name of the profile. Use a unique value when creating a
#     profile.
#     Each profile that shares the same set of product binaries must have a
#     unique name. The default profile name is based on the profile type and a
#     trailing number, for example:
#     profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#     Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#     unique profile name. The value for this parameter must not contain spaces
#     or characters that are not valid such as the following: *, ?, ", <, >, ,,
#     /, \, and |. The profile name that you choose must not be in use. On
#     Windows platforms: If the fully qualified path contains spaces, enclose
#     the value in quotation marks.  The default value is based on the
#     install_root directory, the profiles subdirectory, and the name of the
#     file. For example, the default for profile creation is:
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#     WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#     file in the install_root/properties directory.  The value for this
#     parameter must be a valid path for the target system and must not be
#     currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
profileName=Custom01

###############################################################################
# Parameter:  cellName
#
# Description:
#     Specifies the cell name of the profile. Use a unique cell name for each
#     profile.
#     The default value for this parameter is based on a combination of the
#     short host name, the constant Cell, and a trailing number, for example:
#     if (DMgr)
#         shortHostNameCellCellNumber
#     else
#         shortHostNameNodeNodeNumberCell
#     where CellNumber is a sequential number starting at 01 and NodeNumber is
#     the node number that you used to define the node name.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
#
# Parameter:  nodeName
#
```

```
# Description:
#     Specifies the node name for the node that is created with the new profile.
#     Use a unique value within the cell or on the workstation. Each profile
#     that shares the same set of product binaries must have a unique node name.
#     This parameter is required for profile creation only with the
#     dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#     procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#     The default value for this parameter is based on the short host name,
#     profile type, and a trailing number, for example:
#     if (DMgr)
#             shortHostNameCellManagerNodeNumber
#     else
#             shortHostNameNodeNodeNumber
#     where NodeNumber is a sequential number starting at 01.
#     The value for this parameter must not contain spaces or any characters
#     that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#     Varies
# Default Values:
#     Varies
################################################################################
cellName=cell_name
nodeName=node_name


################################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#     For IBM Business Process Manager omit this parameter. Administrative
#     security is always enabled for IBM Business Process Manager profiles.
#     You must also specify the parameters -adminUserName and -adminPassword
#     along with the values for these parameters.
#
# Valid Values:
#     true
# Default Values:
#     true
################################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


################################################################################
# Parameter:  signingCertDN
#
# Description:
#     Specifies the distinguished name of the root signing certificate that you
#     create when you create the profile. Specify the distinguished name in
#     quotation marks. This default personal certificate is located in the
#     server keystore file. If you do not specifically create or import a root
#     signing certificate, one is created by default. See the
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
```

```
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  keyStorePassword
#
# Description:
#    Specifies the password to use on all keystore files created during profile
#    creation. Keystore files are created for the default personal certificate
#    and the root signing certificate.
#
# Valid Values:
#    Varies
# Default Values:
#    None
##############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

##############################################################################
# Parameter:  enableService
#
# Description:
#    Enables the creation of a Linux service. Valid values include true or
#    false. The default value for this parameter is false.  When the
#    manageprofiles command-line utility is run with the -enableService option
#    set to true , the Linux service is created with the profile when the
#    command is run by the root user. When a nonroot user runs the
#    manageprofiles command-line utility, the profile is created, but the Linux
#    service is not. The Linux service is not created because the nonroot user
#    does not have sufficient permission to set up the service. An
#    INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#    creation and the profile creation log
#    install_root/logs/manageprofiles/profile_name_create.log contains a
#    message indicating the current user does not have sufficient permission to
#    set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#    false
#    true
# Default Values:
#    None
#
# Parameter:  serviceUserName
#
# Description:
#    Specifies the user ID that is used during the creation of the Linux
#    service so that the Linux service will run under this user ID. The Linux
#    service runs whenever the user ID is logged on. Not required.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  winserviceAccountType
#
# Description:
#    The type of the owner account of the Windows service created for the
#    profile. Valid values include specifieduser or localsystem. The
#    localsystem value runs the Windows service under the local account of the
#    user who creates the profile. The default value for this parameter is
#    localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
```

```
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
###########################################################################
#enableService=true
```

```
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual

###############################################################################
# Parameter:  dbDriverType
#
# Description:
#     The database driver type. For an Oracle database, specify ORACLE. For
#     databases other than Oracle, the value is automatically set based on the
#     server operating system. Server installations on all operating systems use
#     type 4.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
dbDriverType=ORACLE_THIN

###############################################################################
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.   If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#     false
#     true
# Default Values:
#     true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.  The
#     host name can be the long or short DNS name or the IP address of the
#     deployment manager workstation. Specifying this optional parameter directs
#     the manageprofiles command-line utility to attempt to federate the custom
#     node into the deployment manager cell as it creates the custom profile.
#     This parameter is ignored when creating a deployment manager profile or
#     stand-alone server profile. If you federate a custom node when the
#     deployment manager is not running, the installation indicator in the logs
#     is INSTCONFFAILED to indicate a complete failure. The resulting custom
#     profile is unusable. You must move the custom profile directory out of the
#     profile repository (the profile's installation root directory) before
#     creating another custom profile with the same profile name. If you have
#     changed the default JMX connector type, you cannot federate with the
#     manageprofiles command-line utility. Use the addNode command later after
#     the node is created instead. The default value for this parameter is
#     localhost. The value for this parameter must be a properly formed host
#     name and must not contain spaces or characters that are not valid such as
```

```
#      the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#      deployment manager must also be available in conjunction with the dmgrPort
#      parameter.  Required if you are creating a managed profile with the
#      -federateLaterBPM parameter set to false and the deployment manager is not
#      on the local machine.
#
# Valid Values:
#    Varies
# Default Values:
#    None
#
# Parameter:  dmgrPort
#
# Description:
#      Identifies the SOAP port of the deployment manager. Specify this parameter
#      and the dmgrHost parameter to federate a custom profile as it is created
#      or augmented. The deployment manager must be running and accessible.  If
#      you have changed the default JMX connector type, you cannot federate with
#      the manageprofiles command-line utility. Use the addNode command later
#      after the node is created instead. The default value for this parameter is
#      8879. The port that you indicate must be a positive integer and a
#      connection to the deployment manager must be available in conjunction with
#      the dmgrHost parameter. Required if you are creating a managed profile,
#      the -federateLaterBPM parameter is set to false, and the deployment
#      manager was configured with a port that was not a default port.
#
# Valid Values:
#    Varies
# Default Values:
#    None
################################################################################
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password


################################################################################
# Parameter:  environmentType
#
# Description:
#      Specifies the environment type. The environment type refers to how IBM
#      Business Process Manager is used; for example, in a production, stage or
#      test environment. Load testing might be done on a test server, while a
#      stage environment type might be used as a temporary location to host
#      changes before putting those changes into production. You might specify
#      Stage as the Environment type if the server you are configuring will be
#      accessed and used to review content and new functionality.  Valid values
#      are as follows:
#      Test
#      Use Test if the server you are configuring is to be used as a testing
#      environment.
#      Stage
#      Use Stage if the server is to serve as a staging platform to be used as a
#      preproduction server.
#      Production
#      Use Production if the server is to serve in a production capacity.
#      The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
################################################################################
environmentType=Test
```

494   Installing

```
#############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
#############################################################################
#isDeveloperServer=false
```

Response file for creating a Standard Process Server custom profile for SQL:

Modify and use the response file provided here with the **manageprofiles** command-line utility to create a custom profile for a SQL database.

Note: You might need to comment or uncomment information in the response file for your operating system. For more information, see the comments in the response file.

```
# BEGIN COPYRIGHT
# ***************************************************************************
#
#  Licensed Materials - Property of IBM
#  5725-C94
#  (C) Copyright IBM Corporation 2011, 2012. All Rights Reserved.
#  This sample program is provided AS IS and may be used, executed, copied
#  and modified without royalty payment by customer (a) for its own
#  instruction and study, (b) in order to develop applications designed to
#  run with an IBM WebSphere product, either for customer's own internal
#  use or for redistribution by customer, as part of such an application,
#  in customer's own products.
#
# ***************************************************************************
# END COPYRIGHT

#############################################################################
# These response file has the applicable parameters for creating a
# Managed PS Std profile that uses SQLServer.
#
# Depending on your environment, you may need to change the default values.
#
# To create a profile with this response file specify:
#
# was.install.root/bin/manageprofiles -response PS_Std_Managed_SQLServer.response
#
# If you use the -response parameter, it must be the only parameter on
# the command-line. If you include any manageprofile parameters
# (in addition to -response) on the command-line, the response file
# is ignored. And default WebSphere Application server profile will be created .
#############################################################################

#############################################################################
# Parameter:  create
#
# Description:
```

```
#      Creates the profile.
#      If you are creating a profile, this parameter is required.
##############################################################################
create

##############################################################################
# Parameter:  templatePath
#
# Description:
#      Specifies the directory path to the template files in the installation
#      root directory. Within the profileTemplates directory are various
#      directories that correspond to different profile types and that vary with
#      the type of product installed. The profile directories are the paths that
#      you indicate while using the -templatePath option.
#      Use absolute paths. This parameter must exist as a directory and point to
#      a valid template directory.
#
#      When using the -templatePath parameter, specify the fully qualified file
#      path for the parameter.
#
# Valid Values:
#      Varies
# Default Values:
#      None
##############################################################################
templatePath=BPM/managed.procsvr

##############################################################################
# Parameter:  profileName
#
# Description:
#      Specifies the name of the profile. Use a unique value when creating a
#      profile.
#      Each profile that shares the same set of product binaries must have a
#      unique name. The default profile name is based on the profile type and a
#      trailing number, for example:
#      profileType ProfileNumberwhere profileType is a value such as ProcSrv,
#      Dmgr, or Custom and ProfileNumber is a sequential number that creates a
#      unique profile name. The value for this parameter must not contain spaces
#      or characters that are not valid such as the following: *, ?, ", <, >, ,,
#      /, \, and |. The profile name that you choose must not be in use. On
#      Windows platforms: If the fully qualified path contains spaces, enclose
#      the value in quotation marks.  The default value is based on the
#      install_root directory, the profiles subdirectory, and the name of the
#      file. For example, the default for profile creation is:
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME/profileNamewhere
#      WS_WSPROFILE_DEFAULT_PROFILE_HOME is defined in the wasprofile.properties
#      file in the install_root/properties directory.  The value for this
#      parameter must be a valid path for the target system and must not be
#      currently in use. You must have permissions to write to the directory.
#
# Valid Values:
#      Varies
# Default Values:
#      None
##############################################################################
profileName=Custom01

##############################################################################
# Parameter:  cellName
#
# Description:
#      Specifies the cell name of the profile. Use a unique cell name for each
#      profile.
#      The default value for this parameter is based on a combination of the
#      short host name, the constant Cell, and a trailing number, for example:
#      if (DMgr)
```

```
#          shortHostNameCellCellNumber
#      else
#          shortHostNameNodeNodeNumberCell
#      where CellNumber is a sequential number starting at 01 and NodeNumber is
#      the node number that you used to define the node name.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
#
# Parameter:  nodeName
#
# Description:
#      Specifies the node name for the node that is created with the new profile.
#      Use a unique value within the cell or on the workstation. Each profile
#      that shares the same set of product binaries must have a unique node name.
#      This parameter is required for profile creation only with the
#      dmgr.esbserverdefault.procsvrdefault.procctrdefault.procsvr.advdefault.
#      procctr.advdmgr.procctrmanaged.procctrmanaged.procsvr templates.
#      The default value for this parameter is based on the short host name,
#      profile type, and a trailing number, for example:
#      if (DMgr)
#              shortHostNameCellManagerNodeNumber
#      else
#              shortHostNameNodeNodeNumber
#      where NodeNumber is a sequential number starting at 01.
#      The value for this parameter must not contain spaces or any characters
#      that are not valid such as the following: *, ?, ", <, >, ,, /, \, and |.
#
# Valid Values:
#    Varies
# Default Values:
#    Varies
###############################################################################
cellName=cell_name
nodeName=node_name


###############################################################################
# Parameter:  enableAdminSecurity
#
# Description:
#      For IBM Business Process Manager omit this parameter. Administrative
#      security is always enabled for IBM Business Process Manager profiles.
#      You must also specify the parameters -adminUserName and -adminPassword
#      along with the values for these parameters.
#
# Valid Values:
#    true
# Default Values:
#    true
###############################################################################
enableAdminSecurity=true
adminUserName=adminUser_ID
adminPassword=adminPassword


###############################################################################
# Parameter:  signingCertDN
#
# Description:
#      Specifies the distinguished name of the root signing certificate that you
#      create when you create the profile. Specify the distinguished name in
#      quotation marks. This default personal certificate is located in the
#      server keystore file. If you do not specifically create or import a root
#      signing certificate, one is created by default. See the
```

```
#     -signingCertValidityPeriod parameter and the -keyStorePassword.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  signingCertValidityPeriod
#
# Description:
#     An optional parameter that specifies the amount of time in years that the
#     root signing certificate is valid. If you do not specify this parameter
#     with the -signingCertDN parameter, the root signing certificate is valid
#     for 20 years.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  keyStorePassword
#
# Description:
#     Specifies the password to use on all keystore files created during profile
#     creation. Keystore files are created for the default personal certificate
#     and the root signing certificate.
#
# Valid Values:
#     Varies
# Default Values:
#     None
###############################################################################
#signingCertDN =distinguished_name
#signingCertValidityPeriod =validity_period
#keyStorePassword =keyStore_password

###############################################################################
# Parameter:  enableService
#
# Description:
#     Enables the creation of a Linux service. Valid values include true or
#     false. The default value for this parameter is false.  When the
#     manageprofiles command-line utility is run with the -enableService option
#     set to true , the Linux service is created with the profile when the
#     command is run by the root user. When a nonroot user runs the
#     manageprofiles command-line utility, the profile is created, but the Linux
#     service is not. The Linux service is not created because the nonroot user
#     does not have sufficient permission to set up the service. An
#     INSTCONPARTIALSUCCESS result is displayed at the end of the profile
#     creation and the profile creation log
#     install_root/logs/manageprofiles/profile_name_create.log contains a
#     message indicating the current user does not have sufficient permission to
#     set up the Linux service. Use this parameter when creating profiles only.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
#
# Parameter:  serviceUserName
#
# Description:
#     Specifies the user ID that is used during the creation of the Linux
#     service so that the Linux service will run under this user ID. The Linux
#     service runs whenever the user ID is logged on. Not required.
#
```

```
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceAccountType
#
# Description:
#     The type of the owner account of the Windows service created for the
#     profile. Valid values include specifieduser or localsystem. The
#     localsystem value runs the Windows service under the local account of the
#     user who creates the profile. The default value for this parameter is
#     localsystem.  Use this parameter when creating profiles only.
#
# Valid Values:
#     localsystem
#     specifieduser
# Default Values:
#     localsystem
#
# Parameter:  winserviceUserName
#
# Description:
#     Specify your user ID so that the Windows operating system can verify you
#     as an ID that is capable of creating a Windows service. Your user ID must
#     belong to the Administrator group and have the following advanced user
#     rights:
#     Act as part of the operating system
#     Log on as a service
#     The default value for this parameter is the current user name. The value
#     for this parameter must not contain spaces or characters that are not
#     valid such as the following: *, ?, ", <, >, ,, /, \, and |. The user that
#     you specify must have the proper permissions to create a Windows service.
#     You must specify the correct password for the user name that you choose.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winservicePassword
#
# Description:
#     Specify the password for the specified user or the local account that is
#     to own the Windows service.
#
# Valid Values:
#     Varies
# Default Values:
#     None
#
# Parameter:  winserviceCheck
#
# Description:
#     The value can be either true or false. Specify true to create a Windows
#     service for the server process that is created within the profile. Specify
#     false to not create the Windows service. The default value for this
#     parameter is false.
#
# Valid Values:
#     false
#     true
# Default Values:
#     false
#
# Parameter:  winserviceStartupType
#
```

```
# Description:
#     The following values for Windows service startup can be used:
#     manual
#     automatic
#     disabled
#     The default value for this parameter is manual.
#
# Valid Values:
#     manual
#     automatic
#     disabled
# Default Values:
#     manual
##############################################################################
#enableService=true
#serviceUserName=service_user_ID
#winserviceAccountType=localsystem
#winserviceUserName=winservice_user_ID
#winservicePassword=winservice_password
#winserviceCheck=false
#winserviceStartupType=manual


##############################################################################
# Parameter:  dbWinAuth
#
# Description:
#     An optional parameter that is valid in all default profile templates.
#     Specify true to indicate that you are running SQL Server using integrated
#     windows authentication.
#
#     Note: Do not use this parameter if you are using -bpmdbDesign.
#
# Valid Values:
#     false
#     true
# Default Values:
#     None
##############################################################################
#dbWinAuth=true

##############################################################################
# Parameter:  federateLaterBPM
#
# Description:
#     Indicates if the managed profile is to be federated later using either the
#     profileTemplates/managed.esbserver or the
#     profileTemplates/BPM/managed.templateName template. Valid values are true
#     or false.    If the -dmgrHost, -dmgrPort, -dmgrAdminUserName and
#     -dmgrAdminPassword parameters are not set, the default value for this
#     parameter is true.  If you use -federateLaterBPM, you must either omit the
#     WebSphere Application Server-based parameter -federateLater or set it to
#     true.
#
# Valid Values:
#     false
#     true
# Default Values:
#     true
#
# Parameter:  dmgrHost
#
# Description:
#     Identifies the workstation where the deployment manager is running.
#     Specify this parameter and the dmgrPort parameter to federate a custom
#     profile as it is created or augmented. This parameter is available with
#     the managed.templateName and managed.esbserver profile templates.  The
#     host name can be the long or short DNS name or the IP address of the
```

```
#      deployment manager workstation. Specifying this optional parameter directs
#      the manageprofiles command-line utility to attempt to federate the custom
#      node into the deployment manager cell as it creates the custom profile.
#      This parameter is ignored when creating a deployment manager profile or
#      stand-alone server profile. If you federate a custom node when the
#      deployment manager is not running, the installation indicator in the logs
#      is INSTCONFFAILED to indicate a complete failure. The resulting custom
#      profile is unusable. You must move the custom profile directory out of the
#      profile repository (the profile's installation root directory) before
#      creating another custom profile with the same profile name. If you have
#      changed the default JMX connector type, you cannot federate with the
#      manageprofiles command-line utility. Use the addNode command later after
#      the node is created instead. The default value for this parameter is
#      localhost. The value for this parameter must be a properly formed host
#      name and must not contain spaces or characters that are not valid such as
#      the following: *, ?, ", <, >, , , /, \, and |. A connection to the
#      deployment manager must also be available in conjunction with the dmgrPort
#      parameter.  Required if you are creating a managed profile with the
#      -federateLaterBPM parameter set to false and the deployment manager is not
#      on the local machine.
#
# Valid Values:
#      Varies
# Default Values:
#      None
#
# Parameter:  dmgrPort
#
# Description:
#      Identifies the SOAP port of the deployment manager. Specify this parameter
#      and the dmgrHost parameter to federate a custom profile as it is created
#      or augmented. The deployment manager must be running and accessible.  If
#      you have changed the default JMX connector type, you cannot federate with
#      the manageprofiles command-line utility. Use the addNode command later
#      after the node is created instead. The default value for this parameter is
#      8879. The port that you indicate must be a positive integer and a
#      connection to the deployment manager must be available in conjunction with
#      the dmgrHost parameter. Required if you are creating a managed profile,
#      the -federateLaterBPM parameter is set to false, and the deployment
#      manager was configured with a port that was not a default port.
#
# Valid Values:
#      Varies
# Default Values:
#      None
################################################################################
federateLaterBPM=true
#dmgrHost=dmgr_host_name
#dmgrPort=dmgr_port_number
#dmgrAdminUserName=dmgr_admin_user_name
#dmgrAdminPassword=dmgr_admin_password

################################################################################
# Parameter:  environmentType
#
# Description:
#      Specifies the environment type. The environment type refers to how IBM
#      Business Process Manager is used; for example, in a production, stage or
#      test environment. Load testing might be done on a test server, while a
#      stage environment type might be used as a temporary location to host
#      changes before putting those changes into production. You might specify
#      Stage as the Environment type if the server you are configuring will be
#      accessed and used to review content and new functionality.  Valid values
#      are as follows:
#      Test
#      Use Test if the server you are configuring is to be used as a testing
#      environment.
```

```
#     Stage
#     Use Stage if the server is to serve as a staging platform to be used as a
#     preproduction server.
#     Production
#     Use Production if the server is to serve in a production capacity.
#     The default value is Test.
#
# Valid Values:
#    Test
#    Production
#    Stage
# Default Values:
#    Test
###############################################################################
environmentType=Test

###############################################################################
# Parameter:  isDeveloperServer
#
# Description:
#     Specifies whether the server is intended for development purposes only.
#     This parameter is useful when creating profiles to test applications on a
#     non-production server prior to deploying the applications on their
#     production application servers.  If -isDeveloperServer is set when
#     creating an IBM Business Process Manager profile, then a preconfigured VMM
#     file repository is installed. This file repository contains a sample
#     organization that can be used to test Business Process Choreographer
#     people resolution, ready for you to use as is.
#
# Valid Values:
#    false
#    true
# Default Values:
#    false
###############################################################################
#isDeveloperServer=false
```

manageprofiles examples:

The examples in this section show how to create stand-alone, deployment manager, and custom (managed node) profiles using the **manageprofiles** command-line utility.

Related tasks:

"Creating Process Server deployment manager and custom profiles using manageprofiles" on page 408
Use the **manageprofiles** command-line utility to create Process Server deployment manager and custom profiles for a network deployment configuration.

"Creating Process Server stand-alone profiles using manageprofiles" on page 240
Use the **manageprofiles** command-line utility to create Process Server stand-alone profiles for IBM Business Process Manager Standard .

"Creating Process Center stand-alone profiles using manageprofiles" on page 191
Use **manageprofiles** command-line utility to create Process Center stand-alone profiles for IBM Business Process Manager Standard .

Examples: Creating Process Server profiles with manageprofiles command-line utility using a DB2 database:

This topic contains example profile creation commands to help you create *Process Server* stand-alone server, deployment manager, and custom profiles using the **manageprofiles** command-line utility on your installation with a DB2 database.

Stand-alone server profile

The following command example creates an IBM Business Process Manager stand-alone server profile called *my_BPM_PSVRSA_profile* on a Windows server. The parameters in Specified **manageprofiles** command-line utility parameters and Defaulted **manageprofiles** command-line utility parameters specify the following:

- The DB2 database product will be used for the Process Server and Performance Data Warehouse databases, which are all assumed to already exist on the localhost. All databases are set to be configured later (the **-dbDelayConfig "true"** command parameter value specifies that configuration scripts be created but not run). For complete listings of database-related **manageprofiles** parameters, see the topic *manageprofiles parameters*.
- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

  Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.
- Administrative security will be enabled.

Specified manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a stand-alone server profile.

Table 65. Specified **manageprofiles** *command-line utility parameters*

Parameter	Value
-create	N/A
-templatePath	"*install_root*\profileTemplates\BPM\default.procsvr" (must be fully qualified)
-profileName	"*my_BPM_PSVRSA_profile*"
-adminPassword	"*admin_pwd*"
-adminUserName	"*admin_id*"
-dbServerPort	"50000"
-dbHostName	"localhost"
-dbType	"DB2_UNIVERSAL" or "DB2_DATASERVER"
-procSvrDbName	"BPMDB"
-perfDWDbName	"PDWDB"
-dbCreateNew	"false"
-dbDelayConfig	"true"
-dbUserId	"*db_id*"
-dbPassword	"*db_pwd*"
-samplesPassword	"*samples_pwd*"
-environmentName	"*environment_name*"

Remember: The **enableAdminSecurity** parameter is always true and cannot be false so it does not need to be specified.

Tip: The **samplesPassword** parameter is only required when using Samples.

Defaulted manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

*Table 66. Defaulted **manageprofiles** command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PSVRSA_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*Node*node_number*"
-cellName	"*host_name*Node*node_number*cell_number*Cell"
Windows -environmentType	"Test"
Windows -processCenterURL	Defaults to an offline server when no URL is specified.
Windows -winserviceCheck	"true"
Windows -winserviceAccountType	"localsystem"
Windows -winserviceStartupType	"manual"
Windows -winserviceUserName	"Administrator"
-dbJDBCClasspath	"*install_root*\jdbcdrivers\DB2"
-dbOutputScriptDir	"*install_root*\profiles*my_BPM_PSVRSA_profile*\dbscripts"

Deployment manager profile

The following command example creates a deployment manager profile called *my_BPM_PSVRDMGR_profile* on a Windows server.

The parameters in Table 67 and Table 68 on page 505 specify the following:

- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

  Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.
- Administrative security will be enabled.

Specified manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a deployment manager profile.

*Table 67. Specified **manageprofiles** command-line utility parameters*

Parameter	Value
-create	N/A
-templatePath	"*install_root*\profileTemplates\BPM\dmgr.procsvr" (must be fully qualified)
-profileName	"*my_BPM_PSVRDMGR_profile*"
-adminPassword	"*admin_pwd*"
-adminUserName	"*admin_id*"
-dbType	"DB2_UNIVERSAL" or "DB2_DATASERVER"
-dbCreateNew	"false"

Parameter	Value
-dbDelayConfig	"true"
-dbUserId	"*db_id*"
-dbPassword	"*db_pwd*"
-dbHostName	"*remote_host_name*"
-dbServerPort	"50000"

Remember: The **enableAdminSecurity** parameter is always true and cannot be false so it does not need to be specified.

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 68. Defaulted `manageprofiles` *command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PSVRDMGR_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*CellManager*node_number*"
-cellName	"*host_name*Cell*cell_number*"
`Windows` -winserviceCheck	"true"
`Windows` -winserviceAccountType	"localsystem"
`Windows` -winserviceStartupType	"manual"
`Windows` -winserviceUserName	"Administrator"
-dbJDBCClasspath	"*install_root*\jdbcdrivers\DB2"
-dbOutputScriptDir	"*install_root*\profiles*my_BPM_PSVRDMGR_profile*\dbscripts"

Custom profile

The following command example creates a custom profile called *my_BPM_PSVRCUSTOM_profile* on a Windows server.

This example is set to operate with the deployment manager profile created above.

The parameters in Table 69 on page 506 and Table 70 on page 506 specify the following:
- Administrative security will be enabled on the deployment manager to which the custom profile will be federated.
- The custom node will be federated during profile creation for which the deployment manager must be running.

   Tip: If the deployment manager is not running, or you want to federate the custom node after profile creation, set the **federateLaterBPM** to true.

See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

Specified **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a custom profile.

Table 69. Specified **manageprofiles** *command-line utility parameters*

Parameter	Value
-create	N/A
-templatePath	"*install_root*\profileTemplates\BPM\managed.procsvr" (must be fully qualified)
-profileName	"*my_BPM_PSVRCUSTOM_profile*"
-dmgrHost	"*remote_host*"
-dmgrPort	"8879" (To find the -dmgrPort value, go to the *dmgr_profile_root*\logs directory for the deployment manager associated with this custom profile. In this directory, open the AboutThisProfile.txt file and find the value for the entry "Management SOAP connector port:".)
-dmgrAdminPassword	"*admin_pwd*"
-dmgrAdminUserName	"*admin_id*"
-federateLaterBPM	"false"
-dbType	"DB2_UNIVERSAL" or "DB2_DATASERVER"
-dbJDBCClasspath	"*install_root*\jdbcdrivers\DB2"

Remember: If the **federateLaterBPM** parameter is set to true, then the dmgrHost, dmgrPort, dmgrAdminPassword and dmgrAdminUserName do not need to be specified.

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 70. Defaulted **manageprofiles** *command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PSVRCUSTOM_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*Node*node_number*"

Examples: Creating Process Center profiles with manageprofiles command-line utility using an Oracle database:

This topic contains example profile creation commands to help you create Process Center stand-alone server, deployment manager and custom profiles using the manageprofiles command-line utility on your installation with an Oracle database.

Stand-alone server profile

The following command example creates a Process Center stand-alone profile called *my_BPM_PCTRSA_profile* on a Windows server.

The parameters in Specified **manageprofiles** command-line utility parameters specify the following features:

- The Oracle database product will be used for both the Process Server and Performance Data Warehouse databases, which are assumed to already exist on the localhost. All databases are set to be configured later (the **-dbDelayConfig "true"** command parameter value specifies that configuration scripts be created but not run).

- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

  Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.
- Administrative security will be enabled.

All user IDs specified for profile creation should already exist in the database before any database configuration is performed:
- If **dbDelayConfig** is set to false, the user IDs must be created before profile creation.
- If **dbDelayConfig** is set to true, the user IDs can be created at the same time as the database tables after profile creation is complete and before starting the server.

  Note: If **dbDelayConfig** is set to true, the bootstrapPocessServerData script must be run before starting the server. See, "Loading the database with system information" on page 581 for details. In this scenario, this step does apply to stand-alone profiles.

Specified manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a stand-alone server profile.

Table 71. Specified **manageprofiles** *command-line utility parameters*

Parameter	Value
-create	N/A
-templatePath	"*install_root*\profileTemplates\BPM\default.procctr" (must be fully qualified)
-profileName	"*my_BPM_PCTRSA_profile*"
-adminPassword	"*admin_pwd*"
-adminUserName	"*admin_id*"
-dbServerPort	"1521"
-dbType	"ORACLE"
-procSvrDbName	"BPMDB"
-dbProcSvrUserId	"*procsvr_db_id*"
-dbProcSvrPassword	"*procsvr_db_pwd*"
-perfDWDbName	"PDWDB"
-dbPerfDWUserId	"*perfdw_db_id*"
-dbPerfDWPassword	"*perfdw_db_pwd*"
-dbDelayConfig	"true"
-samplesPassword	"*samples_pwd*"
-environmentName	"*environment_name*"

Remember: The **enableAdminSecurity** parameter is always true and cannot be false so it does not need to be specified.

Tip: The **samplesPassword** parameter is only required when using Samples.

Restriction: The **dbProcSvrUserId** and **dbPerfDWUserId** parameters cannot have the same value if the same value was given for the **procSvrDbname** and **perfDWDbName** parameters.

Defaulted manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 72. Defaulted **manageprofiles** *command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PCTRSA_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*Node*node_number*"
-cellName	"*host_name*Node*node_number*cell_number*Cell"
`Windows` -environmentType	"Test"
`Windows` -winserviceCheck	"true"
`Windows` -winserviceAccountType	"localsystem"
`Windows` -winserviceStartupType	"manual"
`Windows` -winserviceUserName	"Administrator"
-dbJDBCClasspath	"*install_root*\jdbcdrivers\Oracle"
-dbOutputscriptDir	"*install_root*\profiles*my_BPM_PCTRSA_profile*\dbscripts"
-dbHostName	"local_host_name"

Additional manageprofiles command-line utility parameters for Oracle shows additional **manageprofiles** command-line utility parameters that are not displayed via the Profile Management Tool that can be specified to select your own user name and password combinations for Oracle.

Table 73. Additional **manageprofiles** *command-line utility parameters for Oracle*

Parameter	Value	Remarks
-dbCeiMeUserId	"*cei_me_userID*" (used to create CEI ME objects)	This parameter is needed if you select the Custom Users and Passwords option during profile creation. For Generate Users with Single Password option, this user will be generated automatically based on Oracle Database name [SID]. For example: If SID is ORCL, dbCeiMeId is ORCCM00
-dbCeiMePassword	"*cei_me_pwd*"	This parameter is needed if you select the Custom Users and Passwords option during profile creation. For the Generate Users with Single Password option, dbCeiPassword will be set to dbPassword. For example: dbCeiMePassword = dbPassword
-dbAppMeUserId	"*app_me_userID*" (used to create SCAAPP ME objects)	This parameter is needed if you select the Custom Users and Passwords option during profile creation. For the Generate Users with Single Password option, this user will be generated automatically based on Oracle Database name [SID]. For example: If SID is ORCL, dbAppMeId is ORCSA00
-dbAppMePassword	"*app_me_pwd*"	This parameter is needed if you select the Custom Users and Passwords option during profile creation. For the Generate Users with Single Password option, dbCeiPassword will be set to dbPassword. For example: dbAppMePassword = dbPassword

Deployment manager profile

The following command example creates a deployment manager profile called
my_BPM_PCTRDMGR_profile on a Windows server.

The parameters in Specified manageprofiles command-line utility parameters specify the following:
- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

  Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.
- Administrative security will be enabled.

Specified manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a deployment manager profile.

Table 74. Specified **manageprofiles** *command-line utility parameters*

Parameter	Value
-create	N/A
-templatePath	"*install_root*\profileTemplates\BPM\dmgr.procctr" (must be fully qualified)
-profileName	"*my_BPM_PCTRDMGR_profile*"
-adminPassword	"*admin_pwd*"
-adminUserName	"*admin_id*"
-dbType	"ORACLE"
-dbDelayConfig	"true"
-dbHostName	"*remote_host_name*"
-dbServerPort	"1521"

Remember: The **enableAdminSecurity** parameter is always true and cannot be false so it does not need to be specified.

Defaulted manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 75. Defaulted **manageprofiles** *command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PCTRDMGR_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*CellManager*node_number*"
-cellName	"*host_name*Cell*cell_number*"
`Windows` -winserviceCheck	"true"
`Windows` -winserviceAccountType	"localsystem"
`Windows` -winserviceStartupType	"manual"
`Windows` -winserviceUserName	"Administrator"

Table 75. Defaulted `manageprofiles` *command-line utility parameters (continued)*

Parameter	Default values
-dbJDBCClasspath	"*install_root*\jdbcdrivers\Oracle"
-dbOutputScriptDir	"*install_root*\profiles*my_BPM_PCTRDMGR_profile*\dbscripts"

Additional manageprofiles command-line utility parameters for Oracle shows additional **manageprofiles** command-line utility parameters that are not displayed via the Profile Management Tool that can be specified to select your own user name and password combinations for Oracle.

Custom profile

The following command example creates a custom profile called *my_BPM_PCTRCUSTOM_profile* on a Windows server.

This example is set to operate with the deployment manager profile created above.

The parameters in Table 76 and Table 77 on page 511 specify the following:
- Administrative security will be enabled on the deployment manager to which the custom profile will be federated.
- The custom node will be federated during profile creation for which the deployment manager must be running.

    Tip: If the deployment manager is not running, or you want to federate the custom node after profile creation, set the **federateLaterBPM** to true.

See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

Specified **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a custom profile.

Table 76. Specified `manageprofiles` *command-line utility parameters*

Parameter	Value
-create	N/A
-templatePath	"*install_root*\profileTemplates\BPM\managed.procctr" (must be fully qualified)
-profileName	"*my_BPM_PCTRCUSTOM_profile*"
-dmgrHost	"*remote_host*"
-dmgrPort	"8879" (To find the -dmgrPort value, go to the *dmgr_profile_root*\logs directory for the deployment manager associated with this custom profile. In this directory, open the AboutThisProfile.txt file and find the value for the entry "Management SOAP connector port:".)
-dmgrAdminPassword	"*admin_pwd*"
-dmgrAdminUserName	"*admin_id*"
-federateLaterBPM	"false"
-dbType	"ORACLE"
-dbJDBCClasspath	"*install_root*\jdbcdrivers\Oracle"

Remember: If the **federateLaterBPM** parameter is set to true, then the dmgrHost, dmgrPort, dmgrAdminPassword and dmgrAdminUserName do not need to be specified.

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 77. Defaulted **manageprofiles** *command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PCTRCUSTOM_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*Node*node_number*"

Examples: Creating Process Server and WebSphere ESB profiles with **manageprofiles** *command-line utility using an Oracle database:*

This topic contains example profile creation commands to help you create *Process Server* stand-alone server, deployment manager, and custom profiles using the **manageprofiles** command-line utility on your installation with an Oracle database.

Stand-alone server profile

The following command example creates an IBM Business Process Manager stand-alone profile called *my_BPM_PSVRSA_profile* on a Windows server.

The parameters in Specified **manageprofiles** command-line utility parameters, Defaulted **manageprofiles** command-line utility parameters, and Additional **manageprofiles** command-line utility parameters for Oracle specify the following features:

- The Oracle database product will be used for the Process Server and Performance Data Warehouse databases, which are all assumed to already exist on the localhost. All databases are set to be configured later (the **-dbDelayConfig "true"** command parameter value specifies that configuration scripts be created but not run). For complete listings of database-related **manageprofiles** parameters, see the topic *manageprofiles parameters*.
- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

  Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.
- Administrative security will be enabled.

All user IDs specified for profile creation should already exist in the database before any database configuration is performed:

- If **dbDelayConfig** is set to false, the user IDs must be created before profile creation.
- If **dbDelayConfig** is set to true, the user IDs can be created at the same time as the database tables after profile creation is complete and before starting the server.

Specified manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a stand-alone server profile.

Table 78. Specified `manageprofiles` *command-line utility parameters*

Parameter	Value
-create	N/A
-templatePath	"*install_root*\profileTemplates\BPM\default.procsvr" (must be fully qualified)
-profileName	"*my_BPM_PSVRSA_profile*"
-adminPassword	"*admin_pwd*"
-adminUserName	"*admin_id*"
-dbServerPort	"1521"
-dbType	"ORACLE"
-procSvrDbName	"BPMDB
-dbProcSvrUserId	"*procsvr_db_id*"
-dbProcSvrPassword	"*procsvr_db_pwd*"
-perfDWDbName	"PDWDB"
-dbPerfDWUserId	"*perfdw_db_id*"
-dbPerfDWPassword	"*perfdw_db_pwd*"
-dbDelayConfig	"true"
-environmentName	"*environment_name*"

Remember: The **enableAdminSecurity** parameter is always true and cannot be false so it does not need to be specified.

Tip: The **samplesPassword** parameter is only required when using Samples.

Restriction: The **dbProcSvrUserId** and **dbPerfDWUserId** parameters cannot have the same value if the same value was given for the **procSvrDbname** and **perfDWDbName** parameters.

Defaulted manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 79. Defaulted `manageprofiles` *command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PSVRSA_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*Node*node_number*"
-cellName	"*host_name*Node*node_number*cell_number*Cell"
`Windows` -winserviceCheck	"true"
`Windows` -winserviceAccountType	"localsystem"
`Windows` -winserviceStartupType	"manual"
`Windows` -winserviceUserName	"Administrator"
-dbJDBCClasspath	"*install_root*\jdbcDrivers\Oracle"
-dbOutputscriptDir	"*install_root*\profiles*my_BPM_PSVRSA_profile*\dbscripts"
-dbHostName	"local_host_name"

Additional manageprofiles command-line utility parameters for Oracle shows additional **manageprofiles** command-line utility parameters that are not displayed via the Profile Management Tool that can be specified to select your own user name and password combinations for Oracle.

*Table 80. Additional **manageprofiles** command-line utility parameters for Oracle*

Parameter	Value	Remarks
-dbCeiMeUserId	"*cei_me_userID*" (used to create CEI ME objects)	This parameter is needed if you select the Custom Users and Passwords option during profile creation. For Generate Users with Single Password option, this user will be generated automatically based on Oracle Database name [SID]. For example: If SID is ORCL, dbCeiMeId is ORCCM00
-dbCeiMePassword	"*cei_me_pwd*"	This parameter is needed if you select the Custom Users and Passwords option during profile creation. For the Generate Users with Single Password option, dbCeiPassword will be set to dbPassword. For example: dbCeiMePassword = dbPassword
-dbAppMeUserId	"*app_me_userID*" (used to create SCAAPP ME objects)	This parameter is needed if you select the Custom Users and Passwords option during profile creation. For the Generate Users with Single Password option, this user will be generated automatically based on Oracle Database name [SID]. For example: If SID is ORCL, dbAppMeId is ORCSA00
-dbAppMePassword	"*app_me_pwd*"	This parameter is needed if you select the Custom Users and Passwords option during profile creation. For the Generate Users with Single Password option, dbCeiPassword will be set to dbPassword. For example: dbAppMePassword = dbPassword

Deployment manager profile

The following command example creates a deployment manager profile called *my_BPM_PSVRDMGR_profile* on a Windows server.

The parameters in Table 81 and Table 82 on page 514 specify the following:
- The Windows service will be set for manual startup.
- The profile creation process will set the port values automatically (except for database-related ports). The process will validate the new profile against other profiles to ensure there are no port conflicts.

  Tip: To override the port values that the **manageprofiles** command-line utility will specify, use the **-portsFile** parameter. See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.
- Administrative security will be enabled.

Specified manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a deployment manager profile.

*Table 81. Specified **manageprofiles** command-line utility parameters*

Parameter	Value
-create	N/A

Table 81. Specified `manageprofiles` *command-line utility parameters (continued)*

Parameter	Value
-templatePath	"*install_root*\profileTemplates\BPM\dmgr.procsvr" (must be fully qualified)
-profileName	"*my_BPM_PSVRDMGR_profile*"
-adminPassword	"*admin_pwd*"
-adminUserName	"*admin_id*"
-dbType	"ORACLE"
-dbDelayConfig	"true"
-dbPassword	"*db_pwd*"
-dbHostName	"*remote_host_name*"
-dbServerPort	"1521"

Remember: The **enableAdminSecurity** parameter is always true and cannot be false so it does not need to be specified.

Defaulted manageprofiles command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

Table 82. Defaulted `manageprofiles` *command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PSVRDMGR_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*CellManager*node_number*"
-cellName	"*host_name*Cell*cell_number*"
Windows -winserviceCheck	"true"
Windows -winserviceAccountType	"localsystem"
Windows -winserviceStartupType	"manual"
Windows -winserviceUserName	"Administrator"
-dbJDBCClasspath	"*install_root*\jdbcDrivers\Oracle"
-dbOutputScriptDir	"*install_root*\profiles*my_BPM_PSVRDMGR_profile*\dbscripts"

Custom profile

The following command example creates a custom profile called *my_BPM_PSVRCUSTOM_profile* on a Windows server.

This example is set to operate with the deployment manager profile created above.

The parameters in Table 83 on page 515 and Table 84 on page 515 specify the following:
* The Oracle database product will be used for the Common database, which is assumed to already exist. The custom profile creation needs to point to the database used by the deployment manager to which the custom profile will be federated.
* Administrative security will be enabled on the deployment manager to which the custom profile will be federated.

See *manageprofiles parameters* in the reference documentation for a listing of all valid **manageprofiles** parameters.

Specified **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with example values used to create a custom profile.

*Table 83. Specified **manageprofiles** command-line utility parameters*

Parameter	Value
-create	N/A
-templatePath	"*install_root*\profileTemplates\BPM\managed.procsvr" (must be fully qualified)
-profileName	"*my_BPM_PSVRCUSTOM_profile*"
-dmgrHost	"*remote_host*"
-dmgrPort	"8879" (To find the -dmgrPort value, go to the *dmgr_profile_root*\logs directory for the deployment manager associated with this custom profile. In this directory, open the AboutThisProfile.txt file and find the value for the entry "Management SOAP connector port:".)
-dmgrAdminPassword	"*admin_pwd*"
-dmgrAdminUserName	"*admin_id*"
-federateLaterBPM	"false"
-dbType	"ORACLE"
-dbJDBCClasspath	"*install_root*\jdbcDrivers\Oracle"

Remember: If the **federateLaterBPM** parameter is set to true, then the dmgrHost, dmgrPort, dmgrAdminPassword and dmgrAdminUserName do not need to be specified.

Defaulted **manageprofiles** command-line utility parameters shows **manageprofiles** command-line utility parameters with default values that do not normally have to be changed.

*Table 84. Defaulted **manageprofiles** command-line utility parameters*

Parameter	Default values
-profilePath	"*install_root*\profiles*my_BPM_PSVRCUSTOM_profile*"
-hostName	"*host_name*"
-nodeName	"*host_name*Node*node_number*"

Examples: Using a database design file with the manageprofiles command:

If you create a database design file using the database design tool, the generated file contains the values for the manageprofiles command.

If you specify the **-bpmdbDesign** parameter and provide the name for the database design file, the only other database parameter that you need to specify is the **dbDelayConfig** parameter.

Stand-alone server profile for BPM Standard

The following example creates a stand-alone server profile called *SAPC* for BPM Standard:

```
manageprofiles.bat -create -templatePath C:\w\4210\profileTemplates\BPM\default.procctr -profileName SAPC -profilePath C
```

Stand-alone server profile for BPM Advanced

The following example creates a stand-alone server profile called *SAAPC* for BPM Advanced:

```
manageprofiles.bat -create -templatePath C:\w\4210\profileTemplates\BPM\default.procctr.adv -profileName SAAPC -profilePath
manageprofiles.bat -create -templatePath C:\w\4210\profileTemplates\default.esbserver -profileName SAESB -profilePath C:\w\
```

Augmenting profiles

You can augment an existing profile for WebSphere Application Server version 7.0 or WebSphere Application Server Network Deployment version 7.0 to add support for WebSphere Enterprise Service Bus.

- See the list of prerequisites for creating or augmenting profiles in the topic Prerequisites for creating or augmenting profiles.
- Ensure that the profile has the following characteristics:
  - It exists on a system with an installation of IBM Business Process Manager.
  - It is not federated to a deployment manager. You cannot use the Profile Management Tool or the **manageprofiles** command-line utility to augment federated profiles.
  - It does not have running servers.

If you have existing WebSphere Application Server or WebSphere Application Server Network Deployment profiles on your system, you might want the operating environments defined by those profiles to have WebSphere ESB functionality.

Restrictions:

- You cannot use the Profile Management Tool to augment profiles on IBM Business Process Manager installations on 64-bit architectures except on the Linux on zSeries platform. To augment profiles on other 64-bit architectures, you can use the **manageprofiles** command-line utility. For information about using the **manageprofiles** command-line utility, see Augmenting profiles using the manageprofiles command-line utility. You can also use the Profile Management Tool on these architectures if you use an IBM Business Process Manager 32-bit installation.

Use the instructions in this section and its subsections to augment Process Server and Process Center profiles interactively by using the Profile Management Tool graphical user interface (GUI) or, from a command line, by using the **manageprofiles** command-line utility.

Augmenting profiles using the Profile Management Tool:

Use the Profile Management Tool to augment WebSphere Application Server version 7.0, WebSphere Application Server Network Deployment version 7.0, or WebSphere Enterprise Service Bus version 7.5.1 profiles into IBM Business Process Manager version 7.5.1 Process Server profiles.

Ensure that the following prerequisites are satisfied:

- The profile type you will augment to (stand-alone server, deployment manager, or custom) is the same as the type of the profile from which you will augment.
- You have reviewed the list of prerequisites for creating or augmenting profiles at "Prerequisites for creating or augmenting profiles" on page 131.
- You have shut down any servers associated with the profile you plan to augment.
- If you plan to augment a stand-alone server or custom profile, you ensured that it is *not* federated to a deployment manager.
- Solaris When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons of the Profile Management Tool. To fix the problem, add the following lines to the *install_root*/.Xdefaults file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

1. Start the IBM Business Process Manager Profile Management Tool.

   Use one of the following commands:

   - **`Linux`** **`UNIX`** *install_root*/**bin/ProfileManagement/pmt.sh**

   - **`Windows`** *install_root***bin\ProfileManagement\pmt.bat**

   See the topic Starting the Profile Management Tool for other methods of starting this tool.

   The Welcome page is displayed.

2. In the Welcome page, click the **Launch Profile Management Tool** button or the **Profile Management Tool** tab.

   The **Profiles** tab is displayed.

3. In the **Profiles** tab, highlight the profile you want to augment and click **Augment**.

   The **Profiles** tab lists the profiles that exist on your system. For this procedure, it is assumed you are augmenting an existing profile.

   Restrictions:
   - You cannot augment WebSphere Application Server, WebSphere Application Server Network Deployment, or WebSphere Enterprise Service Bus version 6.2 profiles into IBM Business Process Manager version 7.0 profiles.
   - You cannot augment cell stand-alone server, management administrative agent, management job manager, or secure proxy profiles.
   - If you augment a WebSphere Application Server or WebSphere Application Server Network Deployment profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button is cannot be selected unless a profile can be augmented.

   The Augment Selection page opens in a separate window.

4. In the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**.

   The Profile Augmentation Options page is displayed.

5. In the Profile Augmentation Options page, choose to perform a **Typical** or an **Advanced** profile augmentation, and click **Next**.

   The **Typical** option augments a profile with default configuration settings.

   The **Advanced** option lets you specify your own configuration values for a profile.

   Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:
   - The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
   - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
   - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

6. Before continuing to the next page in the Profile Management Tool, proceed to one of the following topics to configure and complete augmentation of your profile.

Type of profile augmentation you selected	Procedure to complete profile augmentation based on your profile type (stand-alone server, deployment manager, or custom)
Typical	For Process Server profiles: • "Augmenting stand-alone profiles for Process Server using the Typical option" on page 532 • "Augmenting deployment manager profiles for Process Server" on page 543 • "Augmenting custom profiles (managed nodes) for Process Server" on page 544 For Process Center profiles: • "Augmenting stand-alone profiles for Process Center using the **Typical** option" • "Augmenting Deployment Manager profiles for Process Center" on page 529 • "Augmenting custom profiles (managed nodes) for Process Center" on page 531
Advanced	For Process Server profiles: • "Augmenting stand-alone profiles for Process Server using the **Advanced** option" on page 537 For Process Center profiles: • "Augmenting stand-alone profiles for Process Center using the **Advanced** option" on page 524

Augmenting profiles for Process Center:

You can use the **Typical** profile augmentation option or the **Advanced** profile augmentation option to augment profiles for Process Center.

Augmenting stand-alone profiles for Process Center using the **Typical** *option:*

Use the Profile Management Tool to augment stand-alone profiles for Process Center.

Understand the concepts of profiles, including the differences between stand-alone, network deployment and custom profiles.

Understand the differences between the **Typical profile augmentation** option and the **Advanced profile augmentation** option, including under which scenarios you should use one over the other.
• Review the list of prerequisites for creating or augmenting profiles at "Prerequisites for creating or augmenting profiles" on page 131.

• ⬛ Solaris When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/.Xdefaults file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

Use the procedure in this task topic if you are:
• Augmenting a stand-alone server profile for *Process Center*.

The steps describe **Typical profile augmentation**.

1. Start the Profile Management Tool.

   Use one of the following commands:

   - `Linux` `UNIX` *install_root*/**bin/ProfileManagement/pmt.sh**

   - `Windows` *install_root***bin\ProfileManagement\pmt.bat**

   The Welcome page displays.

2. In the Welcome page, click **Launch Profile Management Tool** or select the **Profile Management Tool** tab.

   The **Profiles** tab displays.

   The **Profiles** tab contains a list of profiles on your machine currently. For this procedure, it is assumed you are augmenting an existing stand-alone profile for Process Center. For instructions on creating a new Process Center profile, see "Creating Process Center stand-alone profiles using the Profile Management Tool" on page 137

   Restrictions:
   - You cannot augment cell stand-alone server, management administrative agent, management job manager, or secure proxy profiles.
   - If you augment a WebSphere Application Server or WebSphere Application Server Network Deployment profile, it must be from the version of WebSphere Application Server on which IBM Business Process Manager is installed. The **Augment** button cannot be selected unless a profile can be augmented.

   The Augment Selection page opens in a separate window.

3. In the Augment Selection page, select the type of augmentation you want to apply to the profile. Then click **Next**

   The Profile Augmentation Options page displays.

   From the Profile Augmentation Options page, select the profile augmentation method to apply. The following options are available:
   - **Typical profile augmentation**

     Augments a profile with default configuration settings.
   - **Advanced profile augmentation**

     Lets you specify your own configuration values for the profile you are augmenting.

   Restriction: The Profile Management Tool displays a warning message if any of the following conditions occur:
   - The profile you selected to augment has a running server. You cannot augment the profile until you stop the server or click **Back** and choose another profile that does not have running servers.
   - The profile you selected to augment is federated. You cannot augment a federated profile. You must click **Back** and choose another profile that is not federated.
   - The profile you selected to augment is already augmented with the product you selected. You must click **Back** and choose another profile to augment.

4. From the Administrative Security page, re-enter the administrative user ID and password for the profile you are augmenting.

5. Select the configuration settings for the Performance Data Warehouse.

   To configure performance data warehouse functionality for the profile, select the **Configure Performance Data Warehouse** checkbox.

6. Configure the databases using a design file. **This option is for the Advanced path only.**

   a. Select **Use a database design file** .

   b. Click **Browse**.

c. Specify the fully qualified path name for the design file.

d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

> **Important:** If you select **Run database scripts to create database tables**, ensure that **Use an existing local or remote database** is *not* selected. If both options are selected, errors occur.

e. Click **Next**.

If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration. For more information on using a design file for database configuration, see "Creating database design files by using the database design tool" on page 569.

7. On the Database Configuration - Part 1 page, perform the following actions:

- From the **Select a database product** drop-down, select the database product to be used by the profile.

- Select whether to **Create a new local database** or to **Use an existing local or remote database**.

  If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

  If the database product you are using with the software already exists, select **Use an existing local or remote database** .

- In the **Process Server database name** field, enter a name for the database or accept the **BPMDB** default value.

- In the **Performance Data Warehouse database name** field, enter a name for the database or accept the **PDWDB** default value.

- Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

  If you do not select the checkbox, the scripts are output to the default directory.

- Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

8. In the Database Configuration - Part 2 page, complete the database configuration and click **Next**.

> **Note:** You must complete this page even if you decided not to run the database scripts. The values you choose on the Database Configuration - Part 2 page are added to the database configuration scripts stored in the directory you specified in the **Database script output directory** field on the previous page (or in the default directory for these scripts if you did not specify a different location).

> **Restriction:** You cannot create a new database if you are using DB2 for z/OS V8 or V9, or Oracle. In these cases, the Process Server / Performance Data Warehouse databases must exist.
> From the following list, select the link that names the database you are using.

- DB2 Database
- DB2 for z/OS V8, V9 and V10
- Microsoft SQL Server
- Oracle

a. Set values for the DB2 Database

The following table lists the fields you must complete on the Database Configuration (Part 2) page when you select DB2 Universal Database as your database product.

Table 85. Required database configuration fields for DB2 Database

Field	Action needed
JDBC driver	Select from the following options: • DB2 Universal • DB2 DataServer
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in the following directory: • If you selected the DB2 Express feature during the installation: `${WAS_INSTALL_ROOT}/db2/java` • If you did not select the DB2 Express feature during the installation: `${WAS_INSTALL_ROOT}/jdbcdrivers/DB2`
Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of `50000` or enter the correct server port number.

b. Set values for the DB2 for z/OS V8, V9 and V10

The following table lists the fields you must complete on the Database Configuration (Part 2) page when you select DB2 for z/OS V8 and V9 as your database product. You cannot create a new database using these databases. The performance data warehouse database and, for a stand-alone server profile, the process server database, must exist.

Table 86. Required database configuration fields for DB2 Universal Database for z/OS

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}/jdbcdrivers/DB2`
Database server host name (for example IP address)	Enter the database server host name.
Server port	Accept the default value of `446` or enter the correct server port number.
Process Server Schema name	Enter the database schema name for Process Server.
Performance Server Schema name	Enter the database schema name for Performance Server.
Connection location	Enter the connection location.
Storage group name	Enter the storage group name.

c. Set values for the Microsoft SQL Server

The following table lists the fields you must complete on the Database Configuration (Part 2) page when you select Microsoft SQL Server as your database product.

Table 87. Required database configuration fields for Microsoft SQL Server

Field	Action needed
JDBC driver	Select from the following options • **Microsoft SQL Server JDBC 2.0** **Note:** JDBC 3.0 is also supported. Select **Microsoft SQL Server JDBC 2.0** for JDBC 3.0. • **Microsoft SQL Server JDBC 1.2**

Process Server database	For the Process Server database, enter values for the following parameters:
	• `User name`
	  Enter the Process Server database user name.
	• `Password`
	  Enter a password to authenticate with the Process Server database.
	• `Confirm password`
	  Confirm the password by reentering it.

Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters:
	• `User name`
	  Enter the Performance Data Warehouse database user name.
	• `Password`
	  Enter a password to authenticate with the Performance Data Warehouse database.
	• `Confirm password`
	  Confirm the password by reentering it.

Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer`

d. Set values for the Oracle database

The following table lists the fields you must complete on the Database Configuration (Part 2) page when you select Oracle as your database product. You cannot create a new database using this database.

Important: You must have a user ID that has SYSDBA privileges before creating any profile.

Table 88. Required database configuration fields for Oracle

Field	Action needed

Process Server database	For the Process Server database, enter values for the following parameters:
	• `User name`
	  Enter the Process Server database user name.
	• `Password`
	  Enter a password to authenticate with the Process Server database.
	• `Confirm password`
	  Confirm the password by reentering it.
	
	Note: Process Server database user name and the Performance Data Warehouse database user name cannot be the same.

Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters:
	• `User name`
	  Enter the Performance Data Warehouse database user name.
	• `Password`
	  Enter a password to authenticate with the Performance Data Warehouse database.
	• `Confirm password`
	  Confirm the password by reentering it.
	
	Note: The Performance Data Warehouse database user name and the Process Server database cannot be the same.

Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle`

9. In the Profile Augmentation Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When profile augmentation completes, the Profile Augmentation Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might display in place of the success message, for example:

   • **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
   • **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

Augmenting stand-alone profiles for Process Center using the **Advanced** *option:*

Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Center. Selecting the **Advanced** option augments profiles with customized configuration settings.

Understand the concepts of profiles, including the differences between stand-alone, network deployment and custom profiles.

- Review the list of prerequisites for creating or augmenting profiles at "Prerequisites for creating or augmenting profiles" on page 131.

- Solaris When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/.Xdefaults file:

  ```
  Eclipse*spacing:0
  Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
  ```

  After adding the lines, run the following command before starting the Profile Management Tool:

  ```
  xrdb -load user_home/.Xdefaults
  ```

An *IBM BPM Advanced, Process Center stand-alone profile* is a defined server environment that is managed from its own console and functions independently from all other process centers.

This task describes how to use the **Advanced profile augmentation** option on the Profile Management Tool to augment a Stand-alone profile.

By selecting the **Advanced** option, you can perform the following tasks:
- Configure the databases using a database design file.

Important: The procedure in this topic includes all pages from the Profile Management Tool that might display when augmenting a stand-alone process server profile using the Advanced option. However, if the profile you are augmenting already includes configured components (for example, the common database or Business Space powered by WebSphere) the pages within the profile management tool for these components will not display.

Use the "Augmenting profiles using the Profile Management Tool" on page 516 procedure to get started with your profile augmentation and then resume with the following steps depending on where you left off.

Remember: All IBM Business Process Manager profiles have administrative security enabled.
1. Using the Profile Management Tool, select the profile you want to augment for Process Center and click **Augment**. The Augment Selection page opens.
2. Select the type of augmentation to apply to the profile and click **Next**. The Profile Augmentation Options page opens.
3. Select **Advanced profile augmentation** and click **Next**. The Administrative Security page opens.
4. From the Administrative Security page, re-enter the administrative user ID and password for the profile you are augmenting.

   If the profile you are augmenting has the WebSphere(r) Application Server sample application deployed, it requires an account under which to run. Supply the password for the account. You cannot change the user name of the account.
5. If you have multiple servers defined on your system, select the server. The Application Scheduler Configuration page is opened. In the **For Advanced profile augmentation when profile has multiple servers defined:** section of the Application Scheduler Configuration page, select the server from the drop-down list and click **Next**.

You have completed the initial steps to augment a stand-alone profile for Process Center using the **Advanced** option.

Complete the augmentation process depending on the following system set up:

- Databases are *not* already defined on your system.
- Databases *are* already defined on your system.

Related tasks:

"Augmenting stand-alone profiles for Process Center with the Advanced option with a database defined"
Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Center with customized configuration settings. You can augment the stand alone profiles when you already have a database set up on your system.

"Augmenting stand-alone profiles for Process Center with the Advanced option with databases not defined" on page 526
Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Center with customized configuration settings. You can augment the stand alone profiles when you do not yet have a database set up on your system.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Augmenting stand-alone profiles for Process Center with the Advanced option with a database defined:

Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Center with customized configuration settings. You can augment the stand alone profiles when you already have a database set up on your system.

Use the "Augmenting stand-alone profiles for Process Center using the **Advanced** option" on page 524 procedure to get started with your advanced profile augmentation and then resume with the steps in this task.

This task describes using the **Advanced profile augmentation** option on the Profile Management Tool user interface to augment stand-alone profiles if databases *are* already defined on your system.

Remember: All IBM Business Process Manager profiles have administrative security enabled.

1. In the Profile Augmentation Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When profile augmentation completes, the Profile Augmentation Complete page is displayed with the following message: **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might display in place of the success message, for example:

   The Profile Management tool augmented the profile but errors occurred
   >   Indicates that profile augmentation completed but errors were generated.

   The Profile Management tool cannot augment the profile
   >   Indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference to troubleshoot the problems.

2. Complete the Process Center stand-alone profile configuration by performing one of the following tasks, depending on whether you chose to postpone actual database configuration by producing scripts to be run manually.

   - If you completed configuration of the Performance Data Warehouse database and the Process Center database using the Profile Management Tool, complete the following steps:

     a. Ensure that the **Launch the First steps console** option is selected.

     b. Click **Finish** to exit.

     c. Close the Profiles page, which is open in a separate window.

     d. Use the First steps console to start the server.

   - If you chose to postpone actual database configuration by producing scripts to be run manually, perform the following steps:

     a. Clear the check box beside **Launch the First steps console**.

     b. Click **Finish** to close the Profile Management Tool.

     c. Close the Profiles page, which is open in a separate window.

     d. Use your site's standard database definition tools and procedures to edit and run the scripts the Profile Management Tool generated to create, or create and configure the databases.

     e. When the databases are configured, start the First steps console associated with the profile, as described in Starting the First steps console.

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into an IBM Business Process Manager profile.

Check server operation by selecting **Start the server** from the First steps console. An output window opens. If you see a message like the following message, your server is operating properly:

```
ADMU3000I: Server server1 open for e-business; process id is 3348
```

You can also check server operation by running the Installation Verification Test (IVT) from the First steps console or running the **wbi_ivt** command-line utility. This test is to verify that your deployment manager or stand-alone server installation is operating properly.

Related tasks:

"Augmenting stand-alone profiles for Process Center using the **Advanced** option" on page 524
Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Center. Selecting the **Advanced** option augments profiles with customized configuration settings.

Augmenting stand-alone profiles for Process Center with the Advanced option with databases not defined:

Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Center with customized configuration settings. You can augment the stand alone profiles when you do not yet have a database set up on your system.

Use the "Augmenting stand-alone profiles for Process Center using the **Advanced** option" on page 524 procedure to get started with your advanced profile augmentation and then resume with the steps in this task.

This task describes using the **Advanced profile augmentation** option on the Profile Management Tool user interface to augment stand-alone profiles if databases are *not* defined on your system and need to be defined. The Database Design page is displayed.

Remember: All IBM Business Process Manager profiles have administrative security enabled.

1. Optional: Configure the databases using a design file. This option is available for both stand-alone server and deployment manager profiles created using the **Advanced** option.

   Note: Using a database design file is *not* recommended for IBM Business Process Manager Express configurations.

   a. Select **Use a database design file for database configuration**.

   b. Click **Browse**.

   c. Specify the fully qualified path name for the design file.

   d. Click **Next**.

   If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration. For more information on using a design file for database configuration, see "Creating database design files by using the database design tool" on page 569.

2. From the Process Center Configuration page, set the values for the following parameters:

   Environment name
   Specifies the name of the environment you are configuring.

   Select an environment type
   The environment type refers to how the Process Center is used. Select one of the following options:

   Production
   Select **Production** if the server is to be used in a production capacity.

   Stage   Select **Stage** if the server is to be a staging platform to be used as a preproduction server.

   Test

   Select **Test** if the server is to be used as a testing environment.

   For example, load testing might be done on a test server, while a stage environment type might be used as a temporary location to host changes before putting those changes into production. You might specify **Stage** as the **Environment type** if the Process Center you are configuring is to be accessed and used to review content and new functionality.

   Process Center configuration information
   Configure the Process Server that the Process Center will connect to. Set parameters for the following fields:

   Use server offline
   Indicate whether the server you are configuring is an offline server. An offline server is a Process Server that is not connected to the Process Center. Offline servers can be used when deploying snapshots of process applications. However the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

   Protocol
   Select either **http://** or **https://** as the connection protocol to the Process Server.

   Host name

   Type the host or virtual host that this Process Center needs to communicate with Process Server. Use a fully qualified host name.

   In an environment with a load balancer or proxy server between the Process Center and the Process Server services, make sure that what you designate here matches the URL for accessing the Process Server.

   Port   Type the port number of the Process Server. In an environment with a load balancer or

proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Server.

You can test the connection to the Process Server by clicking **Test Connection**.

Click **Next**.

3. On the Database Configuration - Part 1 page, perform the following actions:
   - From the **Select a database product** drop-down, select the database product to be used by the profile.
   - Select whether to **Create a new local database** or to **Use an existing local or remote database**.

     If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

     If the database product you are using with the software already exists, select **Use an existing local or remote database** .
   - In the **Process Server database name** field, enter a name for the database or accept the **BPMDB** default value.
   - In the **Performance Data Warehouse database name** field, enter a name for the database or accept the **PDWDB** default value.
   - Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

     If you do not select the checkbox, the scripts are output to the default directory.
   - Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

4. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. See the *Database configuration parameters for Profile Management Tool configuration* information for details about each required field.

   You can configure parameters for the following databases:

   DB2   After setting the values for the DB2 Database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Augmentation Summary page.

   DB2 Universal Database for z/OS
   You cannot create a new database using DB2 Universal Database for z/OS. The Performance Data Warehouse database and, for a stand-alone server profile, the Process Server database, must exist. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration - Part 2 page, click **Next** to go to the Profile Augmentation Summary page.

   Microsoft SQL Server
   After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Augmentation Summary page.

   Oracle   You cannot create a new database using this database.

   Important: You must have a user ID that has SYSDBA privileges before creating any profile. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Augmentation Summary page.

5. In the Profile Augmentation Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When profile augmentation completes, the Profile Augmentation Complete page is displayed with the following message: **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might display in place of the success message, for example:

The Profile Management tool augmented the profile but errors occurred
> Indicates that profile augmentation completed but errors were generated.

The Profile Management tool cannot augment the profile
> Indicates that profile augmentation failed completely.

The Profile Complete page identifies the log file to reference to troubleshoot the problems.

6. Complete the Process Center stand-alone profile configuration by performing one of the following tasks, depending on whether you chose to postpone actual database configuration by producing scripts to be run manually:

   - If you completed configuration of the Performance Data Warehouse database and the Process Center database using the Profile Management Tool, complete the following steps:
     a. Ensure that the **Launch the First steps console** option is selected.
     b. Click **Finish** to exit.
     c. Close the Profiles page, which is open in a separate window.
     d. Use the First steps console to start the server.

   - If you chose to postpone actual database configuration by producing scripts to be run manually, perform the following steps:
     a. Clear the **Launch the First steps console** check box.
     b. Click **Finish** to close the Profile Management Tool.
     c. Close the Profiles page that is open in a separate window.
     d. Use your site's standard database definition tools and procedures to edit and run the scripts the Profile Management Tool generated to create, or create and configure the databases.
     e. When the databases are configured, start the First steps console associated with the profile, as described in Starting the First steps console for more information.

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into an IBM Business Process Manager profile.

Check server operation by selecting **Start the server** from the First steps console. An output window opens. If you see a message like the following message, your server is operating properly:

```
ADMU3000I: Server server1 open for e-business; process id is 3348
```

You can also check server operation by running the Installation Verification Test (IVT) from the First steps console or running the **wbi_ivt** command-line utility. This test is to verify that your deployment manager or stand-alone server installation is operating properly.

Related tasks:

"Augmenting stand-alone profiles for Process Center using the **Advanced** option" on page 524
Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Center. Selecting the **Advanced** option augments profiles with customized configuration settings.

Augmenting Deployment Manager profiles for Process Center:

Use the Profile Management Tool graphical user interface (GUI) to augment and configure a deployment manager profile for Process Center.

Understand the concepts of profiles, including the differences between stand-alone, network deployment and custom profiles.

- Review the list of prerequisites for creating or augmenting profiles at "Prerequisites for creating or augmenting profiles" on page 131.

- When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/.Xdefaults file:

  ```
  Eclipse*spacing:0
  Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
  ```

  After adding the lines, run the following command before starting the Profile Management Tool:

  ```
  xrdb -load user_home/.Xdefaults
  ```

This topic assumes that you are using the Profile Management Tool to augment profiles. As a result, it is assumed that you have started the Profile Management Tool and have chosen to augment a deployment manager profile for Process Center.

The task describes the procedure for augmenting a deployment manager profile for Process Center.

As a result of following the procedure in "Augmenting profiles using the Profile Management Tool" on page 516, you are viewing the Administrative Security page. All IBM Business Process Manager profiles have security enabled.

1. Enable administrative security

   From the Administrative Security page, re-enter the administrative user ID and password for the profile you are augmenting.

   If the profile you are augmenting has the WebSphere(r) Application Server sample application deployed, it requires an account under which to run. Supply the password for the account. You cannot change the user name of the account.

2. In the Profile Augmentation Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When profile augmentation completes, the Profile Augmentation Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might display in place of the success message, for example:

   - **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
   - **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

3. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console to start the server.

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into an IBM Business Process Manager profile.

Check the server operation by selecting **Start the deployment manager** from the First steps console. An output window opens. If you see a message like the following one, your deployment manager is operating properly:

```
 ADMU3000I: Server dmgr open for e-business; process id is 3072
```

In a deployment environment, you must create and configure other databases, create custom profiles, and federate them to your deployment manager, create servers, create clusters if you want workload

management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

Augmenting custom profiles (managed nodes) for Process Center:

Use the augmentation option on the Profile Management Tool graphical user interface (GUI) to augment and configure WebSphere Application Server version 7.0 or WebSphere Application Server Network Deployment version 7.0 custom profiles (managed nodes).

You have started the Profile Management Tool and have chosen to augment a custom profile.

In this type of configuration, you can choose to federate the node to an existing deployment manager during the augmentation process, or federate it later using the **addNode** command.

1. In the Federation page, choose to federate the node into the deployment manager now as part of the profile augmentation, or at a later time and apart from profile augmentation.

   - If you choose to federate the node as part of the profile augmentation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if administrative security is enabled on the deployment manager. Leave the **Federate this node later** check box deselected. Then click **Next**.

     The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).

     Attention: Federate the custom node during profile augmentation only if all the following conditions are true:

     – You do not plan to use this custom node as a migration target.
     – No other node is being federated. (Node federation must be serialized.)
     – The deployment manager is running.
     – The deployment manager is an IBM Business Process Manager deployment manager.
     – The deployment manager is at a release level the same or higher than the release level of the custom profile you are augmenting.
     – The deployment manager has a JMX administrative port enabled. The default protocol is SOAP. (Click **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

     If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

   - If you choose to federate the node at a later time and apart from profile augmentation, select the **Federate this node later** check box and click **Next**.

   The Profile Summary page is displayed.

2. In the Profile Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   When the profile augmentation is complete, the Profile Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might appear in place of the success message, for example:

   - **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
   - **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

3. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.

You have augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a Process Center profile.

If you did not federate the profile during profile augmentation, federate it now. The node within the profile is empty until you federate the node and use the deployment manager to customize the node.

Augmenting profiles for Process Server:

You can use the **Typical** profile augmentation option or the **Advanced** profile augmentation option to augment profiles for Process Server.

Augmenting stand-alone profiles for Process Server using the Typical option:

Use the **Typical profile augmentation** option of the Profile Management Tool to augment and configure stand-alone profiles for IBM Business Process Manager Process Server. Selecting the **Typical** option augments profiles with default configuration settings.

Understand the concepts of profiles, including the differences between stand-alone, network deployment and custom profiles.

Understand the differences between the **Typical profile augmentation** option and the **Advanced profile augmentation** option, including under which scenarios you should use one over the other.
- Review the list of prerequisites for creating or augmenting profiles at "Prerequisites for creating or augmenting profiles" on page 131.

This topic assumes that you are using the Profile Management Tool to augment profiles. It is assumed that you have started the Profile Management Tool, have chosen to augment a stand-alone profile for Process Server, and have selected the **Typical profile augmentation** option.

In this type of configuration, the Profile Management Tool performs the following tasks:
- Gives option to deploy the administrative console.

Restriction: If you plan to federate the stand-alone server profile to a deployment manager, do not use the **Typical** option to create it. The default values for messaging engine storage and database type provided in a **Typical** profile augmentation are not suitable deployment environment installations. Use the **Advanced** option to augment the profile instead. See "Augmenting stand-alone profiles for Process Server using the **Advanced** option" on page 537 for instructions.

As a result of following the procedure in "Augmenting profiles using the Profile Management Tool" on page 516, you are viewing the Administrative Security page. All IBM Business Process Manager profiles have security enabled.

1. From the Administrative Security page, re-enter the administrative user ID and password for the profile you are augmenting.

   If the profile you are augmenting has the WebSphere(r) Application Server sample application deployed, it requires an account under which to run. Supply the password for the account. You cannot change the user name of the account.

2. From the Process Server Configuration page, set the values for the following parameters

   Environment name
       Specifies the name of the environment you are configuring.

   Select an environment type

Options are as follows:

- Production

  Select **Production** if the server will serve in a production capacity.

- Stage

  Select **Stage** if the server will serve as a staging platform to be used as a preproduction server.

- Test

  Select **Test** if the server you are configuring will be used as a testing environment.

The environment type refers to how the Process Server is used. For example, in what capacity will the Process Server be used - *production*, *stage* or *test* . Load testing might be done on a test server, while a stage environment type might be used as a temporary location to host changes before putting those changes into production. You might specify **Stage** as the **Environment type** if the Process Server you are configuring will be accessed and used to review content and new functionality.

Process Center configuration information

Configure the Process Center that the Process Server will connect to.

Set parameters for the following fields:

- **Use server offline**

  Indicate whether the server you are configuring is an offline server.

  An offline server is a Process Server that is not connected to the Process Center.

  Offline servers can still be used when deploying snapshots of process applications. However the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

- **Protocol**

  Select either **http://** or **https://** as the connection protocol to the Process Center.

- **Host name**

  Type the host or virtual host that this Process Server needs to communicate with Process Center. Use a fully qualified host name.

  In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port**

  Type the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.

You can test the connection to the Process Center by clicking **Test Connection**.

Click **Next**.

3. On the Database Configuration - Part 1 page, perform the following actions: When augmenting a Process Server for IBM Business Process Manager Standard, the **Performance Data Warehouse database name** and **Process server database name** fields do not display.

- From the **Select a database product** drop-down, select the database product to be used by the profile.

- Select whether to **Create a new local database** or to **Use an existing local or remote database**.

  If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

  If the database product you are using with the software already exists, select **Use an existing local or remote database** .

- In the **Process Server database name** field, enter a name for the database or accept the **BPMDB** default value.

- In the **Performance Data Warehouse database name** field, enter a name for the database or accept the **PDWDB** default value.
- Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

  If you do not select the checkbox, the scripts are output to the default directory.
- Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

4. In the Database Configuration - Part 2 page, complete the database configuration and click **Next**.

   Note: You must complete this page even if you decided not to run the database scripts. The values you choose on the Database Configuration - Part 2 page are added to the database configuration scripts stored in the directory you specified in the **Database script output directory** field on the previous page (or in the default directory for these scripts if you did not specify a different location).

   Restriction: You cannot create a new database if you are using DB2 for z/OS V8 or V9, or Oracle. In these cases, the Process Server / Performance Data Warehouse databases must exist.
   From the following list, select the link that names the database you are using.
   - DB2 Database
   - DB2 for z/OS V8, V9 and V10
   - Microsoft SQL Server
   - Oracle

   a. Set values for the DB2 Database

      The following table lists the fields you must complete on the Database Configuration (Part 2) page when you select DB2 Universal Database as your database product.

Table 89. Required database configuration fields for DB2 Database

Field	Action needed
JDBC driver	Select from the following options: • DB2 Universal • DB2 DataServer
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in the following directory: • If you selected the DB2 Express feature during the installation: `${WAS_INSTALL_ROOT}/db2/java` • If you did not select the DB2 Express feature during the installation: `${WAS_INSTALL_ROOT}/jdbcdrivers/DB2`
Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of `50000` or enter the correct server port number.

   b. Set values for the DB2 for z/OS V8, V9 and V10

      The following table lists the fields you must complete on the Database Configuration (Part 2) page when you select DB2 for z/OS V8 and V9 as your database product. You cannot create a new database using these databases. The performance data warehouse database and, for a stand-alone server profile, the process server database, must exist.

Table 90. Required database configuration fields for DB2 Universal Database for z/OS

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}/jdbcdrivers/DB2`
Database server host name (for example IP address)	Enter the database server host name.
Server port	Accept the default value of 446 or enter the correct server port number.
Process Server Schema name	Enter the database schema name for Process Server.
Performance Server Schema name	Enter the database schema name for Performance Server.
Connection location	Enter the connection location.
Storage group name	Enter the storage group name.

c. Set values for the Microsoft SQL Server

The following table lists the fields you must complete on the Database Configuration (Part 2) page when you select Microsoft SQL Server as your database product.

Table 91. Required database configuration fields for Microsoft SQL Server

Field	Action needed
JDBC driver	Select from the following options • **Microsoft SQL Server JDBC 2.0** **Note:** JDBC 3.0 is also supported. Select **Microsoft SQL Server JDBC 2.0** for JDBC 3.0. • **Microsoft SQL Server JDBC 1.2**

Process Server database	For the Process Server database, enter values for the following parameters:
	• `User name`
	  Enter the Process Server database user name.
	• `Password`
	  Enter a password to authenticate with the Process Server database.
	• `Confirm password`
	  Confirm the password by reentering it.

Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters:
	• `User name`
	  Enter the Performance Data Warehouse database user name.
	• `Password`
	  Enter a password to authenticate with the Performance Data Warehouse database.
	• `Confirm password`
	  Confirm the password by reentering it.

Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.

Table 91. Required database configuration fields for Microsoft SQL Server (continued)

Field	Action needed
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer`

    d. Set values for the Oracle database

        The following table lists the fields you must complete on the Database Configuration (Part 2) page when you select Oracle as your database product. You cannot create a new database using this database.

        Important: You must have a user ID that has SYSDBA privileges before creating any profile.

Table 92. Required database configuration fields for Oracle

Field	Action needed

Process Server database	For the Process Server database, enter values for the following parameters:
	• `User name`
	  Enter the Process Server database user name.
	• `Password`
	  Enter a password to authenticate with the Process Server database.
	• `Confirm password`
	  Confirm the password by reentering it.
	
	Note: Process Server database user name and the Performance Data Warehouse database user name cannot be the same.

Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters:
	• `User name`
	  Enter the Performance Data Warehouse database user name.
	• `Password`
	  Enter a password to authenticate with the Performance Data Warehouse database.
	• `Confirm password`
	  Confirm the password by reentering it.
	
	Note: The Performance Data Warehouse database user name and the Process Server database cannot be the same.

Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle`

5. In the Profile Augmentation Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When profile augmentation completes, the Profile Augmentation Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might display in place of the success message, for example:

   - **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
   - **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

You have completed one of the following tasks:

- Augmented a WebSphere Application Server, WebSphere Application Server Network Deployment, or WebSphere Enterprise Service Bus profile into an IBM Business Process Manager profile.
- Augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere Enterprise Service Bus profile.

Check server operation by selecting **Start the server** from the First steps console. An output window opens. If you see a message like the following message, your server is operating properly:

ADMU3000I: Server server1 open for e-business; process id is 3348

You can also check server operation by running the Installation Verification Test (IVT) from the First steps console or running the **wbi_ivt** command-line utility. This test is to verify that your deployment manager or stand-alone server installation is operating properly. For a Process Server stand-alone server profile, it also runs a System Health check and generates a report.

Augmenting stand-alone profiles for Process Server using the **Advanced** *option:*

Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Server. Select the **Advanced** option to augment profiles with customized configuration settings.

An *IBM BPM Advanced, Process Server stand-alone profile* is a defined server environment that is managed from its own console and functions independently from all other process servers. Before you begin, you should have stopped the server for this profile and launched the Profile Management Tool from your BPM Advanced installation.

This task describes how to use the **Advanced profile augmentation** option on the Profile Management Tool to augment a Stand-alone profile.

By selecting the **Advanced** option, you can perform the following tasks:
- Configure the databases using a database design file.

Important: The procedure in this topic includes all pages from the Profile Management Tool that might display when augmenting a stand-alone process server profile using the Advanced option. However, if the profile you are augmenting already includes configured components (for example, the common database or Business Space powered by WebSphere) the pages within the profile management tool for these components will not display.

Use the "Augmenting profiles using the Profile Management Tool" on page 516 procedure to get started with your profile augmentation and then resume with the following steps depending on where you left off.

Remember: All IBM Business Process Manager profiles have administrative security enabled.

1. Using the Profile Management Tool, select the profile you want to augment for Process Server and click **Augment**. The Augment Selection page opens.
2. Select the type of augmentation to apply to the profile and click **Next**. The Profile Augmentation Options page opens.
3. Select **Advanced profile augmentation** and click **Next**. The Administrative Security page opens.
4. From the Administrative Security page, re-enter the administrative user ID and password for the profile you are augmenting.

   If the profile you are augmenting has the WebSphere(r) Application Server sample application deployed, it requires an account under which to run. Supply the password for the account. You cannot change the user name of the account.
5. If you have multiple servers defined on your system, select the server. The Application Scheduler Configuration page is opened. In the **For Advanced profile augmentation when profile has multiple servers defined:** section of the Application Scheduler Configuration page, select the server from the drop-down list and click **Next**.

You have completed the initial steps to augment a stand-alone profile for Process Server using the **Advanced** option.

Complete the augmentation process depending on the following system set up:
- Databases are *not* already defined on your system.
- Databases *are* already defined on your system.

Related tasks:

"Augmenting stand-alone profiles with the **Advanced** option when databases are defined"
Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Server with customized configuration settings. You can augment these stand-alone profiles for Process Server if databases *are* already defined on your system.

"Augmenting stand-alone profiles with the **Advanced** option when databases are not defined" on page 540
Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Server with customized configuration settings. You can augment these stand-alone profiles for Process Server if databases are *not* already defined on your system.

"Loading the database with system information" on page 581
You must run the `bootstrapProcessServerData` command before you try to start or use the IBM Business Process Manager server.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Augmenting stand-alone profiles with the **Advanced** *option when databases are defined:*

Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Server with customized configuration settings. You can augment these stand-alone profiles for Process Server if databases *are* already defined on your system.

An *IBM BPM Advanced, Process Server stand-alone profile* is a defined server environment that is managed from its own console and functions independently from all other process servers. Before you begin, you

should have stopped the server for this profile and launched the Profile Management Tool from your BPM Advanced installation. Use the "Augmenting stand-alone profiles for Process Server using the **Advanced** option" on page 537 procedure to get started with your advanced profile augmentation and then resume with the steps in this task.

This task describes using the **Advanced profile augmentation** option on the Profile Management Tool user interface to augment stand-alone profiles if databases *are* already defined on your system,

Remember: All IBM Business Process Manager profiles have administrative security enabled.

1. In the Profile Augmentation Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When profile augmentation completes, the Profile Augmentation Complete page is displayed with the following message: **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might display in place of the success message, for example:

   The Profile Management tool augmented the profile but errors occurred
   > Indicates that profile augmentation completed but errors were generated.

   The Profile Management tool cannot augment the profile
   > Indicates that profile augmentation failed completely.

   The Profile Complete page identifies the log file to reference to troubleshoot the problems.

2. Complete the Process Server stand-alone profile configuration by performing one of the following tasks, depending on whether you chose to postpone actual database configuration by producing scripts to be run manually.

   - If you completed configuration of the Performance Data Warehouse database and the Process Server database using the Profile Management Tool, complete the following steps:

     a. Ensure that the **Launch the First steps console** option is selected.

     b. Click **Finish** to exit.

     c. Close the Profiles page, which is open in a separate window.

     d. Use the First steps console to start the server.

   - If you chose to postpone actual database configuration by producing scripts to be run manually, perform the following steps:

     a. Clear the check box beside **Launch the First steps console**.

     b. Click **Finish** to close the Profile Management Tool.

     c. Close the Profiles page, which is open in a separate window.

     d. Use your site's standard database definition tools and procedures to edit and run the scripts that the Profile Management Tool generated to create, or create and configure the databases.

     e. When the databases are configured, start the First steps console associated with the profile.

You have completed one of the following tasks:

- Augmented a WebSphere Application Server, WebSphere Application Server Network Deployment, or WebSphere Enterprise Service Bus profile into an IBM Business Process Manager profile.
- Augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere Enterprise Service Bus profile.

Check server operation by selecting **Start the server** from the First steps console. An output window opens. If you see a message like the following message, your server is operating properly:

```
ADMU3000I: Server server1 open for e-business; process id is 3348
```

You can also check server operation by running the Installation Verification Test (IVT) from the First steps console or running the **wbi_ivt** command-line utility. This test is to verify that your deployment manager or stand-alone server installation is operating properly. For a Process Server stand-alone server profile, it also runs a System Health check and generates a report.

Related tasks:
"Augmenting stand-alone profiles for Process Server using the **Advanced** option" on page 537
Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Server. Select the **Advanced** option to augment profiles with customized configuration settings.

Augmenting stand-alone profiles with the **Advanced** *option when databases are not defined:*

Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Server with customized configuration settings. You can augment these stand-alone profiles for Process Server if databases are *not* already defined on your system.

An *IBM BPM Advanced, Process Server stand-alone profile* is a defined server environment that is managed from its own console and functions independently from all other process servers. Before you begin, you should have stopped the server for this profile and launched the Profile Management Tool from your BPM Advanced installation. Use the "Augmenting stand-alone profiles for Process Server using the **Advanced** option" on page 537 procedure to get started with your advanced profile augmentation and then resume with the steps in this task.

Remember: All IBM Business Process Manager profiles have administrative security enabled.

1. Optional: Configure the databases using a design file. This option is available for both stand-alone server and deployment manager profiles created using the **Advanced** option.

   a. Select **Use a database design file for database configuration**.

   b. Click **Browse**.

   c. Specify the fully qualified path name for the design file.

   d. Click **Next**.

   If you choose to specify a design file, the database configuration panels in the Profile Management Tool are skipped. Instead, the design file location is passed to the command line to complete the database configuration. For more information on using a design file for database configuration, see "Creating database design files by using the database design tool" on page 569.

2. From the Process Server Configuration page, set the values for the following parameters.

   Environment name
      Specifies the name of the environment you are configuring.

   Select an environment type

      Options are as follows:
      • Production

         Select **Production** if the server will serve in a production capacity.
      • Stage

         Select **Stage** if the server will serve as a staging platform to be used as a preproduction server.
      • Test

         Select **Test** if the server you are configuring will be used as a testing environment.

      The environment type refers to how the Process Server is used. For example, in what capacity will the Process Server be used - *production*, *stage* or *test* . Load testing might be done on a test server, while a stage environment type might be used as a temporary location to host changes before

putting those changes into production. You might specify **Stage** as the **Environment type** if the Process Server you are configuring will be accessed and used to review content and new functionality.

Process Center configuration information

Configure the Process Center that the Process Server will connect to.

Set parameters for the following fields:

- **Use server offline**

  Indicate whether the server you are configuring is an offline server.

  An offline server is a Process Server that is not connected to the Process Center.

  Offline servers can still be used when deploying snapshots of process applications. However the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

- **Protocol**

  Select either **http://** or **https://** as the connection protocol to the Process Center.

- **Host name**

  Type the host or virtual host that this Process Server needs to communicate with Process Center. Use a fully qualified host name.

  In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port**

  Type the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.

- **User name**

  Type the user name that you want to use to log into the Process Center.

- **Password**

  Type the password that you want to use to log into the Process Center.

- **Confirm Password**

  Confirm the password that you want to use to log into the Process Center.

You can test the connection to the Process Center by clicking **Test Connection**.

Click **Next**.

3. On the Database Configuration - Part 1 page, perform the following actions:

- From the **Select a database product** drop-down, select the database product to be used by the profile.

- Select whether to **Create a new local database** or to **Use an existing local or remote database**.

  If you selected DB2 as the database product, you can select to create a new database and the Profile Management Tool will create a new DB2 database, as DB2 is embedded with the software.

  If the database product you are using with the software already exists, select **Use an existing local or remote database** .

- In the **Process Server database name** field, enter a name for the database or accept the **BPMDB** default value.

- In the **Performance Data Warehouse database name** field, enter a name for the database or accept the **PDWDB** default value.

- Select the **Override the default output directory for database scripts** checkbox if you want to set the directory into which the sql scripts used to create the database tables are written.

  If you do not select the checkbox, the scripts are output to the default directory.

- Select the **Run database scripts to initialize the databases** checkbox if you want to run the database scripts automatically (as part of the profile creation process). If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

4. In the Database Configuration - Part 2 page, complete the configuration of the database you selected. See the *Database configuration parameters for Profile Management Tool configuration* information for details about each required field.

   You can configure parameters for the following databases:

   DB2   After setting the values for the DB2 Database on the Database Configuration (Part 2) page, click **Next** to go to the Profile Augmentation Summary page.

   DB2 Universal Database for z/OS
   > You cannot create a new database using DB2 Universal Database for z/OS. The Performance Data Warehouse database and, for a stand-alone server profile, the Process Server database, must exist. After setting the values for the DB2 Universal Database for z/OS on the Database Configuration (Part 2) page, click **Next** to go to the Profile Augmentation Summary page.

   Microsoft SQL Server
   > After setting the values for the Microsoft SQL Server database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Augmentation Summary page.

   Oracle You cannot create a new database using this database.

   > **Important:** You must have a user ID that has SYSDBA privileges before creating any profile. After setting the values for the Oracle database on the Database Configuration - Part 2 page, click **Next** to go to the Profile Augmentation Summary page.

5. In the Profile Augmentation Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile. The progress of the configuration displays on the Profile Configuration Progress window.

   When profile augmentation completes, the Profile Augmentation Complete page is displayed with the following message: `The Profile Management tool augmented the profile successfully.`

   Attention: If errors are detected during profile augmentation, other messages might display in place of the success message, for example:

   The Profile Management tool augmented the profile but errors occurred
   > Indicates that profile augmentation completed but errors were generated.

   The Profile Management tool cannot augment the profile
   > Indicates that profile augmentation failed completely.

   The Profile Complete page identifies the log file to reference to troubleshoot the problems.

6. Complete the Process Server stand-alone profile configuration by performing one of the following tasks, depending on whether you chose to postpone actual database configuration by producing scripts to be run manually.

   - If you completed configuration of the Performance Data Warehouse database and the Process Server database using the Profile Management Tool, perform the following steps:

     a. Ensure that the **Launch the First steps console** option is selected.

     b. Click **Finish** to exit.

     c. Close the Profiles page, which is open in a separate window.

     d. Use the First steps console to start the server.

   - If you chose to postpone actual database configuration by producing scripts to be run manually, perform the following steps:

     a. Clear the check box beside **Launch the First steps console**.

     b. Click **Finish** to close the Profile Management Tool.

c. Close the Profiles page, which is open in a separate window.

d. Use your site's standard database definition tools and procedures to edit and run the scripts that the Profile Management Tool generated to create, or create and configure the databases.

e. When the databases are configured, start the First steps console associated with the profile.

You have completed one of the following tasks:

- Augmented a WebSphere Application Server, WebSphere Application Server Network Deployment, or WebSphere Enterprise Service Bus profile into an IBM Business Process Manager profile.

- Augmented a WebSphere Application Server or WebSphere Application Server Network Deployment profile into a WebSphere Enterprise Service Bus profile.

Check server operation by selecting **Start the server** from the First steps console. An output window opens. If you see a message like the following message, your server is operating properly:

```
ADMU3000I: Server server1 open for e-business; process id is 3348
```

You can also check server operation by running the Installation Verification Test (IVT) from the First steps console or running the `wbi_ivt` command-line utility. This test is to verify that your deployment manager or stand-alone server installation is operating properly. For a Process Server stand-alone server profile, it also runs a System Health check and generates a report.

Related tasks:

"Augmenting stand-alone profiles for Process Server using the **Advanced** option" on page 537
Use the **Advanced profile augmentation** option on the Profile Management Tool graphical user interface (GUI) to augment stand-alone profiles for Process Server. Select the **Advanced** option to augment profiles with customized configuration settings.

Augmenting deployment manager profiles for Process Server:

Use the Profile Management Tool graphical user interface (GUI) to augment and configure a deployment manager profile for Process Server or .

Understand the concepts of profiles, including the differences between stand-alone, network deployment and custom profiles.

- Review the list of prerequisites for creating or augmenting profiles at "Prerequisites for creating or augmenting profiles" on page 131.

- Solaris When you use the Profile Management Tool with the Motif graphical user interface on the Solaris operating system, the default size of the Profile Management Tool might be too small to view all the messages and buttons. To fix the problem, add the following lines to the *install_root*/ `.Xdefaults` file:

```
Eclipse*spacing:0
Eclipse*fontList:-misc-fixed-medium-r-normal-*-10-100-75-75-c-60-iso8859-1
```

After adding the lines, run the following command before starting the Profile Management Tool:

```
xrdb -load user_home/.Xdefaults
```

This topic assumes that you are using the Profile Management Tool to augment profiles. As a result, it is assumed that you have started the Profile Management Tool, have chosen to augment a deployment manager profile for Process Server.

The task describes the procedure for augmenting a deployment manager profile for Process Server.

As a result of following the procedure in "Augmenting profiles using the Profile Management Tool" on page 516, you are viewing the Administrative Security page. All IBM Business Process Manager profiles have security enabled.

1. Enable administrative security

From the Administrative Security page, re-enter the administrative user ID and password for the profile you are augmenting.

If the profile you are augmenting has the WebSphere(r) Application Server sample application deployed, it requires an account under which to run. Supply the password for the account. You cannot change the user name of the account.

2. In the Profile Augmentation Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   The progress of the configuration displays on the Profile Configuration Progress window.

   When profile augmentation completes, the Profile Augmentation Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might display in place of the success message, for example:

   - **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
   - **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

3. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console to start the server.

The node defined by the profile has a deployment manager named **Dmgr**.

Check the server operation by selecting **Start the deployment manager** from the First steps console. An output window opens. If you see a message like the following one, your deployment manager is operating properly:

```
 ADMU3000I: Server dmgr open for e-business; process id is 3072
```

In a deployment environment, you must create and configure other databases, create custom profiles, and federate them to your deployment manager, create servers, create clusters if you want workload management capabilities, and perform other tasks specific to your planned installation environment. Your planned environment dictates which tasks you must perform and the order in which you perform them.

Augmenting custom profiles (managed nodes) for Process Server:

Use the Profile Management Tool to augment and configure WebSphere Application Server version 7.0, WebSphere Application Server Network Deployment version 7.0, or WebSphere Enterprise Service Bus version 7.0 custom profiles (managed nodes) into Process Server profiles.

You have started the Profile Management Tool and have chosen to augment a custom profile.

In this type of configuration, you can choose to federate the node to an existing deployment manager during the augmentation process, or federate it later using the **addNode** command.

1. In the Federation page, choose to federate the node into the deployment manager now as part of the profile augmentation, or at a later time and apart from profile augmentation.

   - If you choose to federate the node as part of the profile augmentation, specify the host name or IP address and SOAP port of the deployment manager, and an authentication user ID and password if administrative security is enabled on the deployment manager. Leave the **Federate this node later** check box deselected. Then click **Next**.

The Profile Management Tool verifies that the deployment manager exists and can be contacted, and that the authentication user ID and password are valid for that deployment manager (if it is secured).

Attention: Federate the custom node during profile augmentation only if all the following conditions are true:

- You do not plan to use this custom node as a migration target.
- No other node is being federated. (Node federation must be serialized.)
- The deployment manager is running.
- The deployment manager is an IBM Business Process Manager deployment manager.
- The deployment manager is at a release level the same or higher than the release level of the custom profile you are augmenting.
- The deployment manager has a JMX administrative port enabled. The default protocol is SOAP. (Click **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

If you attempt to federate a custom node when the deployment manager is not running or is not available for other reasons, a warning box prevents you from continuing. If this warning box appears, click **OK** to exit from it, and then make different selections on the Federation page.

- If you choose to federate the node at a later time and apart from profile augmentation, select the **Federate this node later** check box and click **Next**.

The Profile Summary page is displayed.

2. In the Profile Summary page, click **Augment** to augment the profile or **Back** to change the characteristics of the profile.

   When the profile augmentation is complete, the Profile Complete page is displayed with the message **The Profile Management tool augmented the profile successfully**.

   Attention: If errors are detected during profile augmentation, other messages might appear in place of the success message, for example:

   - **The Profile Management tool augmented the profile but errors occurred**, which indicates that profile augmentation completed but errors were generated.
   - **The Profile Management tool cannot augment the profile**, which indicates that profile augmentation failed completely.

   The Profile Complete page identifies the log file to reference in order to troubleshoot the problems.

3. In the Profile Complete page, ensure that **Launch the First steps console** is selected and click **Finish** to exit. Also, close the Profiles page, which is open in a separate window. Use the First steps console access the product documentation.

If you did not federate the profile during profile augmentation, federate it now. The node within the profile is empty until you federate the node and use the deployment manager to customize the node.

Augmenting profiles using the manageprofiles command-line utility:

Augmentation is the ability to change an existing profile with an augmentation template. You can augment existing WebSphere Application Server or WebSphere Application Server Network Deployment profiles into IBM Business Process Manager profiles.You can augment existing WebSphere Application Server or WebSphere Application Server Network Deployment profiles into WebSphere Enterprise Service Bus profiles. You can augment a profile by using the `manageprofiles` command-line utility.

Before using this procedure, ensure that you have done the following tasks:

- You have reviewed the list of prerequisites for creating or augmenting a profile at Prerequisites for creating or augmenting profiles.
- You have shut down any servers associated with the profile that you plan to augment.

- If you plan to augment a stand-alone server or custom profile, you have determined if it has already been federated to a deployment manager:
  – If the profile you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.
  – If the profile you want to augment has not already been federated to a deployment manager, when you do federate it later by using the **addNode** command, the following must be true of the deployment manager with which it is federated in order for the augmentation to complete successfully:
    - It must be running.
    - It must be an IBM Business Process Manager deployment manager profile.
    - It must be at a release level the same or higher than that of the profile you are augmenting.
      • IBM Business Process Manager profiles cannot use a WebSphere Enterprise Service Bus deployment manager.
      • WebSphere Enterprise Service Bus profiles can use a WebSphere Enterprise Service Bus deployment manager or IBM Business Process Manager deployment manager.
    - It must have a JMX administrative port enabled. The default protocol is SOAP.
- You have reviewed the manageprofiles parameters and default values in manageprofiles parameters.
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Security role required for this task: See Granting write permission of files and directories to nonroot users for profile creation .

To use the **manageprofiles** command-line utility to augment a profile, perform the following steps.

1. Determine the template that the existing profile was created with (deployment manager, stand-alone, or managed). You can determine the template that was used for creating the profile by viewing the profile registry in *install_root*/properties/profileRegistry.xml. Do not modify this file, use it only to view the templates.

2. Find the appropriate template to augment to. You can augment an existing WebSphere Application Server or WebSphere Application Server Network Deployment profile into an IBM Business Process Manager or WebSphere ESB profile.

   Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates.

   The following profile templates are available:
   - default.procctr: for a Process Center stand-alone profile, which defines a stand-alone server with capabilities and functionality specific to IBM BPM Standard or Express.
   - default.procsvr: for a Process Server stand-alone profile, which defines a stand-alone servers with capabilities and functionality specific to IBM BPM Standard or Express.
   - dmgr.procctr: for a Process Center deployment manager profile for IBM BPM Standard, which defines a deployment manager. A *deployment manager* provides one administrative interface to a logical group of servers on one or more workstations.
   - dmgr.procsvr: for a Process Server deployment manager profile for IBM BPM Standard, which defines a deployment manager. A *deployment manager* provides one administrative interface to a logical group of servers on one or more workstations.
   - managed.procctr for a Process Center custom profile for IBM BPM Standard or Express, which, when federated to a deployment manager, defines a managed node. If you have decided that your solution requires a deployment environment, your runtime environment requires one or more managed nodes. A *custom profile* contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node. Do not federate a node unless the deployment manager you are federating to is at a release level the same or higher than that of the custom profile you are creating.

- `managed.procsvr`: for a Process Server custom profile for IBM BPM Standard, which, when federated to a deployment manager, defines a managed node. If you have decided that your solution requires a network deployment configuration, your runtime environment requires one or more managed nodes. A *custom profile* contains an empty node that you must federate into a deployment manager cell to make operational. Federating the custom profile changes it into a managed node. Do not federate a node unless the deployment manager you are federating to is at a release level the same or higher than that of the custom profile you are creating.

  Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

  Note: Do not manually modify the files that are located in the *install_dir*/profileTemplates directory.

3. Create a response file. Do not supply a **-profilePath** parameter, and replace "create" with "augment" when creating the response file. For the complete list of parameters and their default values, see manageprofiles parameters.

   Sample response files are available under the install_root/BPM/samples/manageprofiles/ directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server) and customize the file for your environment.

   Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. See Creating a response file using the IBM Business Process Manager samples.

4. Run the **manageprofiles** command from the command line, using the **-response** parameter and the name of the response file that you created.

   - `Linux` `UNIX` **manageprofiles.sh -response** *myResponseFile*
   - `Windows` **manageprofiles.bat -response** *myResponseFile*

   The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

You can see that your profile augmentation completed successfully if you receive a `INSTCONFSUCCESS:` `Profile augmentation succeeded.` message, and you can check the following log file:

- `Linux` `UNIX` *install_root*/logs/manageprofiles/*profile_name*_augment.log
- `Windows` *install_root*\logs\manageprofiles*profile_name*_augment.log

Run the Installation Verification Test (IVT) tool to verify that the profile was augmented successfully. To do this, run the following command:

- `Linux` `UNIX` **On Linux and UNIX platforms:** *profile_root*/bin/wbi_ivt.sh
- `Windows` **On Windows platforms:** *profile_root*\bin\wbi_ivt.bat

Augmenting Process Center stand-alone profiles using the manageprofiles command-line utility:

Augmentation is the ability to change an existing profile with an augmentation template. You can augment a profile from the command line using the **manageprofiles** command-line utility.

- You have reviewed the list of prerequisites for creating or augmenting a profile at "Prerequisites for creating or augmenting profiles" on page 131.

- You have shut down any servers associated with the profile that you plan to augment.
- If you plan to augment a stand-alone server or custom profile, you have determined if it has already been federated to a deployment manager:
  - If the profile you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.
  - If the profile you want to augment has not already been federated to a deployment manager, when you do federate it via the **addNode** command later, the following must be true of the deployment manager with which it is federated in order for the augmentation to complete successfully:
    - It must be running.
    - It must be at a release level the same or higher than that of the profile you are augmenting.
    - It must have a JMX administrative port enabled. The default protocol is SOAP.
    - It must have already been augmented into an IBM Business Process Manager or WebSphere Enterprise Service Bus profile, depending on the product you have installed.
- You have reviewed example profile augmentation commands in this section.
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Security role required for this task: See "Granting write permission of files and directories to nonroot users for profile creation" on page 132.

Determine the template that the existing profile was created with (deployment manager, stand-alone, or managed). You can determine the template that was used for creating the profile by viewing the profile registry in *install_root*/properties/profileRegistry.xml. Do not modify this file, use it only to view the templates. For this procedure it is assumed that you are augmenting a Process Center stand-alone profile.

This task describes how to use **manageprofiles** to augment Process Center stand-alone profiles.

To use the **manageprofiles** command-line utility to augment a stand-alone profile, perform the following steps.

1. Find the appropriate template to augment to. You can augment an existing WebSphere Application Server or WebSphere Application Server Network Deployment profile into an IBM Business Process Manager or WebSphere ESB profile.

   Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates.

   The following profile templates are available:
   - default.procctr: for a Process Server stand-alone profile, which defines a stand-alone server that includes the capabilities and functionality of IBM Business Process Manager Standard or Express.

   Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

   Note: Do not manually modify the files that are located in the *install_dir*/profileTemplates/BPM directory.

2. Create a response file. Do not supply a **-profilePath** parameter, and replace "create" with "augment" when creating the response file. For the complete list of parameters and their default values, see manageprofiles parameters.

Sample response files are available under the `install_root/BPM/samples/manageprofiles/` directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server) and customize the file for your environment.

Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. See Creating a response file using the IBM Business Process Manager samples.

3. Run the **manageprofiles** command from the command line, using the **-response** parameter and the name of the response file that you created.

   - [Linux] [UNIX] **manageprofiles.sh -response** *myResponseFile*

   - [Windows] **manageprofiles.bat -response** *myResponseFile*

   The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

Augmenting Process Server stand-alone profiles using the manageprofiles command-line utility:

Augmentation is the ability to change an existing profile with an augmentation template. You can augment existing WebSphere Application Server or WebSphere Application Server Network Deployment profiles into IBM Business Process Manager profiles. You can augment a profile from the command line using the **manageprofiles** command-line utility.

- You have reviewed the list of prerequisites for creating or augmenting a profile at "Prerequisites for creating or augmenting profiles" on page 131.
- You have shut down any servers associated with the profile that you plan to augment.
- If you plan to augment a stand-alone server or custom profile, you have determined if it has already been federated to a deployment manager:
  - If the profile you want to augment has already been federated to a deployment manager, you cannot augment it using the **manageprofiles** command-line utility.
  - If the profile you want to augment has not already been federated to a deployment manager, when you do federate it via the **addNode** command later, the following must be true of the deployment manager with which it is federated in order for the augmentation to complete successfully:
    - It must be running.
    - It must be at a release level the same or higher than that of the profile you are augmenting.
    - It must have a JMX administrative port enabled. The default protocol is SOAP.
    - It must have already been augmented into an IBM Business Process Manager or WebSphere Enterprise Service Bus profile, depending on the product you have installed.
- You have reviewed example profile augmentation commands in this section.
- You have verified that you are not already running the **manageprofiles** command-line utility on the same profile. If an error message is displayed, determine if there is another profile creation or augmentation action in progress. If so, wait until it completes.

Security role required for this task: See "Granting write permission of files and directories to nonroot users for profile creation" on page 132.

Determine the template that the existing profile was created with (deployment manager, stand-alone, or managed). You can determine the template that was used for creating the profile by viewing the profile registry in `install_root/properties/profileRegistry.xml`. Do not modify this file, use it only to view the templates. For this procedure it is assumed that you are augmenting a Process Server stand-alone profile.

This task describes how to use **manageprofiles** to augment Process Server stand-alone profiles.

To use the **manageprofiles** command-line utility to augment a stand-alone profile, perform the following steps.

1. Find the appropriate template to augment to. You can augment an existing WebSphere Application Server or WebSphere Application Server Network Deployment profile into an IBM Business Process Manager or WebSphere ESB profile.

   Templates for each profile are located in the *install_root*/profileTemplates/BPM directory for BPM templates.

   The following profile templates are available:
   - `default.procsvr`: for a Process Server stand-alone profile, which defines a stand-alone server that includes the capabilities and functionality of IBM Business Process Manager Standard and Express configurations.
   - `default.esbserver`: for a WebSphere Enterprise Service Bus stand-alone server profile, which defines a stand-alone server.

   Use the augment parameter to make changes to an existing profile with an augmentation template. The augment parameter causes the **manageprofiles** command-line utility to update or augment the profile identified in the **-profileName** parameter using the template in the **-templatePath** parameter. The augmentation templates that you can use are determined by which IBM products and versions are installed in your environment. Make sure that you specify the fully qualified file path for **-templatePath**, because a relative file path for the **-templatePath** parameter results in the specified profile not being fully augmented.

   Note: Do not manually modify the files that are located in the *install_dir*/profileTemplates/BPM directory.

2. Create a response file. Do not supply a **-profilePath** parameter, and replace "create" with "augment" when creating the response file. For the complete list of parameters and their default values, see manageprofiles parameters.

   Sample response files are available under the `install_root/BPM/samples/manageprofiles/` directory. Choose an appropriate response file based on the profile that you want to create (stand-alone or network, Process Center or Process Server) and customize the file for your environment.

   Important: Ensure that you do not leave blank spaces after the values; for example **'personalCertValidityPeriod=1 '** or **'winserviceCheck=false '**. Spaces will cause profile creation to fail. See Creating a response file using the IBM Business Process Manager samples.

3. Run the **manageprofiles** command from the command line, using the **-response** parameter and the name of the response file that you created.

   - Linux   UNIX   **manageprofiles.sh -response** *myResponseFile*

   - Windows   **manageprofiles.bat -response** *myResponseFile*

   The command displays status as it runs. Wait for it to finish. Normal syntax checking on the response file applies as the file is parsed like any other response file. Individual values in the response file are treated as command-line parameters.

You can see that your profile augmentation completed successfully if you receive a `INSTCONFSUCCESS: Profile augmentation succeeded.` message, and you can check the following log file:

- Linux   UNIX   *install_root*/logs/manageprofiles/*profile_name*_augment.log

- Windows   *install_root*\logs\manageprofiles*profile_name*_augment.log

Run the Installation Verification Test (IVT) tool to verify that the profile was augmented successfully. To do this, run the following command:

- Linux   UNIX   **On Linux and UNIX platforms:** *profile_root*/bin/bpm_ivt.sh

- Windows   **On Windows platforms:** *profile_root*\bin\bpm_ivt.bat

Database configuration fields for Profile Management Tool configuration:

To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Refer to one of the following tables to determine the required database parameters for your specific database type.

Table 93. Database configuration parameters for Profile Management Tool configuration

Database type
DB2 Universal Database configuration fields
DB2 Universal Database for z/OS configuration fields
Microsoft SQL Server database configuration fields
Oracle database configuration fields

DB2 Universal Database configuration fields

The following table lists the fields you must complete on the Database Configuration - Part 2 page when you select DB2 Universal Database as your database product.

Table 94. Required database configuration fields for DB2 Database

Field	Action needed
JDBC driver	Select from the following options: • DB2 Universal • DB2 DataServer
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in one of the following directories: • If you selected the DB2 Express feature during the installation: `${WAS_INSTALL_ROOT}/db2/java` • If you did not select the DB2 Express feature during the installation: `${WAS_INSTALL_ROOT}/jdbcdrivers/DB2`
Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of `50000` or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the messaging engine tables are created in that database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database.

DB2 Universal Database for z/OS configuration fields

The following table lists the fields you must complete on the Database Configuration - Part 2 page when you select DB2 Universal Database for z/OS as your database product.

Table 95. Required database configuration fields for DB2 Universal Database for z/OS

Field	Action needed
User name for database authentication	Enter the user name to authenticate with the database.
Password for database authentication	Enter a password to authenticate with the database.
Confirm password	Confirm the password.
Directory location of JDBC driver classpath files	The JDBC driver classpath files are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}/jdbcdrivers/DB2`
Database server host name (for example IP address)	Enter the database server host name.
Server port	Accept the default value of 446 or enter the correct server port number.
Schema name	Enter the database schema name.
Performance Server Schema name	Enter the database schema name for Performance Server.
Connection location	Enter the connection location.
Storage group name	Enter the storage group name.

You cannot create a new database using DB2 Universal Database for z/OS. The Performance Data Warehouse database and, for a stand-alone server profile, the Process Server database, must exist.

Microsoft SQL Server database configuration fields

The following table lists the fields you must complete on the Database Configuration - Part 2 page when you select Microsoft SQL Server as your database product.

Table 96. Required database configuration fields for Microsoft SQL Server

Field	Action needed
Select the **Apply Windows Authentication** option to indicate that you will connect to your databases using your Windows authentication information. If you select this option, the Common database, Process Server database, and Performance Data Warehouse database fields are made inactive.	

Common database	For the Common database, enter values for the following parameters:
	• `User name`
	  Enter the Common database user name.
	• `Password`
	  Enter a password to authenticate with the Common database.
	• `Confirm password`
	  Confirm the password to authenticate with the Common database.

Process Server database	For the Process Server database, enter values for the following parameters:
	• `User name`
	  Enter the Process Server database user name.
	• `Password`
	  Enter a password to authenticate with the Process Server database.
	• `Confirm password`
	  Confirm the password to authenticate with the Process Server database.

Table 96. Required database configuration fields for Microsoft SQL Server (continued)

Field	Action needed
Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters: • `User name` Enter the Performance Data Warehouse database user name. • `Password` Enter a password to authenticate with the Performance Data Warehouse database. • `Confirm password` Confirm the password to authenticate with the Performance Data Warehouse database.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}\jdbcdrivers\ SQLServer`
Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.

For stand-alone configurations only, when you configure the Process Server database, the messaging engine tables are created in that database. When you configure the Performance Data Warehouse database, the Performance Data Warehouse messaging engine tables are created in the Performance Data Warehouse database.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

• SQL Authentication:

  – When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.

  – When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.

  – The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.

• Windows Authentication:

  – With Windows Authentication, because the user and password are always going to be the same, all messaging engines must have custom schemas defined so that the tables can be created with a different schema for each messaging engine. When the messaging engines are configured, the schemas are automatically created using the Windows user. If the Windows user does not have privileges to create the schemas, they must be created before the product is configured.

  – For the messaging engine schema names, see step 2 in *Creating a stand-alone profile for use with Microsoft SQL Server*.

Oracle database configuration fields

The following table lists the fields you must complete when you select Oracle as your database product.

Table 97. Required database configuration fields for Oracle

Field	Action needed

Common database	For the Common database, enter values for the following parameters:
	
	User name
	      Enter the Common database user name.
	
	Password
	      Enter a password to authenticate with the Common database.
	
	Confirm password
	      Confirm the password by reentering it.

Process Server database	For the Process Server database, enter values for the following parameters:
	
	User name
	      Enter the Process Server database user name.
	
	Password
	      Enter a password to authenticate with the Process Server database.
	
	Confirm password
	      Confirm the password by reentering it.
	Note: The Process Server database user name and the Performance Data Warehouse database user name cannot be the same.

Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters:
	
	User name
	      Enter the Performance Data Warehouse database user name.
	
	Password
	      Enter a password to authenticate with the Performance Data Warehouse database.
	
	Confirm password
	      Confirm the password by reentering it.
	Note: The Performance Data Warehouse database user name and the Process Server database user name cannot be the same.

Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of 1521 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}\jdbcdrivers\Oracle`

Because all the components use the same Oracle instance, there is no difference in the databases used for individual components.

For stand-alone configurations only, there are some additional considerations for the users and schemas.

- When you configure the Process Server, the Process Server messaging engine tables are created using the user and password specified for the Process Server messaging engine. If these values are not specified, the user and password for the Process Server are used.
- When you configure the Performance Data Warehouse, the Performance Data Warehouse messaging engine tables are created using the user and password specified for the Performance Data Warehouse messaging engine. If these values are not specified, the user and password for the Performance Data Warehouse is used.
- The Service Component Architecture (SCA) system, SCA application, Common Event Infrastructure (CEI), and Business Process Choreographer messaging engine tables must be configured with different users and passwords. They cannot use the same user and password as any other messaging engine.

Related tasks:

"Creating Process Center stand-alone profiles using the Profile Management Tool" on page 137
Use the Profile Management Tool to create Process Center stand-alone profiles.

"Creating a stand-alone profile for use with Microsoft SQL Server" on page 561
You can create a Process Server stand-alone profile for use with Microsoft SQL Server by passing the required parameters using the Profile Management Tool or the manageprofiles command-line utility. You can also create the profile using the design file, which is generated by the database design tool (DDT).

Configuring databases

Before starting a profile, you must have configured the databases that are to be used with the profile.

You must have planned your database requirements, including a list of all databases and schema names. For more information, see Planning your database configuration

Creating databases using the provided scripts

If you did not create databases while creating profiles, database script files are generated for you to run afterward.

You can create the databases as part of creating a stand-alone or deployment manager profile, you can use the database design tool (dbDesignGenerator), you can manually create the databases, or you can create the databases by running the database script files after profile creation. In a network deployment environment, you must create the databases before starting the deployment manager and creating custom profiles.

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

To create the databases by running the generated SQL scripts:

1. Find the scripts in the following directories:
   - For the Process Server database:

     install_root\dbscripts\ProcessServer*db_type*\createDatabase.sql
   - For the Performance Data Warehouse database:

     install_root\dbscripts\PerformanceDW*db_type*\createDatabase.sql
2. Make any required changes to the appropriate script. Look for @ to find out where changes are needed. For example, replace **@DB_NAME@** with the database name you would like.

   Note: For SQL Server, ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

   ```
   org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'message.routingCache'
   defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
   org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
   ```

```
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.

Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement(SQLServerPreparedStatement.java:38
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute(SQLServerPreparedStatement.java:338
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

3. Run the appropriate script and make sure that it completes without errors.

Related tasks:

"Creating the Process Center or Process Server database tables" on page 558
Run the SQL scripts to create the database tables for your IBM Process Center or IBM Process Server
configuration.

"Creating the Performance Data Warehouse database tables" on page 559
Run the SQL scripts to create the database tables for your IBM Performance Data Warehouse.

"Configuring XA transactions" on page 560
You must configure XA transactions after the database is installed and before you start the server. The
Microsoft SQL Server JDBC Driver provides support for Java Platform, Enterprise Edition/JDBC 2.0
optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can
participate in standard distributed transaction processing environments such as Java Platform, Enterprise
Edition (Java EE) application servers.

Creating databases manually

IBM Business Process Manager Standard requires a Process Server database and Performance Data
Warehouse database.. You must make sure the databases are available before you install IBM Business
Process Manager Standard using a typical installation.

The Process Server and Performance Data Warehouse require their own separate databases and cannot be
configured on the same database as the other BPM components.

These instructions are for DB2 and SQL Server only. For Oracle, you can use a single instance of Oracle
for configuring BPM. The Oracle instance must exist and be available for access. Consult the Oracle
documentation to create an Oracle instance. If you use a single Oracle instance, make sure that you use
different user IDs for the different BPM databases.

Use the following command to create the databases (replace the @ placeholder strings with the
appropriate values):

• **For DB2:**

```
create database @DB_NAME@ automatic storage yes  using codeset UTF-8 territory US pagesize 32768;
connect to @DB_NAME@;
-- A user temporary tablespace is required to support stored procedures in BPM
create user temporary tablespace usrtmpspc1;
grant dbadm on database to user @DB_USER@;
UPDATE DB CFG FOR @DB_NAME@ USING LOGFILSIZ 4096 DEFERRED;
UPDATE DB CFG FOR @DB_NAME@ USING LOGSECOND 64 DEFERRED;
connect reset;
```

• **For SQL Server:**

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database, and **database_name** is the name of the database that you are creating. **COLLATE SQL_Latin1_General_CP1_CI_AS** is included because these databases must be case-insensitive.

You must configure XA transactions after the Microsoft SQL Server database is installed and before you start the server.

Note: For SQL Server, ensure that the databases you create for Process Server and Performance Data Warehouse are case insensitive. The letters **CI** in the **COLLATE** attribute value specifies this. Ensure this variable looks like this: **SQL_Latin1_General_CP1_CI_AS** (not **SQL_Latin1_General_CP1_CS_AS**). Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.

Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement(SQLServerPreparedStatement.java:
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute(SQLServerPreparedStatement.java:3
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

When using Microsoft SQL Server as the database for the Performance Data Warehouse, the following error message might be displayed during report generation using optimizer or portal for an existing process application. This prevents the report from being displayed.

```
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'slathresholdtraversals'.
    at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteCursored(SQLServerStatement.java:1824)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement.doExecuteStatement(SQLServerStatement.java:760)
    at com.microsoft.sqlserver.jdbc.SQLServerStatement$StmtExecCmd.doExecute(SQLServerStatement.java:685)
```

In Microsoft SQL server, the default schema name associated with a user must be the same as the user name. For example, if the Performance Data Warehouse database user name is *perfDB* then the default schema name associated with the user *perfDB* must also be named *perfDB*. You must create an ordinary database user and assign the required rights to the user instead of using a super user, such as *sa*. This is because the default schema for the super user is dbo and this cannot be changed.

You can complete the following steps if existing tables are not associated with a schema that is the same as the user name.

1. In SQL Server Management Studio Object Explorer, right-click the table name and then click **Design**.
2. From the Design view, press F4 to view the Properties window.
3. From the Properties window, update the schema name.

4. Right-click the tab and select **Close** to close the Design view.

5. Click **OK** when prompted to save. The selected table is transferred to the schema.

6. Repeat the previous steps for all the tables in the Performance Data Warehouse database.

Related tasks:

"Creating the Process Center or Process Server database tables"
Run the SQL scripts to create the database tables for your IBM Process Center or IBM Process Server configuration.

"Creating the Performance Data Warehouse database tables" on page 559
Run the SQL scripts to create the database tables for your IBM Performance Data Warehouse.

"Configuring XA transactions" on page 560
You must configure XA transactions after the database is installed and before you start the server. The Microsoft SQL Server JDBC Driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Creating the Process Center or Process Server database tables

Run the SQL scripts to create the database tables for your IBM Process Center or IBM Process Server configuration.

You have performed all of the steps required for your configuration and have selected not to run the database scripts to initialize the databases during the configuration. You have also created the database that you want to create the Process Center or Process Server database tables in. For information about creating the database, see *Creating databases*

To run the Process Center or Process Server SQL scripts that were produced when you completed your configuration follow the steps below:

Note: The Process Server does not support schema. You must login as the user and then execute the SQL file.

1. Go to the directory that holds the SQL scripts for creating the database tables.

   By default, SQL scripts are output to *profile_root*/dbscripts/ProcessServer/*DB_product* at profile creation time. For example,

   profile_root/dbscripts/ProcessServer/DB2

2. Run the following command as a root user or administrator.

   - Linux UNIX **configProcessServerDB.sh createDB**

   - Windows **configProcessServerDB.bat createDB**

   Note: The **createDB** parameter is used to create a local database. Skip this parameter if you have already created the required database. The **createDB** parameter is not applicable for an Oracle database and must be excluded from the command.

3. For an SQL database, create the tables for Process Center or Process Server.

   From the directory navigated to in step 1, run **createTable_ProcessServer.sql** on the database you configured for the Process Center or Process Server (BPMDB).

   Make sure all SQL statements succeed without any errors.

4. For an SQL database, create the table procedures for Process Center or Process Server.

   From the directory navigated to in step 1, run **createProcedure_ProcessServer.sql** on the database you configured for the Process Center or Process Server (BPMDB).

   Make sure all SQL statements succeed without any errors.

5. Reset the connection.

You have created the database tables for your Process Center or Process Server configuration and have completed the database configuration.

You can now load the database with system information and start the servers or clusters on your Process Center or Process Server configuration.

Related tasks:

"Creating databases using the provided scripts" on page 555
If you did not create databases while creating profiles, database script files are generated for you to run afterward.

"Creating databases manually" on page 556
IBM Business Process Manager Standard requires a Process Server database and Performance Data Warehouse database.. You must make sure the databases are available before you install IBM Business Process Manager Standard using a typical installation.

Creating the Performance Data Warehouse database tables

Run the SQL scripts to create the database tables for your IBM Performance Data Warehouse.

You have performed all of the steps required for your configuration and have selected not to run the database scripts to initialize the databases during the configuration. You have also created the database in which you want to create the Performance Data Warehouse database tables. For information about creating the database, see *Creating databases*.

To run the Performance Data Warehouse SQL scripts that were produced when you completed your configuration follow the steps below:

Note: The Performance Data Warehouse does not support schema. You must login as the user and then execute the SQL file.

1. Go to the directory that holds the SQL scripts for creating the database tables.

   By default, SQL scripts are output to *profile_root*/dbscripts/PerformanceDW/*DB_product* at profile creation time. For example,

   profile_root/dbscripts/PerformanceDW/DB2

2. Run the following command as the root user or administrator.

   - `Linux` `UNIX` **configPerformanceDWDB.sh createDB**

   - `Windows` **configPerformanceDWDB.bat createDB**

   Note: The **createDB** parameter is used to create a local database. The **createDB** parameter is not applicable for an Oracle database and must be excluded from the command.

3. For an SQL database, create the tables for the Performance Data Warehouse.

   From the directory navigated to in step 1, run **createTable_PerformanceDW.sql** on the database you configured for the Performance Data Warehouse (PDWDB).

   Make sure all SQL statements succeed without any errors.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

You can now load the database with system information and start the servers or clusters on your Performance Data Warehouse.

Related tasks:

"Creating databases using the provided scripts" on page 555
If you did not create databases while creating profiles, database script files are generated for you to run afterward.

"Creating databases manually" on page 556
IBM Business Process Manager Standard requires a Process Server database and Performance Data Warehouse database.. You must make sure the databases are available before you install IBM Business Process Manager Standard using a typical installation.

Configuring a Microsoft SQL Server database

You can create a stand-alone profile for use with Microsoft SQL Server.

Prerequisites

Before creating a profile, you must install Microsoft SQL Server on the server that hosts the database.

Database privileges and security considerations

When you create your database schemas, you must have a user ID with enough authority to create your tables. After the tables are created, the applications must have enough authority to select, insert, update, and delete information in the tables.

Table 98 shows the database privileges that are required to access the data store.

Table 98. Database privileges

Database management system	Minimum privilege required to use the data store tables	Additional privilege required to create the data store tables
Microsoft SQL Server	Configure the SQL Server for SQL Server so that authentication can be based on an SQL server login ID and password. The user ID can own the tables or be a member of a group that has sufficient authority to issue TRUNCATE TABLE statements.	The user ID requires the CREATE TABLE statement privilege.

Isolation level specifies the transaction locking behavior. You should set the isolation level to READ_COMMITTED_SNAPSHOT. You can check the isolation level of the Process Server database, the Performance Data Warehouse and the Common database with the following SQL command: **SELECT name, is_read_committed_snapshot_on FROM sys.database**. You can set the isolation level with the following SQL command: **ALTER DATABASE <database> SET READ_COMMITTED_SNAPSHOT ON**.

Related tasks:

"Configuring an existing database during a typical installation" on page 107
Use the information in this topic to determine the correct database values for configuring your existing database server during a typical installation.

Configuring XA transactions:

You must configure XA transactions after the database is installed and before you start the server. The Microsoft SQL Server JDBC Driver provides support for Java Platform, Enterprise Edition/JDBC 2.0 optional distributed transactions. JDBC connections obtained from the **SQLServerXADataSource** class can participate in standard distributed transaction processing environments such as Java Platform, Enterprise Edition (Java EE) application servers.

Failure to configure the XA transactions can result in the following error during server start up: **javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: "Could not find stored procedure 'master..xp_sqljdbc_xa_init_ex'."**..

1. The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started. To enable MS DTC for XA transactions, you must follow these steps:

On Windows XP and Windows Server 2003:

a. Select **Control Panel** > **Administrative Tools** > **Component Services**.

b. Select **Component Services** > **Computers** and right-click **My Computer**, and select **Properties**.

c. Click the **MSDTC** tab, and then click **Security Configuration**.

d. Select the **Enable XA Transactions** check box, and then click **OK**. This will cause a MS DTC service restart.

e. Click **OK** again to close the **Properties** dialog box, and then close **Component Services**.

f. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

On Windows Vista and Windows 7:

a. Select **Control Panel** > **Administrative Tools** > **Component Services**.

b. Select **Component Services** > **Computers** > **My Computer** > **Distributed Transaction Coordinator**.

c. Right-click **Local DTC** and then select **Properties**.

d. Click the **Security** tab on the **Local DTC Properties** dialog box.

e. Select the **Enable XA Transactions** check box, and click **OK**. This will restart the MS DTC service.

f. Click OK again to close the Properties dialog box, and then close Component Services.

g. Restart SQL Server to ensure that it syncs up with the MS DTC changes.

2. Configure the JDBC Distributed Transaction Components:

a. Download "Microsoft SQL Server JDBC Drive 2.0" driver from Microsoft Site using URL from Resources section.

b. Unzip archive to any folder.

c. Copy the `sqljdbc_xa.dll` file from the `JDBC unarchived` directory to the `Binn` directory of SQL Server computer. If you are using XA transactions with a 32-bit SQL Server, use the `sqljdbc_xa.dll` file in the x86 folder, even if the SQL Server is installed on a x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the `sqljdbc_xa.dll` file in the x64 folder.

d. Execute the `xa_install.sql` database script on SQL Server . This script installs the extended stored procedures that are called by `sqljdbc_xa.dll`. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You will need to run this script as an administrator of the SQL Server instance.

e. To grant permissions to a specific user to participate in distributed transactions with the JDBC driver, add the user to the SqlJDBCXAUser role in master database (e.g. for lombardi user add master database in User mappings and check SqlJDBCXAUser role).

Creating a stand-alone profile for use with Microsoft SQL Server:

You can create a Process Server stand-alone profile for use with Microsoft SQL Server by passing the required parameters using the Profile Management Tool or the manageprofiles command-line utility. You can also create the profile using the design file, which is generated by the database design tool (DDT).

Before creating a profile, complete the following prerequisites:

• Install Microsoft SQL Server on the server that hosts the database.

• Create the CMNDB (CommonDB) and BPEDB (Business Process Choreographer), BPMDB (Process Server) and PDWDB (Performance Data Warehouse) databases. The following example command is used to create a database for Microsoft SQL Server:

```
osql -b -S hostname -U db_user_account -P db_user_password
-Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
```

where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database, and **database_name** is the name of the database that you are creating. **COLLATE**

SQL_Latin1_General_CP1_CI_AS is included because the BPMDB (Process Server) and PDWDB (Performance Data Warehouse) databases must be case-insensitive.

- If Business Space is configured, it creates tables in the `CMNDB` database with a schema name of `IBMBUSSP`. To create the messaging engines tables in the `CMNDB` database, you must manually create the schemas as described in step 2 on page 566. If you are using a local database, the Common Event Infrastructure database is created when the profile is created; otherwise, you must use the script that is generated when the profile is created to create the Common Event Infrastructure database.

- If you are going to use the Common Event Infrastructure, you must create the CEI database manually. See *Configuring a Common Event Infrastructure (CEI) database*.

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

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

You can configure the CommonDB, Process Server, Performance Data Warehouse, Business Space, Service Component Architecture (SCA) System Messaging Engine and SCA Application Messaging Engine when the stand-alone profile is created.

Configure the Business Process Choreographer container, Business Process Choreographer Messaging Engine, and Business Process Choreographer Explorer using the `bpeconfig.jacl` script or the administrative console. Refer to step 3 on page 567 for information about how to configure Business Process Choreographer.

1. Create the profile using either the Profile Management Tool, the manageprofiles command-line utility, or by importing the `dbDesign` file, which is generated by the DDT.

   - To create the profile using the Profile Management Tool, see "Creating stand-alone profiles using the Profile Management Tool" on page 146. Table 99 lists the fields that you must complete on the Database Configuration (Part 2) page and Required database configuration fields for Microsoft SQL Server - Part 3.

Table 99. Required database configuration fields for Microsoft SQL Server - Part 2

Field	Action needed
JDBC driver	Select from the following options: • **Microsoft SQL Server JDBC 2.0** **Note:** JDBC 3.0 is also supported. Select **Microsoft SQL Server JDBC 2.0** for JDBC 3.0. • **Microsoft SQL Server JDBC 1.2**

Common database	For the Common database, enter values for the following parameters:
	• `User name`
	  Enter the Common database user name.
	• `Password`
	  Enter a password to authenticate with the Common database.
	• `Confirm password`
	  Confirm the password by reentering it.

Table 99. Required database configuration fields for Microsoft SQL Server - Part 2 (continued)

Field	Action needed

Process Server database	For the Process Server database, enter values for the following parameters:
	• `User name`
	  Enter the Process Server database user name.
	• `Password`
	  Enter a password to authenticate with the Process Server database.
	• `Confirm password`
	  Confirm the password by reentering it.

Performance Data Warehouse database	For the Performance Data Warehouse database, enter values for the following parameters:
	• `User name`
	  Enter the Performance Data Warehouse database user name.
	• `Password`
	  Enter a password to authenticate with the Performance Data Warehouse database.
	• `Confirm password`
	  Confirm the password by reentering it.

Database server host name (for example IP address)	Accept the default value of `localhost` or enter the correct database server host name.
Server port	Accept the default value of 1433 or enter the correct server port number.
Directory location of JDBC driver classpath files	The JDBC 2.0 driver classpath files (sqljdbc4.jar) are packaged with the product and are located in the following directory: `${WAS_INSTALL_ROOT}\jdbcdrivers\SQLServer`

Table 100. Required database configuration fields for Microsoft SQL Server - Part 3

Field	Action needed

Common Event Infrastructure bus messaging engine	For the Common Event Infrastructure bus messaging engine, enter values for the following parameters:
	• `User name`
	  Enter the Common Event Infrastructure bus messaging engine user name.
	• `Password`
	  Enter a password to authenticate with the Common Event Infrastructure bus messaging engine.
	• `Confirm password`
	  Confirm the password by reentering it.

Table 100. Required database configuration fields for Microsoft SQL Server - Part 3 (continued)

Field	Action needed

SCA application bus messaging engine	For the SCA application bus messaging engine, enter values for the following parameters:
	
	• `User name`
	  Enter the SCA application bus messaging engine user name.
	
	• `Password`
	  Enter a password to authenticate with the SCA application bus messaging engine.
	
	• `Confirm password`
	  Confirm the password by reentering it.

SCA system bus messaging engine	For the SCA system bus messaging engine, enter values for the following parameters:
	
	• `User name`
	  Enter the SCA system bus messaging engine user name.
	
	• `Password`
	  Enter a password to authenticate with the SCA system bus messaging engine.
	
	• `Confirm password`
	  Confirm the password by reentering it.

Process Server bus messaging engine	For the Process Server bus messaging engine, enter values for the following parameters:
	
	• `User name`
	  Enter the Process Server bus messaging engine user name.
	
	• `Password`
	  Enter a password to authenticate with the Process Server bus messaging engine.
	
	• `Confirm password`
	  Confirm the password by reentering it.

Performance Data Warehouse bus messaging engine	For the Performance Data Warehouse bus messaging engine, enter values for the following parameters:
	
	• `User name`
	  Enter the Performance Data Warehouse bus messaging engine user name.
	
	• `Password`
	  Enter a password to authenticate with the Performance Data Warehouse bus messaging engine.
	
	• `Confirm password`
	  Confirm the password by reentering it.

During a Process Server stand-alone profile configuration, the Profile Management Tool performs the following actions:
– Assigns customized values to ports, to the location of the profile, and to the names of the profile, node, server, host, and cell (when applicable)
– Configures the common database
– Configures the Process Server and Performance Data Warehouse Databases

– Creates the tables in each of the respective databases if the **-dbDelayConfig** parameter is set to **false**. If this parameter is set to **true**, you must manually run the SQL scripts, which are located in the *install_root*/profiles/*profile_name*/dbscripts directory.

   Note: If you are using a remote database and want to delay running the database scripts, set the **-dbDelayConfig** parameter to **true**.
– Deploys the administrative console and the WebSphere Application Server sample application
– Deploys the default application (which contains the Snoop, Hello, and HitCount servlets)
– Creates a web server definition
– Enables administrative security
– If your operating system and the privileges of your user account permit the creation of services, creates a system service to run the server

- To create the profile using the **manageprofiles** command-line utility, see "Creating Process Server stand-alone profiles using manageprofiles" on page 240.

The following example commands are used to create a stand-alone profile that includes the CommonDB, Common Event Infrastructure, Messaging Engines, and Business Space components in one database.

For Microsoft SQL Server JDBC 1.2 driver:

```
install_root/bin/manageprofiles -create -templatePath install_root/profileTemplates/BPM/
default.procsvr -dbType MSSQLSERVER_MICROSOFT -dbJDBCClasspath
path to JDBC driver -dbDriverVersion 1.2 -dbUserId userID -dbPassword password -
dbCommonForME true -dbCeiMeSchemaName
ceiSchema -dbSysMeSchemaName sysSchema
-dbAppMeSchemaName appSchema
```

For Microsoft SQL Server JDBC 2.0 driver:

```
install_root/bin/manageprofiles -create -templatePath install_root/profileTemplates/BPM/
default.procsvr -dbType MSSQLSERVER_MICROSOFT -dbJDBCClasspath
path to JDBC driver -dbDriverVersion 2.0 -dbUserId userID -dbPassword password -
dbCommonForME true -dbCeiMeSchemaName
ceiSchema -dbSysMeSchemaName sysSchema
-dbAppMeSchemaName appSchema
```

Table 101 shows the **manageprofiles** parameters that are available to configure the common database on Microsoft SQL Server.

Table 101. Available manageprofiles parameters to configure the common database using Microsoft SQL Server

Parameters	Description
-dbJDBCClasspath *jdbc_driver_location*	The default is <install_root>/jdbcdrivers/<*dbProduct*>, where **<*dbProduct*>** , where dbProduct=**SQLServer**
-dbType *db_type*	The database type. Set the following value: • **MSSQLSERVER_MICROSOFT** for a Microsoft SQL Server database using Microsoft JDBC driver V1.2 or V2.0
-dbCreateNew true | false	Indicates whether you will create or reuse a database. Valid values are **true** or **false**. Default value is **true**.
-dbDelayConfig true | false	Indicates whether you will create the table after the profile is created. Valid values are **true** or **false**. The parameter is set to **false** by default. If you are using a remote database and want to delay running the database scripts, set this parameter to **true**.

Parameters	Description
-dbDriverVersion *db_driver_version*	The database driver version. This parameter is valid only for Microsoft SQL Server. For a SQL Server database, specify either **1.2** for Microsoft SQL JDBC driver V1.2 or **2.0** for Microsoft SQL JDBC driver V2.0. If not specified, this value automatically defaults to **2.0**.
-dbHostName *db_host_name*	The database server host name or IP address. The default value is **localhost**.
-dbName *db_name*	The name of the database. The default value is **CMNDB**.
-dbOutputScriptDir *db_output_dir*	The value must be an absolute path. If you set a relative path, the SQL scripts will not be exported or executed, which will result in numerous exceptions during server startup.The location for exported database scripts. The default value is **dbscripts\CommonDB\DB2**
-dbPassword *db_pswd*	The password that is required for database authentication. This parameter is required.
-dbServerPort *db_port_number*	The database server port number. Depending on the database that you are using, you can specify a different port number instead of the default port number. The default value is **1433**.
-dbUserId *db_userid*	Specifies the user ID that has privileges to create and drop the databases. The WebSphere data source uses this ID to authenticate the database connection. For Microsoft SQL Server databases, it specifies the SQL Server user ID that will own the database tables.

- To create the database configuration by importing the dbDesign file, which is generated by the DDT, select the **Use a database design file** option, which is available for stand-alone server profiles created using the Advanced option and is displayed in the Database Configuration page of the Profile Management Tool.

  a. Select **Use a database design file** .

  b. Click **Browse**.

  c. Specify the fully qualified path name for the design file and then click **Next**.

  d. To run the database scripts automatically (as part of the profile creation process), select **Run database scripts to create database tables**. If you do not select the checkbox, you or the database administrator can run the scripts manually after profile creation completes.

     Attention: If you select **Use an existing local or remote database** and the tables already exist, make sure **Run database scripts to create database tables** is *not* selected. If both options are selected, errors occur and profile creation fails.

  e. Click **Next**

  If you choose to specify a design file, the design file location is passed to the command line to complete the database configuration. For more information about using a design file for database configuration, see "Creating database design files by using the database design tool" on page 569.

2. Create the following schemas in the CMNDB database as owner sdo using the SQL Server Studio. These schemas are required for the SIBus messaging engines.

   Important: You must configure these schemas either before you start the messaging engine or while the messaging engine is starting.

   - MEDPS00 (Process Server messaging engine)
   - MEDPE00 (Performance Data Warehouse messaging engine)

where MED is the first three characters of the message engine database name.

3. Configure Business Process Choreographer.

   a. Planning the BPEDB database

   b. Using the bpeconfig.jacl script to configure Business Process Choreographer

   c. Creating a Microsoft SQL Server database for Business Process Choreographer

4. Optional: If you chose to delay running the database scripts in the Profile Management Tool or the manageprofiles command-line utility, you must manually create the database tables by executing the following SQL scripts:

   Important: Before you can run the scripts, you must copy the *install_root*/profiles/*profile_name*/dbscripts folder to the database machine.

   a. *install_root*/profiles/*profile_name*/dbscripts/CommonDB/SQLServer/configCommonDB

   b. *install_root*/profiles/*profile_name*/dbscripts/BusinessSpace/SQLServer/configBusinessSpaceDB

   c. install_root/profiles/profile_name/dbscripts/ProcessServer

   d. install_root/profiles/profile_name/dbscripts/PerformanceDW

5. Start the server.

Related reference:

"Database configuration fields for Profile Management Tool configuration" on page 551
To create the profile for a stand-alone environment, database information is required. The required information varies, depending on the database you are using.

Creating network deployment environments for use with Microsoft SQL Server:

This topic describes how to create a network deployment environment for use with Microsoft SQL Server.

Before creating a profile, complete the following prerequisites:

- Install Microsoft SQL Server on the server that hosts the database.
- Create the BPMDB (Process Server) and PDWDB (Performance Data Warehouse) databases. The following example command is used to create a database for Microsoft SQL Server:

  ```
  osql -b -S hostname -U db_user_account -P db_user_password
  -Q "CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS"
  ```

  where **hostname** is the host name of the system that hosts SQL Server, **db_user_account** and **db_user_password** are the user name and password that are used to log in to create the database, and **database_name** is the name of the database that you are creating. **COLLATE SQL_Latin1_General_CP1_CI_AS** is included because the BPMDB (Process Server) and PDWDB (Performance Data Warehouse) databases must be case-insensitive.

- If you are going to use the Common Event Infrastructure, you must create the CEI database manually. See *Configuring a Common Event Infrastructure (CEI) database*.

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

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

You can configure the CommonDB when you create the Deployment Manager profile (IBM Business Process Manager Advanced only); however, the remaining components must be configured using the Deployment Environment panels in the administrative console. The components to be configured are:

- Common Event Infrastructure

- Business Space
- Messaging Engines
  1. Create the Deployment Manager profile Use one of the following methods:
     - Using the Profile Management Tool. For more information, see "Creating Process Server deployment manager profiles using the Profile Management Tool" on page 172.
     - Using the `manageprofiles` command. For more information, see "Creating Process Server deployment manager and custom profiles using manageprofiles" on page 408.
  2. Start the deployment manager using one of the following methods:
     - `Windows` From the **Start** menu, select **IBM** > **BPM Advanced** > **Profiles** > *profile_name* > **Start the deployment manager**.
     - In the First Steps console, click **Start the deployment manager**.
     - Use the `startManager` command.
  3. Create at least one node (managed profile) for use in the deployment environment. For more information, see "Creating Process Server deployment manager and custom profiles using manageprofiles" on page 408.
  4. Create the deployment environment:
     a. In the administrative console, select **Servers** > **Deployment Environment**.
     b. Click **New**.
     c. Provide the information for each step until the step to configure the database.
     d. On the Database page, update the default values for the components that your environment is using.

        Make sure you enter the correct values for the user name and schema name for components below. Deployment Environment configuration does not create any schemas and users as part of configuration. These should exist before the generation of the deployment environment is done. In SQL Server you need to make sure that the default schema for the user is set in the database. It is recommended that for each user you set the same value for the schema in the database - if you do not set the default schema for each user then it would be defaulted to 'dbo' and all the components would get configured with that schema resulting in a non-working environment. The database panel should have values of the schema which correspond to that user. If there is no option to enter a schema value in the field its expected that the default schema which is the same as the user would be set in the database.
     e. Because the component requires manual steps to create the required tables, the Business Space Create Tables check boxes are disabled. Create the tables for this components by following step 6 on page 569.
     f. Complete the rest of the steps to create the environment and save the settings. You can see **Servers** > **Deployment Environment** but the deployment environment is not started. Do not start the deployment environment at this time.
  5. Optional: If you cleared **Create Tables** when you created the profile in 4, generate the scripts for the message engine.
     a. In the administrative console, select **Servers** > **Deployment Environment** > *your_deployment_environment* > **Deferred Configuration**.
     b. On the command line, go to where you want to generate the scripts.
     c. Run `sibDDLGenerator.bat` utility to generate the scripts for each of the schemas required in your environment. For more information about running the utility, see the Deferred Configuration page. The schema names are the values you have chosen in the database panel above.

        ```
        sibDDLGenerator.bat -system sqlserver -version 2005 -platform windows
        -schema WPRCM00 -user user_name -statementend ; > output_script_filename
        ```

        Remember to use the correct schema, which is listed in the Deferred Configuration page, and user name. Also, redirect the result to a file. Otherwise, the generated script is printed at the command prompt instead of in a file.

Note: If you configured the databases using a database design file, it is not necessary to run the `sibDDLGenerator.bat` utility. For more information, see "Creating database design files by using the database design tool."

6. Manually create the Business Space database:

   a. In the administrative console, select **Servers** > **Deployment Environment** > *your_deployment_environment* > **Deferred Configuration**.

   b. Find the Business Space scripts.

   c. Run the `createDatabase_BusinessSpace.sql` script and then the `createTable_BusinessSpace.sql` script.

7. In the administrative console, select **Servers** > **Deployment Environment** > *your_deployment_environment* > **Deferred Configuration** and click **Configuration Done**.

8. Log off of the administrative console, shut down the deployment manager, and then shut down all of the custom profiles.

9. Optional: Clean all applicable profile logs or save them in another directory. You may want to clean or move the logs as they will be appended with the last configuration. This can make it difficult to view the most current information.

10. Start the custom profiles, start the deployment manager, and then log in to the administrative console.

11. Start the deployment environment:

    a. In the administrative console, start the deployment environment by clicking **Servers** > **Server Types** > **Deployment Environments**. Select the check box next to the deployment environment and clicking **Start**.

    b. After 5 to 10 minutes (or longer, depending on the system), refresh the deployment environment page; the Status of the deployment environment changes to **started**.

12. Optional: Check the status of the following items:

    a. In the administrative console, select **Applications** > **Enterprise Applications** and check that the installed applications started successfully.

    b. Select **Resources** > **JDBC** > **Data sources** and test that the connection of every component that is not related to the message engine (that is, every component that does not include **ME** in the name) is successful.

Creating database design files by using the database design tool

Use the database design tool to create and generate a design of your database configuration. The design can be for a specific component or for an enterprise-level database configuration supporting the full functionality of IBM Business Process Manager.

Related concepts:

"Examples: Using a database design file with the manageprofiles command" on page 515
If you create a database design file using the database design tool, the generated file contains the values for the manageprofiles command.

Creating a database design file for a stand-alone profile or deployment environment by using the database design tool:

You can use the database design tool to generate a design file for database tables that can be used by profile creation or when using the deployment environment wizard. The database design tool generates the design file from user-interactive input or from an existing design file.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator (DBA) or solution architect. Alternatively, it might be a description of required parameters and properties.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

Before you run the database design tool, ensure that you have made the following decisions:
- The type of deployment environment in which the database will be used (stand-alone profile or network deployment environment) based on scalability and high-availability requirements.
- The location of database tables.
- Details about the database type, specifically, but not limited to, the following items:
  - Type of database (DB2, Oracle, DB2 for zOS, SQL Server)
  - Location of the JDBC driver on the system where the server profile will be created
  - User ID and password for authenticating to the database

Tip: Plan for database use when you review information about your planned usage of IBM Business Process Manager so that you make the necessary decisions on information needed by the database design tool.

This task describes how to use the database design tool to create a database design file for a stand-alone profile or deployment environment. The input for the database design tool is either user-interactive input or an existing design file. The available options change depending on your environment.

The **DbDesignGenerator** command has the following options.

```
-? , -help
   display help info.

-e db_design_file_name
   edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
   when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
   generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

1. Access the **DbDesignGenerator** command and run the file.

   You can find the **DbDesignGenerator** command in the following location:

   - Windows *install_root*\util\dbUtils

     For example, **C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat**

   - Linux   UNIX /*install_root*/util/dbUtils

     For example, **/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh**

Tip: If you see the message `The system cannot find the specified path`, you might have entered the path name incorrectly. Re-enter the path.

When the database design tool launches successfully, you see the following information:

```
[info] running DbDesignGenerator in interactive mode...

[info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
help at any time.
[info] To accept the given default values, simply press the 'Enter' key.

[info] Please pick one of the following [design option(s)] :

(1)Create a database design for Standalone profile or Deployment Environment
(2)Create a database design for a single component
(3)Edit an existing database design
(4)Generate database scripts from a database design
(5)exit [q]

Please enter the number for the design option :
```

2. To select the option **(1)Create a database design for Standalone profile or Deployment Environment**, type the number 1 and press Enter.

   You are prompted to choose a database pattern; for example:

   ```
   [info] Please pick one of the following [database pattern(s)] :

   (1)bpm.advanced.nd.topology
   (2)bpm.advanced.standalone
   (3)bpm.standard.standalone
   (4)wesb.nd.topology
   (5)wesb.standalone
   ```

3. To create a database design pattern for the stand-alone profile or deployment environment that you plan to configure, type the number for the appropriate option and press Enter. For a stand-alone profile, select options that include ".standalone;" for a deployment environment, select options that include ".nd."

   For example, to configure the database pattern for a deployment environment for IBM Business Process Manager Advanced, type the number 1 to select option **(1)bpm.advanced.nd.topology**, and press Enter. You see information similar to the following example:

   ```
   [info] Please edit any database component with status of 'not complete' for required properties.
   [info] Completed database components can be edited to change existing or defaulted property values.
   [info] Design the 'master' component first, and then any parent components, since other components may inherit value

   [info] Please pick one of the following [database component(s)] :

   (1)[CommonDB]    WBI_CommonDB : [master] [status = not complete]
   (2)[BPCReporting]      WBI_BPCEventCollector : [status = not complete]
   (3)[BPC]        WBI_BPC :  [status = not complete]
   (4)[BPM_PerformanceDW]  BPM_PerformanceDW :  [status = not complete]
   (5)[BPM_ProcessServer]  BPM_ProcessServer :  [status = not complete]
   (6)[BSpace]     WBI_BSPACE :  [status = not complete]
   (7)[SibME]      WBI_BPC_ME :  [status = not complete]
   (8)[SibME]      WBI_CEI_ME :  [status = not complete]
   (9)[SibME]      WBI_SCA_APP_ME :  [status = not complete]
   (10)[SibME]     WBI_SCA_SYS_ME :  [status = not complete]
   (11)[SibMe]     BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = not complete]
   (12)[SibMe]     BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = not complete]
   (13)[save and exit]
   ```

4. Type the number for the appropriate option to configure the master database component, and press Enter. You see the database components that can be configured for the previously selected environment. The database component listed as the master component lists **[master]** beside the name and must be configured first.

For example, to configure the master component for the **(1)bpm.advanced.nd.topology** design
pattern, type the number 1 to select option **(1)[CommonDB] WBI_CommonDB : [master] [status =
not complete]**, and press Enter. You see information similar to the following example:

```
[status] WBI_CommonDB is not complete with 1 remaining item(s):
[ 1 ] CommonDB.WBI_CommonDB :  : DbType key is not set.

Edit this database component? (y/n) [default=y] :
```

5. To edit the database component and select the database type that you are configuring, type y and
   press Enter.

   After you choose to edit the database component, you see information similar to the following
   example:

```
[info] Please pick one of the following [database type(s)] :

(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

6. Type the number that corresponds to the database type that you want to use for your environment,
   and press Enter. You obtain a set of prompts to specify the database properties. These prompts vary,
   depending on the database type that you plan to use.

   For example, type the number 1 to select **(1)DB2-distributed** as the database type. After you select
   this database type for configuration of the Common DB, you see information similar to the following
   example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=CMNDB] :
Database User name[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=] :

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

(1)false
(2)true

Please enter the number for the Is this database for a Process Center? [default=false] :1
The user ID you use for administrative security[default=] :
The password for the name specified with the adminUserName parameter[default=] :1
Regular pagesize[default=32k] :
Regular TableSpace[default=WBISPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=WBITEMPSPACE] :
```

7. At each prompt, if a default value is listed, enter the appropriate value for your database
   configuration, or press Enter to accept the default value. The administrative security prompt (if
   present) refers to the user name and password for the WebSphere administrative console.

   After you complete the last prompt for the database properties, you see information similar to the
   following example:

```
[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :
```

8. To configure the data source properties component, type anything other than **s** and press Enter. To
   skip this configuration and accept the defaults, type **s** and press Enter.

   Tip: If you plan to use the database design tool to generate a database design file for use as input
   for profile creation or topology configuration, you must configure the data source. If you plan to use
   the database design tool to generate SQL, this step is optional.
   If you chose to configure the data source for your selected database type, you see the list of database
   providers for the data source. For example, you might see the following database providers for the
   DB2-distributed database type:

```
[info] Please pick one of the following [database provider(s)] :

(1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA)
(2)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

a. Type the appropriate number to select a database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA)** as the database provider, type the number 1 and press Enter. After you select this database provider, you see information similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

b. At each prompt, enter the appropriate value for your database configuration, or if a default value is listed, press Enter to accept the default value. If you are asked whether you want to create the database tables during configuration, specify **No** if you are using the database design file as input for creating a profile, because this option is also available in the Profile Management Tool and causes a conflict. If you are using the administrative console to create your deployment environment, you can decide based on whether you want the tables to be created automatically.

After you complete the last prompt, you see information similar to the following example:

```
[status] WBI_CommonDB is complete with 0 remaining item(s):

-------------------------------------------------------------------------------

[info] Please edit any database component with status of 'not complete' for required properties.
[info] Completed database components can be edited to change existing or defaulted property values.
[info] Design the 'master' component first, and then any parent components, since other components may inherit value

[info] Please pick one of the following [database component(s)] :

(1)[CommonDB]    WBI_CommonDB : [master] [status = complete]
(2)[BPCReporting]       WBI_BPCEventCollector :  [status = complete]
(3)[BPC]          WBI_BPC :  [status = complete]
(4)[BPM_PerformanceDW]  BPM_PerformanceDW :  [status = complete]
(5)[BPM_ProcessServer]  BPM_ProcessServer :  [status = not complete]
(6)[BSpace]      WBI_BSPACE :  [status = complete]
(7)[SibME]       WBI_BPC_ME :  [status = complete]
(8)[SibME]       WBI_CEI_ME :  [status = complete]
(9)[SibME]       WBI_SCA_APP_ME :  [status = complete]
(10)[SibME]      WBI_SCA_SYS_ME :  [status = complete]
(11)[SibMe]      BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(12)[SibMe]      BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(13)[save and exit]
```

After you finish configuring the master database component, the database design tool propagates the values that you entered, to the remaining components. If this can be done successfully, these components are also marked as **[status = complete]** along with the master component. If this cannot be done for any reason, they remain listed as **[status = not complete]**.

9. Optional: Follow the preceding steps to configure the remaining database components that are listed as **[status = not complete]**. For any database components that are listed as a parent to another component, configure the parent before the other components because the information provided will be used as default settings for the database component listing the parent. You can also choose to reconfigure any components that are listed as **[status = complete]** as the result of configuring the master database component.

Note: If your database type is DB2 for z/OS, you must configure each of the remaining components. The following guidelines apply:

- Ensure that you specify the appropriate schema name for each of the remaining database components. You might also be prompted for other values that are unique to the type of component being configured; for example, a prefix for table space names or a value for VSAM catalog names (VCAT).
- If you intend to use a single database for all the system integration bus messaging engines, specify unique prefixes for their table space names because these prefixes will be used to keep the table space name of each messaging engine unique within the database.
- When configuring the Process Server component (**BPM_ProcessServer**), you are asked to specify whether the database is for a Process Center or Process Server:

  ```
  [info] Please pick one of the following [Is this database for a Process Center?(s)] :

  (1)false
  (2)true
  ```

  If you want to create a Process Center profile or cluster in a network deployment environment, you must choose the **true** option. If you set the option to **false**, you will not be able to access the Process Center console.

  If the database is for a Process Server, you must choose the **false** option.

- As you configure each component, verify that the values inherited from the master component are appropriate, and amend as required. For example, if you are planning to use more than one database, you might want to amend the database name to reflect the planned distribution of database objects across the DB2 for z/OS subsystem.

10. When all database components for your database pattern have been configured and are listed as **[status = complete]** in the database design tool, enter the appropriate number to select **[save and exit]**, and press Enter. For example, after you finish configuring the **(1)bpm.advanced.nd.topology** database pattern, type the number 13 and press Enter. You see information similar to the following example:

    ```
    [status] bpm.advanced.nd.topology is complete with 0 remaining item(s):

    Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
    ```

11. Enter the location where you want to save the database design file, and press Enter. After you enter the location at the prompt, you see information similar to the following example:

    ```
    Please enter the output filename [default=bpm.advanced.nd.topology.dbDesign] :
    ```

12. Enter the file name for the generated database design file, and press Enter. After you enter the file name at the prompt, you see information similar to the following example:

    ```
    generate database scripts? (y/n) [default=y] :
    ```

13. Optional: If you also want to generate database scripts based on the information provided to the database design tool, type y and press Enter.

    a. Specify the full path of the output directory that will contain the scripts for that database component, and press Enter.

       After you type y and press Enter to indicate that you want to generate database scripts, you see information similar to the following example for each database component:

       ```
       Please enter the output directory for WBI_CommonDB [default=DB2-distributed-CommonDB] :
       ```

       After you type the location for the output directory and press Enter, you see information similar to the following example after each entry:

       ```
       [info] The script(s) have been generated in C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-CommonDB
       ```

    After you enter the values for each prompt, you see information similar to the following example:

    ```
    [info] thanks, quitting now ...
    ```

A database design file and, optionally, database scripts are created at the locations that you specified.

You can choose to use the output from the database design tool in one of the following ways:

- If you generated only the database design file, you can specify the database design file and select the option to have it create the database tables as part of those configuration steps.
- If you generated both the database design file and SQL scripts, you can specify only the database design file to ensure that the configured run time matches the database tables created from the SQL scripts.

You can specify the database design file in several ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

The database design file can also be used when you configure a server as a Process Server, Process Center, or Performance Data Warehouse Server in the administrative console.

Related concepts:

"Examples: Using a database design file with the manageprofiles command" on page 515
If you create a database design file using the database design tool, the generated file contains the values for the manageprofiles command.

Creating a database design file for a specific component by using the database design tool:

You can use the database design tool to generate a design file for database tables required by specific components. The database design tool generates the design file from user-interactive input or from an existing design file.

Ensure that you have installed IBM Business Process Manager. The database design tool is available only from the installation binary files.

Before you run the database design tool, prepare the following information:

- Information about the database configuration that you are designing. This might be a document that describes the general purpose of the database configuration, supplied by the database administrator (DBA) or solution architect. Alternatively, it might be a description of required parameters and properties.
- Information about how IBM Business Process Manager and its components have been installed, the database software used, and the properties required by that type of database.
- An understanding of the profiles you plan to create, specifically, the functional relationship between the profile types and the databases.
- Information about the topology pattern to be implemented, and an understanding of how the database design fits into the pattern that you plan to use.

Before you run the database design tool, ensure that you have made following decisions:

- The type of deployment environment in which the database will be used (stand-alone profile or network deployment environment) based on scalability and high-availability requirements.
- The location of database tables.
- Details about the database type, specifically, but not limited to, the following items:
  - Type of database (DB2, Oracle, DB2 for zOS, SQL Server)
  - Location of the JDBC driver on the system where the server profile will be created
  - User ID and password for authenticating to the database

Tip: Plan for database use when you review information about your planned usage of IBM Business Process Manager so that you make the necessary decisions on information needed by the database design tool.

This task describes how to use the database design tool to create a database design file for a specific component. The input for the database design tool is either user-interactive input or an existing design file. The available options change depending on your environment.

The **DbDesignGenerator** command has the following options.

```
-? , -help
   display help info.

-e db_design_file_name
   edit the specified database design file (e.g. *.dbDesign, *.properties).

-v db_design_file | db_scripts_output_directory
   when a db_design_file is given, validation will be done on the specified
 database design file based on the database specs.
When a db_scripts_output_directory is given, the database scripts
in the specified directory will be validated. Currently only
scripts generated from template ddl generator can be validated.

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ...
[db_design_fileN] [-d output_directoryN]
   generate the database scripts from the specified design files in batch mode.
The generated scripts will be put in the corresponding output
directories or the default locations if output directories are absent.
```

Important: If you intend to generate database scripts for the service integration bus messaging engines, you must repeat step 1 to step 9 on page 578 of the following procedure, to define the database configuration and generate the scripts for each messaging engine that you require. For example, if you require scripts for six messaging engines, perform this procedure six times. For more information about the considerations for creating messaging engine data stores in a DB2 for z/OS subsystem, see Creating the messaging engine data stores.

1. Access the **DbDesignGenerator** command and run the file.

   You can find the **DbDesignGenerator** command in the following location:

   - Windows *install_root*\util\dbUtils

     For example, **C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat**

   - Linux UNIX /*install_root*/util/dbUtils

     For example, **/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh**

   Tip: If you see the message **The system cannot find the specified path.** you might have entered the path name incorrectly. Re-enter the path.
   When the database design tool launches successfully, you see information similar to the following example:

   ```
   [info] running DbDesignGenerator in interactive mode...

   [info] Enter 'q' to quit without saving; '-' for back to previous menu; '?' for
   help at any time.
   [info] To accept the given default values, simply press the 'Enter' key.

   [info] Please pick one of the following [design option(s)] :

   (1)Create a database design for Standalone profile or Deployment Environment
   (2)Create a database design for a single component
   (3)Edit an existing database design
   (4)Generate database scripts from a database design
   (5)exit [q]
   ```

2. To select the option **(2)Create a database design for a single component**, type the number 2 and press Enter.

   You are prompted for a component; for example:

```
[info] Please pick one of the following [component(s)] :


(3)bpm_performancedw
(4)bpm_processserver

(6)cei
(7)commondb
(8)sca
(9)sibme
```

3. To create a database design for the component that you plan to configure, type the number for the appropriate option and press Enter.

For example, to configure the IBM Process Server database component, type the number 4 to select option **(4)bpm_processserver**, and press Enter. You see information similar to the following example:

```
[info] Please pick one of the following [database type(s)] :

(1)DB2-distributed
(2)DB2-zOS
(3)Oracle
(4)SQL Server
```

4. Type the number that corresponds to the database type that you want to use for your environment, and press Enter. You obtain a set of prompts to specify the database properties. The prompts vary, depending on the database type that you plan to use.

For example, type the number 1 to select **(1)DB2-distributed** as the database type. After you select this database type for configuration of the database, you see information similar to the following example:

```
[info] Please enter the values for the properties in the database objects section.
Database name[default=BPMDB] :
Database User name[default=] :
System password(this is required ONLY for creating the database as a part of standalone profile creation.)[default=]

[info] Please pick one of the following [Is this database for a Process Center?(s)] :

(1)false
(2)true

Please enter the number for the Is this database for a Process Center? [default=false] :1
The user ID you use for administrative security[default=] :
The password for the name specified with the adminUserName parameter[default=] :1
Regular pagesize[default=32k] :
Regular TableSpace[default=TWPSSPACE] :
Temporary pagesize[default=32k] :
Temporary TableSpace[default=TWPSTEMPSPACE] :
```

5. At each prompt, enter the appropriate value for your database configuration, or if a default value is listed, press Enter to accept the default value. The administrative security prompt (if present) refers to the user name and password for the WebSphere administrative console.

After you complete the last prompt, you see information similar to the following example:

```
[info] You have completed database objects section properties needed for database scripts generation.

To skip data source properties, enter 's'; or enter anything else to continue :
```

6. To configure the data source properties component, type anything other than **s** and press Enter. To skip this configuration and accept the defaults, type **s** and press Enter.

Tip: If you plan to use the database design tool to generate a database design file for use as input for profile creation or topology configuration, you must configure the data source. If you plan to use the database design tool to generate SQL, this step is optional.

If you decided to configure the data source for a database after you selected **DB2-distributed** as your database type, you see information similar to the following example:

```
[info] Please pick one of the following [database provider(s)] :

(1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA)
(2)DB2 Using IBM JCC Driver # XA data source # DB2 Using IBM JCC Driver (XA)
```

a. Type the number for the appropriate option to select the database provider for the data source, and press Enter. For example, to select the option for **(1)DB2 Universal JDBC Driver Provider # XA data source # DB2 Universal JDBC Driver Provider (XA)** as the database provider, type the number 1 and press Enter. After you select this database provider for the data source, you see information similar to the following example:

```
[info] Please enter the values for the properties in the data source properties section.
Database server host[default=] :
Database server port[default=50000] :
Data source user name[default=] :
Data source password[default=] :
DB2 Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
Universal JDBC driver path[default=${WAS_INSTALL_ROOT}/jdbcdrivers/DB2] :
```

Note: The password is encrypted in the generated output files.

b. At each prompt, if a default value is listed, press Enter to accept the default, or enter the appropriate value for your configuration.

After you complete the last prompt, you see information similar to the following example:

```
Please enter the output directory [default=C:\IBM\WebSphere\AppServer\util\dbUtils] :
```

7. Enter the location where you want to save the database design file, and press Enter. After you enter the location, you see information similar to the following example:

```
Please enter the output filename [default=BPM_ProcessServer_DB2-distributed.properties] :
```

8. Enter the file name for the generated database design file and press Enter. After you enter the file name, you see information similar to the following example:

```
generate database scripts? (y/n) [default=y] :
```

9. Optional: If you also want to generate database scripts based on the information provided to the database design tool, perform the following steps:

a. Type y and press Enter.

After you type y and press Enter to indicate that you want to generate database scripts, you see information similar to the following example:

```
Please enter the output directory for BPM_ProcessServer [default=DB2-distributed-BPM_ProcessServer] :
```

b. Specify the full path of the output directory that will contain the scripts for that database component, and press Enter.

After you enter the location for the output directory, you see information similar to the following example:

```
[info] The script(s) have been generated in C:\IBM\WebSphere\AppServer\util\dbUtils\DB2-distributed-BPM_ProcessServe
```

After you enter the values for each prompt, you see information similar to the following example:

```
[info] thanks, quitting now ...
```

A database design file is created and, optionally, database scripts are created at the location that you specified.

After using the database design tool to configure a specific component, the generated SQL scripts can be used to create the database tables. The generated database design file includes only values for this configured component and is not sufficient for use in the following ways:

- when you use the profile management tool to create a profile
- when you use the **manageprofiles** command-line utility to create a profile
- when you use the Deployment Environment wizard to create your environment

Related concepts:

If you create a database design file using the database design tool, the generated file contains the values
for the manageprofiles command.

Database design tool examples:

You can use the database design tool to generate a design file for different types of databases, including
DB2, Oracle, and SQL Server.

DB2

If you select **bpm.standard.standalone** or **bpm.standard.nd**, after you configure the master component,
all database components are listed as complete.

```
[info] Please pick one of the following [database component(s)] :

(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW :  [status = complete]
(3)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(5)[save and exit]

Please enter the number for the database component :5

[status] bpm.standard.standalone is complete with 0 remaining item(s):
```

If you select **wesb.standard** or **wesb.nd.topology**, after you configure the master component, all database
components are listed as complete.

```
[info] Please pick one of the following [database component(s)] :

(1)[CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BSpace] WBI_BSPACE :  [status = complete]
(3)[SibMe] WBI_CEI_ME :  [status = complete]
(4)[SibMe] WBI_SCA_APP_ME :  [status = complete]
(5)[SibMe] WBI_SCA_SYS_ME :  [status = complete]
(6)[save and exit]

Please enter the number for the database component :6

[status] wesb.standalone is complete with 0 remaining item(s):
```

Oracle

If you select **bpm.advanced.standalone** or **bpm.advanced.nd.topology**, after you configure the
CommonDB, you must manually configure the **BPM_ProcessServer** and **BPM_PerformanceDW**
components for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :

(1)[CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPCReporting] WBI_BPCEventCollector :  [status = complete]
(3)[BPC] WBI_BPC :  [status = complete]
(4)[BPM_PerformanceDW] BPM_PerformanceDW :  [status = not complete]
(5)[BPM_ProcessServer] BPM_ProcessServer :  [status = not complete]
(6)[BSpace] WBI_BSPACE :  [status = complete]
(7)[SibME] WBI_BPC_ME :  [status = complete]
(8)[SibME] WBI_CEI_ME :  [status = complete]
(9)[SibME] WBI_SCA_APP_ME :  [status = complete]
(10)[SibME] WBI_SCA_SYS_ME :  [status = complete]
(11)[SibMe] BPM_PerformanceDW_ME :  [status = complete]
(12)[SibMe] BPM_ProcessServer_ME :  [status = complete]
(13)[save and exit]

Please enter the number for the database component :4
```

```
--------------------------------------------------------------------------------

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[ 1 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property 'databaseUser' for DB_USER is empty.

Edit this database component? (y/n) [default=y] :
[info] Please pick one of the following [database component(s)] :

(1)[CommonDB] WBI_CommonDB : [master] [status = complete]
(2)[BPCReporting] WBI_BPCEventCollector :  [status = complete]
(3)[BPC] WBI_BPC :  [status = complete]
(4)[BPM_PerformanceDW] BPM_PerformanceDW :  [status = complete]
(5)[BPM_ProcessServer] BPM_ProcessServer :  [status = not complete]
(6)[BSpace] WBI_BSPACE :  [status = complete]
(7)[SibME] WBI_BPC_ME :  [status = complete]
(8)[SibME] WBI_CEI_ME :  [status = complete]
(9)[SibME] WBI_SCA_APP_ME :  [status = complete]
(10)[SibME] WBI_SCA_SYS_ME :  [status = complete]
(11)[SibMe] BPM_PerformanceDW_ME :  [status = complete]
(12)[SibMe] BPM_ProcessServer_ME :  [status = complete]
(13)[save and exit]

Please enter the number for the database component :5
--------------------------------------------------------------------------------

[status] BPM_ProcessServer is not complete with 3 remaining item(s):
[ 1 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property 'databaseUser' for DB_USER is empty.
[ 2 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property 'adminUserName' for SECURITY_ADMIN_USER is
[ 3 ] BPM_ProcessServer.BPM_ProcessServer : databaseObjects : required property 'adminPassword' for SECURITY_ADMIN_PASSWORD

Edit this database component? (y/n) [default=y] :
```

SQL Server

If you select **bpm.standard.standalone** or **bpm.standard.nd**, after you configure the **BPM_ProcessServer**, you must manually configure the **BPM_PerformanceDW** component for the required database authentication.

```
[info] Please pick one of the following [database component(s)] :

(1)[BPM_ProcessServer] BPM_ProcessServer : [master] [status = complete]
(2)[BPM_PerformanceDW] BPM_PerformanceDW :  [status = not complete]
(3)[SibMe] BPM_PerformanceDW_ME : [parent = BPM_PerformanceDW] [status = complete]
(4)[SibMe] BPM_ProcessServer_ME : [parent = BPM_ProcessServer] [status = complete]
(5)[save and exit]

Please enter the number for the database component :2
--------------------------------------------------------------------------------

[status] BPM_PerformanceDW is not complete with 1 remaining item(s):
[ 1 ] BPM_PerformanceDW.BPM_PerformanceDW : databaseObjects : required property 'databaseUser' for DB_USER is empty.

Edit this database component? (y/n) [default=y] :
```

Related concepts:

"Examples: Using a database design file with the manageprofiles command" on page 515
If you create a database design file using the database design tool, the generated file contains the values for the manageprofiles command.

Troubleshooting the database design tool:

If you have errors in your database scripts, you can use the diagnostic and validation information provided by the database design tool to diagnose the problems.

Required property is empty errors

When the required **userName** and **password** properties are not set, you might see messages of the following type in the output:

```
[status] WBI_BSPACE is not complete with 2 remaining item(s):
[ 1 ] BSpace.WBI_BSPACE : authAlias : required property 'userName' for userId is empty.
[ 2 ] BSpace.WBI_BSPACE : authAlias : required property 'password' for DB_PASSWORD is empty.
```

Sample output of running a validation of the existing database design

When you run a validation of the existing database design, you might see warnings of the following type in the output:

```
DbDesignGenerator.bat -v DB2-distributed-
...
[WARNING] 2 potential problems are found in the scripts. They are
DB_USER @ line 46 in file configCommonDB.bat
DB_USER @ line 80 in file configCommonDB.sh
```

Contents of the database design tool log file

When you run the database design tool, a `dbDesignGenerator.log` file is created in the location from which the database design tool command is run. The log contains all the prompts and values entered. The log file does not contain any additional trace output.

Related concepts:

"Examples: Using a database design file with the manageprofiles command" on page 515
If you create a database design file using the database design tool, the generated file contains the values for the manageprofiles command.

Loading the database with system information

You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the database. This data is required for the BPM applications to run correctly

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation.

- If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.
- In a network deployment (ND) environment, you must run this command after a server or cluster of servers is created. **For a cluster, you need to run the command only once**, specifying a single cluster member. Run this command after the database and its tables have been created, after the profile and deployment environment have been created, and before the first server is started. You do not need to rerun the command if you add another cluster member.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone or deployment manager profile directory. For example:

```
<install_root>/profiles/Dmgr01/bin
C:\<install_root>\profiles\ProcCtr01\bin
```

Run the bootstrap utility using the following syntax:

- `Linux` `UNIX` **bootstrapProcessServerData.sh [-dbJDBCClasspath** *classpath*] **-clusterName** *cluster_name*

- Linux UNIX **bootstrapProcessServerData.sh [-dbJDBCClasspath** *classpath*] **-nodeName** *node_name* **-serverName** *server_name*

- Windows **bootstrapProcessServerData.bat [-dbJDBCClasspath** *classpath*] **-clusterName** *cluster_name*

- Windows **bootstrapProcessServerData.bat [-dbJDBCClasspath** *classpath*] **-nodeName** *node_name* **-serverName** *server_name*

where:

- **-dbJDBCClasspath** is the location of the database JDBC driver. The value specified for the **dbJBBCClasspath** must be surrounded by quotation marks if there are spaces in the path. The default location is *WAS_INSTALL_ROOT*/jdbcdrivers.

- **-clusterName** is the name of the application target cluster. For an ND environment, you must specify either this parameter or the **-nodeName** and **-serverName** parameters. Do not specify for stand-alone.

- **-nodeName** is the name of the node. For an ND environment, you must specify either this parameter and the **-serverName** parameter, or the **-clusterName** parameter. Do not specify for stand-alone.

- **-serverName** is the name of the server. For an ND environment, you must specify either this parameter and the **-nodeName** parameter, or the **-clusterName** parameter. Do not specify for stand-alone.

The parameters are case-sensitive.

You have loaded the database with system information prior to successfully starting the Process Server or Process Center.

Important: For SQL Server, if the bootstrap utility fails, check that when you created the Process Server and Performance Data Warehouse databases, the **COLLATE** attribute indicated case insensitive (**CI**), not case sensitive (**CS**). That is, the create command should look like this:

```
CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS
```

Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.

Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

Bootstrap data onto a server that is part of an ND environment but not part of a cluster:

```
bootstrapProcessServerData -nodeName node01 -serverName server1
```

Bootstrap data onto a cluster that hosts the Process Center or Process Server:

```
bootstrapProcessServerData -clusterName myAppCluster
```

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData
```

Bootstrap data onto the default stand-alone profile server with a custom JDBC driver location:

```
bootstrapProcessServerData -dbJDBCClasspath c:/DB2/java
```

You can start the stand-alone server or first cluster member.

Creating and configuring the DB2 for z/OS database

If your deployment environment relies on a remote DB2 for z/OS database, use the procedures and reference information in this section to help you configure the database and create the database tables.

Related tasks:

"Configuring an existing database during a typical installation" on page 107
Use the information in this topic to determine the correct database values for configuring your existing database server during a typical installation.

Create the DB2 for z/OS databases and storage groups using SPUFI, DSNTEP2, or DButility.sh:

The profile creation process generates Data Definition Language (DDL) scripts that you can use to create the DB2 database objects for the configuration. There are several tools that you can use to run the DDL scripts to create the database objects for your configuration. You can also use tools such as SPUFI or DSNTEP2 to create and populate the database.

Before you create the DB2 databases and storage groups, you must complete the following tasks:

- Create the server configuration. See *Roadmap: Installing and configuring IBM BPM Standard Roadmap: Installing and configuring IBM BPM Express* for information about how to create a configuration for a stand-alone server and network deployment environment.
- Make sure that the DDL has been generated for all the components you want to configure the database with. You can generate the DDL by completing the following tasks:
  - Designing the database configuration

    For a network-deployment environment, using the database design tool (DDT) is recommended.

    For a stand-alone server environment, the database panels of the Profile Management Tool are usually enough to for stand-alone profiles, although you can use the DDT.

    The output of the DDT is a design document (xml file) of the database configuration and, optionally, the SQL scripts to create the database tables.
  - Prepare to use the DDL files
    - You might need to copy the DDL files from the IBM Business Process Manager file system into a partitioned dataset (PDS). You can use a tool such as **Dd12Pds.sh** to copy the files.
    - There is no restriction on the naming or organization conventions that apply to the database objects.
    - The CEI DDL and the SIB DDL files need to be customized before they can be run.

      Note: You can use the sample SIB DDL provided for single database configuration.

You can run the DDL scripts using **DBUtility.sh**, SPUFI, or DSNTEP2. You can choose one tool over another based on experience and familiarity, or personal preference. Your organization might also have implemented standards or conventions for the tools used to create DB2 objects, particularly in a production environment. The tools can produce an audit trail of the DB2 database commands that have been issued.

If you want to create the database objects across multiple databases but still want to work in the z/OS UNIX System Services environment, you can run the DDL scripts using the **DBUtility.sh** script several times specifying different components for each database name.

Important: After converting from ASCII to EBCDIC, check that no SQL statements exceed 71 characters in length. Longer lines will lead to line truncation and invalid statements when copying to fixed width MVS data sets.

1. Create the databases and storage groups.
2. Populate the databases using the generated DDL scripts. The location of the generated DDL scripts depends on how they were generated.

   You can find the **DbDesignGenerator** command in the following location:

   - ▪ `Windows` *install_root*\util\dbUtils

     For example, `C:\Program Files\IBM\WebSphere\AppServer\util\dbUtils> DbDesignGenerator.bat`

   - ▪ `Linux` `UNIX` */install_root*/util/dbUtils

     For example, `/opt/IBM/WebSphere/AppServer/util/dbUtils> DbDesignGenerator.sh`

   For DDL generated by other means, the DDL is in the directories under the following locations:

   - ▪ WAS_HOME/profiles/default/dbScripts for a stand-alone configuration.
   - ▪ WAS_HOME/profiles/default/dbScripts for a network deployment configuration.

   Where WAS_HOME is the top directory of your WebSphere Application Server configuration.

3. If you are running the DDL from a z/OS UNIX environment, assign the appropriate permissions to the copies of the files; for example:

   `chmod 755 createTable_AppScheduler.sql`

4. Edit the values in the file to suit your needs. The database names, storage groups and schema names are customized by the product configuration process. Check the values in each file to make sure they match the values that you entered in the response file that provided input to the configuration script and are suitable for your database.

   Note: The files can be provided in ASCII format. If the tools that you use to view, edit, and run the scripts require the scripts to be in EBCDIC format, use the **iconv** command to convert the file to EBCDIC. For example:

   ```
   iconv —t IBM-1047 —f ISO8859-1 createTable_AppScheduler.sql >
   createTable_AppScheduler_EBCDIC.sql
   ```

   If you have converted the file from ASCII format to EBCDIC but need to run the file in ASCII format, use **iconv** to convert the file back to ASCII. For example:

   ```
   iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >
   createTable_AppScheduler.sql
   ```

5. Run the customized scripts using the tool of your choice. For example:

   SPUFI A utility that runs SQL scripts from z/OS. SPUFI uses EBCDIC input.

   DSNTEP2
   > A sample dynamic SQL program provided with the DB2 for z/OS product.

   DBUtility.sh
   > **DBUtility.sh** is a utility that is supplied with IBM Business Process Manager for z/OS and installed in the installation file system. For example: /bin/DBUtility.sh. You can use this utility to create the database and storage groups, as well as to run the SQL to create the database tables later, from z/OS UNIX. **DBUtility.sh** uses ASCII input. Here is an example of the syntax used with the **DBUtility.sh** script:

```
/WebSphere/V7S03Z1/AppServer/profiles/default/bin/DBUtility.sh
createTable
-DdbStorageGroup=S3DBSTO
-DdbSchemaName=S3CELL
-DsqlScriptName.default=createTable_AppScheduler.sql
-DsqlScriptPath.default=/WebSphere/V7S03Z1/AppServer/profiles/default/dbscripts/CommonDB/DB2zOS/S3CELLDB
/createTable_AppScheduler.sql
-DdbType=DB2UDBOS390
-DdbName=S3CELLDB
-DprofileName=default
```

```
-DprofilePath=/WebSphere/V7S03Z1/AppServer/profiles/default
-DdbJDBCProperties=/wps/dbscripts/db2v9
-DdbConnectionLocation=DSN810PP
-DdbJDBCClasspath=/usr/lpp/db2910/db2910/jcc/classes
-DdbUserId=wsadmin
-DdbPassword=password
-DdbDelayConfig=false
-DdbCreateNew=false
-DdbHostName=winmvsp1.hursley.ibm.com
-DdbServerPort=448
>/tmp/output.out 2>>/tmp/error.out
```

6. Verify that the database, storage group, and tables have been created successfully with no errors by inspecting the output.

7. If you are creating a stand-alone configuration, verify the IBM Business Process Manager installation:

   a. Start the server.

   b. Open the administrative console by opening a browser window and typing the URL of the server that you want to view. For example:

      `http://`*server_name.domain_name:port_number*`/admin`

   c. Log in to the administrative console.

   d. Verify that you can see IBM Business Process Manager for z/OS on the Welcome page. You can click it for more information.

   e. Navigate around the console to check that the server has a status of started. Also check that all the applications are started, and that the messaging engines are started. If anything has failed to start, you can look in the server job logs for "SEVERE" or "WARNING" messages that provide details about the failure.

The DB2 databases and storage groups are created and populated with the necessary database objects, such as tables and indexes.

If you are creating a stand-alone configuration, you can now deploy applications to the server.

If you are creating a network deployment configuration, you must create one or more empty nodes to add to the deployment manager cell. See Configuring the software after a Custom installation to create one Deployment manager and Custom profiles.

Granting table privileges to the JCA authentication alias user ID:

If the schema name you are using is not the same as the JCA authentication alias user ID you must grant a sub-set of DB2 privileges to the JCA authentication alias user ID.

The DDL for the Service Integration Bus already contains commented GRANT commands that you can use as a basis for granting access to the SIB tables. However, the other IBM Business Process Manager components do not supply any GRANT statements.

Use a schema name that is not the same as the JCA authentication alias to prevent the alias user ID having the power to drop tables. (The power to drop tables is implicitly granted to the creator, that is, the schema.) Note that it does not make sense to grant a privilege like DBADM to the JCA authentication alias user ID because DBADM also has the ability to DROP tables.

If you want the IBM Business Process Manager to function while not allowing the alias user ID to have DROP capability, create some GRANT statements by copying the DDL and editing it to construct GRANT commands from the CREATE commands. Create GRANT commands like:

```
GRANT ALL PRIVILEGES ON TABLE
cell.tablename TO userid/sqlid
```

Where *userid/sqlid* is the JCA authentication alias user ID.

Setting the correct schema name for the SIBs:

To ensure the SIB messaging engines can access the appropriate DB2 tables, set the correct schema name for the SIB messaging tables to use to access the DB2 tables.

Start the server (stand-alone server or deployment manager).

Use the administrative console to change the schema names.

1. Log in to the administrative console.
2. Navigate to **Service Integration** > **Buses**.
3. For each bus:
   a. Select **Messaging engines**, then click the name that is displayed.
   b. Click **Message store**.
   c. Change the value of **Schema name** to the name used when creating the DB2 tables for this SIB.
   d. Click **Apply**.
   e. Save your configuration changes.
4. Log out of the administrative console.
5. Stop, then restart the server.
6. Look in the output of the Adjunct job log for successful SIB messaging engine startup messages. For example:

```
BBO00222I: "BusName"
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

The schema name used by the SIB messaging tables to access the DB2 tables is changed.

Verifying the installation with DB2 for z/OS:

When verifying an installation with a DB2 for z/OS database, it is important to check the Servant and Adjunct job logs to see whether there are any error messages that might indicate problems accessing the data store.

1. Ask your DB2 system administrator to check the authorities that have been granted to ensure that you have not granted more authority than necessary to any user ID. It can be tempting to grant DB2 SYSADM authority to the JCA authentication aliases in order to avoid possible problems with DB2 security during the configuration.
2. Ask your DB2 system administrator to check the storage group assignments and buffer pool usage. Incorrect storage group assignment and buffer pool usage might not show up as an error message in a log but might cause problems later. It is better to resolve such problems now rather than when the system has been handed over to people to use. For example, correcting storage groups and VCATs is not easy after the tables and indexes have been used.
3. Log in to the administrative console.
4. In the administrative console, check that all the applications are started, the messaging engines are started, and all the data sources can be accessed using the **Test Connection** option. If any application has failed to start, look in the Servant and Adjunct job logs for SEVERE or WARNING messages that provide detail about the failure.
   - If you see DB2 errors such as SQLCODE -204, in the administrative console, set the correct schema name or currentSQLID value in the custom properties section of the data sources. If the schema name is not the same as the user ID in the JCA authentication aliases, the SQL requests try to find tables qualified by the user ID in the JCA authentication alias.
   - If you see DB2 deadlock errors such as SQLCODE -913 Reason Code 00C90088, set the RRULOCK DB2 parameter to YES to prevent tablespace locks on IBM Business Process Manager tables.

If all the messaging engines have initialized correctly, and you do not see any other errors related to opening JDBC connections, you can continue to customize your configuration of IBM Business Process Manager.

Modifying the database password after configuration

You can reconfigure the database password, as needed, once your database configuration is completed.

You reconfigure the database password using the administrative console. There are three processes you need to complete to change the database password.

- Update the authentication alias for all data sources
- Update the Messaging Engine data store authentication alias
- Manually update the Process Server and Performance Data Warehouse password in the 98database.xml file

Important: Before you make any configuration changes, it is suggested that you backup the config folder. The config folder is located beneath the Deployment Manager at *<Dmgr-profile>*/config. Contact your WebSphere Application Server administrator for more information.

Updating the data source authentication alias:

You need to update the data source authentication alias before you change a database password.

Make sure you do all the following items before you begin this procedure.

- For a Network deployment environment, make sure all sending of requests is stopped. You must also stop all the clusters.
- For a Standalone profile, make sure all sending of requests is stopped.
- You must start the DMGR and all node agents after stopping the network deployment clusters and servers before changing the alias.
- All locations for the 98Database.xml file are listed in "Updating the password in the configuration file" on page 588.

To modify the data source authentication alias, complete the following steps:

1. Login to the administrative console.
2. Select **Resources** > **JDBC** > **Data sources**.

   Note: Make sure you write down the values that are configured for **Component-managed authentication alias** and **Authentication alias for XA recovery**. To identify the datasource for the component, see the *ConsolidateJAASAuthAliases.py script* topic in the related information section.
3. Select the desired datasource.
4. Select the Related Items section and then select **JAAS - J2C authentication data**.
5. Select the appropriate **Component-managed authentication alias** value noted in step 2. Edit the password, and then click **OK** to save the change.
6. Go back to the **JAAS - J2C authentication data** panel and select the **Authentication alias for XA recovery** value noted in step 2. Edit the password, and then click **OK** to save the change.
7. Save the updates and click **Full Synchronize** to synchronize the information to all the nodes.

Updating the Messaging Engine data store authentication alias:

You need to update the Messaging Engine data store authentication alias before you modify the database password.

Make sure you do all the following items before you begin this procedure.

- For a Network deployment environment, make sure all sending of requests is stopped. You must also stop all the clusters.
- For a Standalone profile, make sure all sending of requests is stopped.
- You must start the DMGR and all node agents after stopping the network deployment clusters and servers before changing the alias.
- All locations for the 98Database.xml file are listed in "Updating the password in the configuration file."

To update the Messaging Engine data store authentication alias, complete the following steps:

1. Login to the administrative console.
2. Select **Service Integration** > **Buses**.
3. To update the authentication alias for each Bus, do the following steps:
    a. Select **Buses** > **[Bus_NAME]** > **Messaging engines** > **[MESSAGING_ENGINE]** > **Data Store** .

       Remember: Make sure you write down and save the **Authentication Alias** value in the Data Store.
    b. Select the Related Items section and click **JAAS - J2C authentication data**.
    c. Select the value noted in step a. Edit the password, and then click **OK** to save the change.
    d. Save the updates and click **Full Synchronize** to synchronize the information to all the nodes.

Updating the password in the configuration file:

You need to update some configuration files in order to modify your database password.

IBM Process Center, IBM Process Server, and IBM Performance Data Warehouse store their database information in the 98Database.xml file that you need to update with the new password. To locate where the password is to be updated, open the 98Database.xml file in an editor and then search for an entry that looks similar to the following:

<encrypted-password>7Yu5Kz3XaYL5aUjbWfISHg==:VyP9e8ytnTz6623hERR4lg==</encrypted-password>

The above entry needs to be updated with the password value generated by using the encryption tool. Use the method described in the *Encrypting passwords* topic in the related links section.

Attention: Make sure the DMGR and the node agents are operating before you begin your password update. Once you have updated the 98Database.xml file with the new password, you must click **Full Synchronize** to percolate your changes to all nodes. And then, you must restart your environment to make sure the new password is activated in your network deployment environment or your stand-alone environment.

Table 1 identifies the locations of the 98Database.xml configuration file in a stand-alone environment.

Table 102. 98Database.xml Configuration file locations for a stand-alone environment

Deployment type	File location
Process Center	Process Center **<STANDALONE-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\ <server name>\process-center\config\system** Performance Data Warehouse **<STANDALONE-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\ <server name>\performance-data-warehouse\config\system**

| Process Server | Process Server

<STANDALONE-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\ <server name>\process-server\config\system

Performance Data Warehouse

<STANDALONE-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\ <server name>\performance-data-warehouse\config\system |

Table 2 identifies the locations of the 98Database.xml configuration file in a network deployment environment.

Table 103. 98Database.xml Configuration file locations for a network deployment environment

Deployment type	File location
Process Center	**1 or 2 cluster topologies** Process Center **<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\process-center\config\system** For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\process-center\config\system** Performance Data Warehouse **<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\performance-data-warehouse\config\system** For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\performance-data-warehouse\config\system**
	3 or 4 cluster topologies Process Center **<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\process-center\config\system** For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\process-center\config\system** Performance Data Warehouse **<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<SUPPORT-CLUSTER-NAME>\performance-data-warehouse\config\system** For each node used as part of the Support Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\performance-data-warehouse\config\system**

Process Server	**1 or 2 cluster topologies**
	Process Server
	\<DMGR-PROFILE-DIR\>\config\cells\\\<cell name\>\clusters\\\<APPLICATION-CLUSTER-NAME\>\process-server\config\system
	For each node used as part of the Application Cluster - **\<DMGR-PROFILE-DIR\>\config\cells\ \<cell name\>\nodes\\\<node name\>\servers\\\<server name\>\process-server\config\system**
	Performance Data Warehouse
	\<DMGR-PROFILE-DIR\>\config\cells\\\<cell name\>\clusters\\\<APPLICATION-CLUSTER-NAME\>\performance-data-warehouse\config\system
	For each node used as part of the Application Cluster - **\<DMGR-PROFILE-DIR\>\config\cells\ \<cell name\>\nodes\\\<node name\>\servers\\\<server name\>\performance-data-warehouse\config\ system**
	3 or 4 cluster topologies
	Process Server
	\<DMGR-PROFILE-DIR\>\config\cells\\\<cell name\>\clusters\\\<APPLICATION-CLUSTER-NAME\>\process-server\config\system
	For each node used as part of the Application Cluster - **\<DMGR-PROFILE-DIR\>\config\cells\ \<cell name\>\nodes\\\<node name\>\servers\\\<server name\>\process-server\config\system**
	Performance Data Warehouse
	\<DMGR-PROFILE-DIR\>\config\cells\\\<cell name\>\clusters\\\<SUPPORT-CLUSTER-NAME\>\ performance-data-warehouse\config\system
	For each node used as part of the Support Cluster - **\<DMGR-PROFILE-DIR\>\config\cells\\\<cell name\>\nodes\\\<node name\>\servers\\\<server name\>\performance-data-warehouse\config\ system**

Modifying the transaction log options for a DB2 database

When you configure DB2 for use with IBM Business Process Manager, you must modify the transaction log options.

1. Start a DB2 command line processor.
2. Run the following commands:

```
CONNECT TO [DB_name]
UPDATE DB CFG FOR DB_name USING LOGFILSIZ 4096 IMMEDIATE
UPDATE DB CFG FOR DB_name USING LOGSECOND 64 IMMEDIATE
CONNECT RESET
```

3. Stop and restart DB2.

Configuring a network deployment environment

After you have finished installation and created the deployment manager and custom profiles, you can build a network deployment configuration.

Creating a network deployment environment

After performing a Custom installation, you can use the Profile Management Tool or the **manageprofiles** command-line utility to build a network deployment cell across multiple machines. For each cell, you create a deployment manager profile and one or more custom (managed node) profiles. You can then create the deployment configuration that you want.

Alternatively, to configure a simple network deployment topology, you can use the **configureNode** command to create the required profiles and clusters instead of using the Profile Management Tool and Deployment Environment wizard or the manageprofiles and wsadmin commands.

The information in this section describes how to configure a network deployment environment; in summary:

1. Create a deployment manager profile and one or more custom (managed node) profiles, by using the Profile Management Tool
2. Create a deployment configuration, by using either the Deployment Environment wizard or the administrative console.

The information in this section assumes that you have run the installer to perform a Custom installation.

Tip: Instead of using the Profile Management Tool, you can use the manageprofiles command-line utility to create profiles, as described in Configuring the software using command-line utilities and wsadmin.

Configuring a network deployment environment using the deployment environment wizard:

After performing a Custom installation and creating the deployment manager and custom (managed node) profiles, you can create a network deployment configuration.

The information in this section describes how to use the deployment environment wizard to create a network deployment environment based on the topology pattern templates packaged with the software.

The information in this section assumes that you have run the installer and have performed a Custom installation and that you have created the deployment manager and custom (managed node) profiles.

For information about using wsadmin to create a network deployment configuration, see *Creating deployment environments using the command line*.

Related concepts:

"Overview: Deployment environment topologies and patterns" on page 33
A network deployment environment can have many topologies, and can be created from several standard topology patterns.

"Topology patterns and supported product features" on page 48
A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

⇨ Virtual hosts

Related tasks:

"Creating deployment environments using the command line" on page 610
You can use wsadmin to create a Process Server and Process Center deployment environment. The **createDeploymentEnvDef** and **generateDeploymentEnv** provide a command-line equivalent to creating the deployment environment using the deployment environment wizard.

⇨ Configuring virtual hosts

Creating the Process Server deployment environment using a pattern:

After determining the pattern on which to base your network deployment configuration, use the Deployment Environment wizard to create the IBM BPM Standard Process Server deployment environment that is based on the pattern.

Tip: An alternative to using a pattern to create the deployment environment is using an imported design. For more information, see Importing deployment environment definitions based on design documents using the administrative console

Required security role for this task: When security and role-based authorization are enabled, you must log in to the administrative console as an administrator or configurator to perform this task.

You should have planned the patterns and features that you want to configure. For more information, see Planning a network deployment environment.

It is assumed that you have installed the product and that you have created the deployment manager profile and the associated nodes.

Additionally, one of the steps in the Deployment Environment Configuration wizard includes importing a database design document. The database design document defines the database configuration for the selected deployment environment features. IBM Business Process Manager includes a response-driven database design tool (DDT) that creates a database design document based on user inputs. In addition to creating the design document, the DDT provides an option to create the database tables as well. Typically, the person running the DDT would choose to create the database tables at that point. The document can then be used by the IBM Business Process Manager deployment environment wizard to configure the databases used in the deployment environment. For more information about the DDT and database configuration in general, see Configuring databases.

This task describes the procedure for creating a deployment environment that is based on a specific pattern and uses the Deployment Environment Configuration wizard.

Note: If you make an error while you are working in the wizard, you can go back by clicking **Back**.

1. From the administrative console, go to the Deployment Environments page. For example, click **Servers** > **Deployment Environments**
2. Launch the Deployment Environment Configuration wizard by clicking **New** on the Deployment Environments page.
   a. The **Create a deployment environment based on a pattern** option is selected. **Create a deployment environment based on a pattern** is the system default and it is the option described in this topic.

      Deployment environment patterns capture commonly used business integration topologies. A pattern provides you with a template for the deployment environment that you are creating.

      Note: Patterns have a direct relationship to the products supported by the configured deployment manager. IBM Business Process Manager supports a specific set of patterns, with the *Remote messaging and remote support* pattern being the system default. If your deployment manager supports other products in addition to IBM Business Process Manager, additional patterns may apply. Consult product-specific documentation for information on patterns as they apply to the products.

      For information on the types of patterns provided with IBM Business Process Manager, see Topology types and deployment environment patterns.
   b. Enter a unique name for the deployment environment in the **Deployment environment name** field.
   c. Optional: To view all of the configuration steps in the wizard, select **Detailed: Show all steps**.

      If you choose **Fast path: Show only needed steps** the wizard displays only those pages that **do not** have assigned default values. Choose **Fast path: Show only needed steps** only if you are agreeable to accepting the system-provided default values for the deployment environment configuration.

      This topic assumes that you have chosen **Detailed: Show all steps**
   d. Click **Next** to display the Deployment Environment Features page.

3. On the Deployment Environment Features page, select the feature for the deployment environment and click **Next** to either view a list of compatible features, or to view a list of deployment environment patterns. Features represent the runtime processing capabilities of your deployment environment.

   The list of available features on the Deployment Environment Features page is based on the deployment manager profile. If your deployment manager profile has been augmented to include other products alongside IBM Business Process Manager (for example, IBM Business Monitor , then the Deployment Environment Features page also lists these features.

   If you have installed and configured a profile for IBM Business Process Manager, then the Deployment Environment Features page includes the following details:

   - For IBM Business Process Manager Standard:
     - **BPMSPS**, for an IBM BPM Standard, Process Server deployment manager profile, which provides a Process Server deployment environment that supports business processes, human tasks, and business rules.
     - **BPMSPC**, for an IBM BPM Standard, Process Center deployment manager profile, which provides a Process Center deployment environment that supports business processes, human tasks, and business rules.

   The default value for the deployment environment feature matches the runtime capabilities of your deployment manager.
4. On the Select compatible deployment environment features page, select additional features as necessary and click **Next** to view the list of patterns associated with your primary and ancillary feature selections.

   Note: The Select compatible deployment environment features page is displayed only if the deployment manager has been augmented with other business process management (BPM) features, such as IBM Business Monitor.

   For an understanding of the relationship of features and compatible features, see the information on deployment environments in the Planning section.
5. On the Select the deployment environment pattern page, select the pattern for the selected deployment environment, then click **Next** to display the Select Nodes page.

   The list of patterns that display on the Deployment Environment Patterns page is dynamic. This list is activated by, and dependent on, the following environment conditions and configuration decisions:

   - The platform on which you have installed the software
   - The selections that you have made on the Select the deployment environment feature page and the Select compatible deployment environment features page.

   For a detailed description of the relationship of patterns to features, see Topology patterns and supported product features
6. Optional: On the Select Nodes page, select the nodes to be included in the deployment environment then click **Next** to display the Clusters page.

   Select nodes that have the required capabilities for the environment you selected in step 3. For example, if you selected **BPMSPC** as your Deployment Environment type, the nodes selected should address the capabilities of that environment type.

   Select at least one node for the deployment environment. For high-availability and failover environments, select at least two nodes. For scalability, select all nodes.

   To include a node, select the check box next to the node name. Use **Node Mapping** to map the selected node to another node name.
7. Optional: On the Clusters page, assign the required number of cluster members on each node for each cluster *type* (Application Deployment Target, Messaging Infrastructure and Supporting Infrastructure) of the deployment environment.

By default one cluster member is assigned on each node for each function. You change the number by replacing the number in each column. If you are unfamiliar with the different cluster roles and functions provided by each type of cluster, see "Topology types and deployment environment patterns."

A 0 (zero) value for a node means that the node does not contribute to the selected function, based on features that you have selected.

After assigning cluster members, you can click **Next** to display the Cluster naming pages for each cluster type of the deployment environment. The Cluster naming sub-steps that display will vary depending on the deployment environment pattern selected.

The system generates default values for cluster names and cluster member names.

If you do not want to customize cluster names or cluster member names, you can use the wizard navigation pane to go directly to the REST Services page in a following step.

Each substep page is structured in the same fashion, and is described in Customize the cluster names and cluster member names.

a. Optional: Customize the cluster names and cluster member names.

Use the Cluster Naming page to customize cluster names or cluster member names for the cluster type. There is one substep page for each cluster *type* in the pattern that you have selected. For example, if you selected a **Remote messaging and remote support pattern**, there are 3 sub-steps, one for each type of cluster (Application Deployment Target, Messaging Infrastructure and Supporting Infrastructure) in that pattern.

The information on each substep page is as follows:

Cluster
A read-only field specifying the functional role of the cluster.

The value varies depending on the cluster type, as follows:
- Application Deployment Target
- Messaging Infrastructure
- Supporting Infrastructure
- Web Application Infrastructure

For information on the functional role provided by each cluster type, see Topology types and deployment environment patterns

Cluster Name
Contains the system-generated default value for the cluster name.

The default values are based on a naming convention of **<Deployment Environment Name>.<Cluster type name>**, where**cluster type name** is one of the following values:
- AppTarget

  For clusters performing the role of application deployment target
- Messaging

  For clusters performing the role of messaging infrastructure
- Support

  For clusters performing the role of supporting infrastructure
- Web

  For clusters performing the role of supporting web applications.

  Note: This cluster type name applies for BPM configurations in which WebSphere Business Monitor is the primary feature / product.

Cluster Member Name
Accept the system-generated default value or specify a name of your choosing.

The default value for the cluster member name is based on the following naming convention: **<cluster name>.<node name>.<node number sequence>** .

The number of cluster member names that display in the table match the number of cluster members that you entered for the cluster type column and node row on the Clusters page. See the preceding step for the Clusters page.

8. Optional: On the Import the database configuration page, click **Browse** to go the database design document or enter the path to the database design document and then click **Next** to go to the Data sources page. The design document can be based on a database design that you created using the database design tool (DDT), or it can be the supplied design document based on the pattern and feature that you have selected.

   Note: The database design document that you import for the deployment environment does not change the commonDB created at Profile Creation time.

9. Required: On the Database page, configure the database parameters for data sources of the deployment environment, then click **Next** to go to the Security page.

   Note: The database specified in this panel must already exist. Deployment environment configuration never creates a new database.

   For DB2 and SQL Server databases, IBM Process Server and IBM Performance Data Warehouse should not use the same database as the rest of the components. However, if you are using an Oracle database, IBM Process Server and IBM Performance Data Warehouse can use the same database instance, but should use different users.

   On this page, define the database information for the components that are included in this deployment environment. Where possible, the wizard supplies default information for the parameters, but change those values to match the values that you defined when you planned the environment.

   Note: If you imported a database design document, the information on the Database page reflects the data source configuration as it exists in the database design document that you imported.

   Whether or not this step displays for a fast path deployment environment configuration is conditional. This step displays for a fast path deployment environment configuration if more than one database has been defined.

   This step always displays if you are using DB2 for z/OS or an Oracle database provider.

   Note: The default schema names that are displayed on this page might conflict with your site naming convention or might conflict with existing schemas. As such, it is likely that you will need to change the schema name. Pay close attention to the values specified to avoid potential naming conflicts.

   Oracle database considerations:
   - Make sure that the username and schema name are exactly the same. The user specified should exist in the database before generating the environment.

   SQL Server considerations:
   - Make sure that the username and schema exist before the configuration is done. The schema value should be the default schema for the user chosen.
   - To indicate that users will connect to the databases using Windows credentials, select the individual data source, click **Edit**, and select **Apply Windows authentication**.

   For a production environment, you should set the same values for **User name** and **Schema name** and you should deselect **Create tables**. For a production environment, create the required schemas manually and use the SQL files generated to create the tables.

Note: You cannot select **Create tables** for Business Space (the option is unavailable for selection). The SQL files for Business Space need to be run manually. For information on running the SQL manually for Business Space, see Configuring Business Space database tables.

You can edit all key parameters, such as the database name, whether or not to create tables, the data source runtime user name, and the password for the deployment environment.

You can select which database to use for the given component.

DB2 for z/OS: The **Create tables** option cannot be used if you are using a DB2 for z/OS database provider.

Steps that cannot be completed through the Deployment Environment Configuration wizard, and which need to be completed manually, are listed on the Deferred Configuration page.

10. On the Process Server page, set the values for the Process Server configuration as follows:

- **Environment name**

  Enter an environment name of the Process Server.

  An environment name is the name by which this server or cluster will be known to a Process Center user.

- **Environment type**

  From the pull-down list, select the Environment type for the Process Server you are configuring.

  The Environment type refers to how the Process Server is used. For example, in what capacity will the Process Server be used - *production*, *stage* or *test* . Load testing might be done on a test server, while a stage environment type might be used as a temporary location to host changes before putting those changes into production. You might specify **Stage** as the **Environment type** if the Process Server you are configuring, will be accessed and used to review content and new functionality.

  There are three types of environments available for selection:

  Production
  > Select **Production** if the server will serve in a production capacity.

  Stage   Select **Stage** if the server will serve as a staging platform to be used as a preproduction server.

  Test   Select **Test** if the server you are configuring will be used as a testing environment.

- **Use server offline**

  Indicate whether the server you are configuring is an offline server.

  An offline server is a Process Server that is not connected to the Process Center.

  Offline servers can still be used when deploying snapshots of process applications. However the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

- **Protocol**

  Select either **http://** or **https://** as the connection protocol to the Process Center.

- **Host name or virtual host in a load-balanced environment**

  Type the host or virtual host that this Process Server needs to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port**

  Type the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.

- **User name**

Type a valid user name that exists on the Process Center. Process Server will connect to Process Center as this user.

- **Password**

  Type the password for the above user.

11. On the Security page, configure the authentication aliases WebSphere uses when accessing secure components

   You can change the authentication alias user name and password on this page. These aliases are used to access secure components but do not provide access to data sources

12. Optional: On the System web applications page, set the context root for component-based web applications in your deployment environment or accept the system-provided default values for the context roots. Then click **Next** to display the Summary page.

   The System web applications page displays for deployment environments using the Remote messaging, support and web applications pattern.

   The table contains the following control information.

   Web Application
   > The name of the Web application.
   >
   > Some of the components that are part of the deployment environment you are creating contain web applications. The **Web application** column can include the following components:
   > - Business Process Choreographer Explorer
   > - Business Rules Manager

   Context Root
   > The current value of the context root for the component.
   >
   > By default, the default context root for the web application applies. You can change the context roots by typing over the value in the **Context Root** field.
   >
   > **Note:** The Business Space context root is read only and cannot be edited.

13. Verify that the information on the Summary page is correct and perform the following substeps:

   a. Optional: If you do not want to save the deployment environment configuration, you can click **Cancel**.

   b. Optional: If you want to exit without generating the configuration, click **Finish**.

      To get back to the panel (if you exited without completing), perform the following from the administrative console: **Deployment Environments** > *name of deployment environment* > **Generate Environment** .

   c. To save the deployment environment configuration, click **Finish** and from within the Messages window, click **Save**.

      Clicking **Save** saves the deployment environment to the master configuration. If an error occurs during deployment environment generation, the configuration settings are saved to the master configuration.

   d. Check for deferred configuration steps

      Select **Deployment Environments** > *name of deployment environment* > **Deferred Configuration**

      You need to address any existing deferred configuration steps before starting the Deployment Environment.

   e. If you are satisfied with the deployment environment configuration and you have addressed any of the deferred configuration steps, click **Finish and Generate Environment** to save and complete the configuration of the deployment environment.

14. Run the **bootstrapProcessServerData.sh** command . Refer to "Loading the database with system information" on page 581.

Note: This command must be run before starting the Process Center or Process Server cluster members.

When the configuration completes, you can examine the configuration files to view the changes.

Either save the changes to the master configuration or discard them.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

"Overview: Deployment environment topologies and patterns" on page 33
A network deployment environment can have many topologies, and can be created from several standard topology patterns.

"Topology patterns and supported product features" on page 48
A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

⊳ Virtual hosts

Related tasks:

"Completing the database configuration" on page 615
Before you start the clusters in your network deployment environment, you must complete the database configuration by creating the database tables.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

⊳ Configuring virtual hosts

Creating the Process Center deployment environment using a pattern:

After determining the pattern on which to base your network deployment configuration, use the Deployment Environment Configuration wizard to create the IBM® Process Center deployment environment that is based on the pattern.

On the administrative console of the deployment manager navigate to **Servers > Deployment Environments**.

Required security role for this task: When security and role-based authorization are enabled, you must log in to the administrative console as an administrator or configurator to perform this task.

The procedure for creating deployment environments using the deployment environment wizard includes steps for selecting patterns and features, and therefore it is assumed that you have read and understood the information on patterns and features documented in the planning section.

It is assumed that you have installed the product and that you have created the deployment manager profile and the associated nodes.

Additionally, one of the steps in the Deployment Environment Configuration wizard includes importing a database design document. The database design document defines the database configuration for the selected deployment environment features. IBM Business Process Manager includes a response-driven database design tool (DDT) that creates a database design document based on user inputs. The document then can be used by the DDT to create the database scripts and by the IBM Business Process Manager

deployment environment wizard to configure the databases used in the deployment environment. For more information on the DDT and for more information on database configuration in general, see *Configuring databases*.

This task describes the procedure for creating a deployment environment that is based on a specific pattern and uses the Deployment Environment Configuration wizard.

Note: If you make an error while you are working in the wizard, you can go back by clicking **Back**.

1. From the administrative console, go to the Deployment Environments page by clicking **Servers** > **Deployment Environments** .

2. Launch the Deployment Environment Configuration wizard by clicking **New** on the Deployment Environments page.

   a. The **Create a deployment environment based on a pattern** option is selected. **Create a deployment environment based on a pattern** is the system default and it is the option described in this topic.

   Deployment environment patterns capture commonly used business integration topologies. A pattern provides you with a template for the deployment environment that you are creating.

   Note: Patterns have a direct relationship to the products supported by the configured deployment manager. IBM Business Process Manager supports a specific set of patterns, with the *Remote messaging and remote support* pattern being the system default. If your deployment manager supports other products in addition to IBM Business Process Manager, additional patterns may apply. Consult product-specific documentation for information on patterns as they apply to the products.

   For information about the patterns included with and supported by IBM Business Process Manager, see Overview: Deployment environment topologies and patterns.

   b. Enter a unique name for the deployment environment in the **Deployment environment name** field.

   c. Optional: To view all of the configuration steps in the wizard, select **Detailed: Show all steps**.

   If you choose **Fast path: Show only needed steps** the wizard displays only those pages that **do not** have assigned default values. Choose **Fast path: Show only needed steps** only if you are agreeable to accepting the system-provided default values for the deployment environment configuration.

   This topic assumes that you have chosen **Detailed: Show all steps**

   d. Click **Next** to display the Deployment Environment Features page.

3. On the Deployment Environment Features page, select the feature for the deployment environment and click **Next** to either view a list of compatible features, or to view a list of deployment environment patterns. Features represent the runtime processing capabilities of your deployment environment.

   The list of available features on the Deployment Environment Features page is based on the deployment manager profile. If your deployment manager profile has been augmented to include other products alongside IBM Business Process Manager (for example, IBM Business Monitor , then the Deployment Environment Features page also lists these features.

   If you have installed and configured a profile for IBM Business Process Manager, then the Deployment Environment Features page includes the following:

   • For IBM Business Process Manager Standard:

     – **BPMSPS**, for an IBM BPM Standard, Process Server deployment manager profile, which provides a Process Server deployment environment that supports business processes, human tasks, and business rules.

     – **BPMSPC**, for an IBM BPM Standard, Process Center deployment manager profile, which provides a Process Center deployment environment that supports business processes, human tasks, and business rules.

- For IBM Business Process Manager Advanced:
  - **WESB**, for WebSphere Enterprise Service Bus, which provides a deployment environment that supports mediations.
  - **BPMAPS**, for an IBM BPM Advanced, Process Server deployment manager profile, which provides a Process Server deployment environment that supports mediations, business processes, human tasks, Business Space functions and business rules.
  - **BPMAPC**, for an IBM BPM Advanced, Process Center deployment manager profile, which provides a Process Center deployment environment that supports mediations, business processes, human tasks, Business Space functions and business rules.

4. On the Select compatible deployment environment features page, select additional features as necessary and click **Next** to view the list of patterns associated with your feature selections.

   Note: The Select compatible deployment environment features page is displayed only if the deployment manager has been augmented with other business process management (BPM) features, such as IBM Business Monitor.

   For an understanding of the relationship of features and compatible features, see the information on deployment environments in the Planning section.

5. On the Select the deployment environment pattern page, select the pattern for the selected deployment environment, then click **Next** to display the Select Nodes page.

   The list of patterns that display on the Deployment Environment Patterns page is dynamic. This list is activated by, and dependent on, the following environment conditions and configuration decisions:
   - The platform on which you have installed the software
   - The selections that you have made on the Select the deployment environment feature page and the Select compatible deployment environment features page.

6. Optional: On the Select Nodes page, select the nodes that you want to include in this deployment environment, then click **Next** to display the Clusters page.

   Select nodes that have the required capabilities for the environment you selected in step 3 on page 600. For example, if you selected **BPMSPC** as your Deployment Environment type, the nodes selected should address the capabilities of that environment type.

   Select at least one node for the deployment environment. For high-availability and failover environments, select at least two nodes. For scalability, select all nodes.

   To include a node, select the check box next to the node name. Use **Node Mapping** to map the selected node to another node name.

7. Optional: On the Clusters page, assign the required number of cluster members on each node for each cluster *type* (Application Deployment Target, Messaging Infrastructure and Supporting Infrastructure) of the deployment environment.

   By default one cluster member is assigned on each node for each function. You change the number by replacing the number in each column. If you are unfamiliar with the different cluster roles and functions provided by each type of cluster, see "Topology types and deployment environment patterns."

   A 0 (zero) value for a node means that the node does not contribute to the selected function, based on features that you have selected.

   After assigning cluster members, you can click **Next** to display the Cluster naming pages for each cluster type of the deployment environment. The Cluster naming substeps that display will vary depending on the deployment environment pattern selected.

   The system generates default values for cluster names and cluster member names.

   If you do not want to customize cluster names or cluster member names, you can use the wizard navigation pane to go directly to the REST Services page in a following step.

   a. Optional: Customize the cluster names and cluster member names.

      Use the Cluster Naming page to customize cluster names or cluster member names for the cluster type. There is one substep page for each cluster *type* in the pattern that you have selected.

For example, if you selected a **Remote messaging and remote support pattern**, there are 3 substeps, one for each type of cluster (Application Deployment Target, Messaging Infrastructure and Supporting Infrastructure) in that pattern.

The information on each substep page is as follows:

Field	Description	Value
Cluster	A read-only field specifying the functional role of the cluster.	The value varies depending on the cluster type, as follows: • Application Deployment Target • Supporting Infrastructure • Messaging Infrastructure • Web Application Infrastructure For information on the functional role provided by each cluster type, see Topology types and deployment environment patterns

Cluster name	The system-generated default value for the cluster name.	The default values are based on a naming convention of *Deployment Environment Name.Cluster type name*, where *Cluster type name* is one of the following values:
		• AppTarget
		  For clusters performing the role of application deployment target
		• Messaging
		  For clusters performing the role of messaging infrastructure
		• Support
		  For clusters performing the role of supporting infrastructure
		• Web
		  For clusters performing the role of supporting web applications. **Note:** This cluster type name applies for BPM configurations in which WebSphere Business Monitor is the primary feature / product.

Cluster member name	The system-generated default value for the cluster member name. Servers that are a part of a cluster are called cluster members.	Accept the system-generated default value or specify a name of your choosing. The default value for the cluster member name is based on the following naming convention: *cluster name.node name.node number sequence*. The number of cluster member names that display in the table match the number of cluster members that you entered for the cluster type column and node row on the Clusters page.

8. Optional: On the Import the database configuration page, click **Browse** to go the database design document or enter the path to the database design document and then click **Next** to go to the Data sources page. If you import a design document, the information from the design document is reflected on the Database page of the wizard. The design document can be based on a database

design that you created using the database design tool (DDT), or it can be the supplied design document based on the pattern and feature that you have selected.

Note: The database design document that you import for the deployment environment does not change the commonDB created at Profile Creation time.

9. Required: On the Database page, configure the database parameters for data sources of the deployment environment, then click **Next** to go to the Security page.

Note: The database specified in this panel must already exist. Deployment environment configuration never creates a new database.

For DB2 and SQL Server databases, IBM Process Server and IBM Performance Data Warehouse should not use the same database as the rest of the components. However, if you are using an Oracle database, IBM Process Server and IBM Performance Data Warehouse can use the same database instance, but should use different users.

On this page, define the database information for the components that are included in this deployment environment. Where possible, the wizard supplies default information for the parameters, but change those values to match the values that you defined when you planned the environment.

Note: If you imported a database design document, the information on the Database page reflects the data source configuration as it exists in the database design document that you imported. If you make changes to the data source configuration after importing a database design document, your changes might be incompatible with the DDL generated by the database design tool and the original values.

Whether or not this step displays for a fast path deployment environment configuration is conditional. This step displays for a fast path deployment environment configuration if more than one database has been defined.

This step always displays if you are using DB2 for z/OS or an Oracle database provider.

Note: The default schema names that are displayed on this page might conflict with your site naming convention or might conflict with existing schemas. As such, it is likely that you will need to change the schema name. Pay close attention to the values specified to avoid potential naming conflicts.

Oracle database considerations:

- Make sure that the username/schema name are exactly the same. The user specified should exist in the database before generating the environment.

SQL Server considerations:

- Make sure that the username/schema exist before the configuration is done. The schema value should be the default schema for the user chosen.
- To indicate that users will connect to the databases using Windows credentials, select the individual data source, click **Edit**, and select **Apply Windows authentication**.

For a production environment, you should set the same values for **User name** and **Schema name** and you should deselect **Create tables**. For a production environment, create the required schemas manually and use the SQL files generated to create the tables.

You can edit all key parameters, such as the database name, whether or not to create tables, the data source runtime user name, and the password for the deployment environment.

You can select which database to use for the given component.

DB2 for z/OS: The **Create tables** option cannot be used if you are using a DB2 for z/OS database provider.

Steps that cannot be completed through the Deployment Environment Configuration wizard, and which need to be completed manually, are listed on the Deferred Configuration page.

10. On the Process Center page, set the values for the Process Center configuration as follows:

- **Environment name**

  Enter an environment name of the Process Server.

  An environment name is the name by which this server or cluster will be known to a Process Center user.

- **Environment type**

  From the pull-down list, select the Environment type for the Process Center you are configuring.

  The Environment type refers to how the Process Center is used. For example, in what capacity will the Process Server be used - *production*, *stage* or *test* . Load testing might be done on a test server, while a stage environment type might be used as a temporary location to host changes before putting those changes into production. You might specify **Stage** as the **Environment type** if the Process Center you are configuring, will be accessed and used to review content and new functionality.

  There are three types of environments available for selection:

  Production
  >   Select **Production** if the server will serve in a production capacity.

  Stage   Select **Stage** if the server will serve as a staging platform to be used as a preproduction server.

  Test   Select **Test** if the server you are configuring will be used as a testing environment.

- **Use server offline**

  Indicate whether the server you are configuring is an offline server.

  An offline server is a Process Server that is not connected to the Process Center.

  Offline servers can still be used when deploying snapshots of process applications. However the method for deploying process applications to an offline process server differs from the method for deploying process applications to an online process server.

- **Protocol**

  Select either **http://** or **https://** as the connection protocol to the Process Center.

- **Host name or virtual host in a load-balanced environment**

  Type the host or virtual host that this Process Server needs to communicate with Process Center. Use a fully qualified host name. In an environment with a load balancer or proxy server between the Process Server and the Process Center services, make sure that what you designate here matches the URL for accessing the Process Center.

- **Port**

  Type the port number of the Process Center. In an environment with a load balancer or proxy server between the Process Server and the Process Center, make sure that what you designate here matches the URL for accessing the Process Center.

11. On the Security page, configure the authentication aliases WebSphere uses when accessing secure components

    You can change the authentication alias user name and password on this page. These aliases are used to access secure components but do not provide access to data sources

12. Optional: On the System web applications page, set the context root for component-based web applications in your deployment environment or accept the system-provided default values for the context roots. Then click **Next** to display the Summary page.

    The System web applications page displays for deployment environments using the Remote messaging, support and web applications pattern.

    The table contains the following control information.

    Web Application
    >   The name of the web application.

Context Root

The current value of the context root for the component.

By default, the default context root for the web application applies. You can change the context roots by typing over the value in the **Context Root** field.

Note: The Business Space context root is read only and cannot be edited.

13. Verify that the information on the Summary page is correct and perform the following substeps:

   a. Optional: If you do not want to save the deployment environment configuration, you can click **Cancel**.

   b. Optional: If you want to exit without generating the configuration, click **Finish**.

   c. To save the deployment environment configuration, click **Finish** and from within the Messages window, click **Save**.

      Clicking **Save** saves the deployment environment to the master configuration. If an error should occur during deployment environment generation, the configuration settings will have been saved to the master configuration.

   d. Check for deferred configuration steps

      Select **Deployment Environments** > *name of deployment environment* > **Deferred Configuration**

      You need to address any existing deferred configuration steps before starting the Deployment Environment.

   e. If you are satisfied with the deployment environment configuration and you have addressed any of the deferred configuration steps, click **Finish and Generate Environment** to save and complete the configuration of the deployment environment.

14. Run the **bootstrapProcessServerData.sh** command . Refer to "Loading the database with system information" on page 581.

When the configuration completes, you can examine the configuration files to view the changes.

Either save the changes to the master configuration or discard them.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

"Overview: Deployment environment topologies and patterns" on page 33
A network deployment environment can have many topologies, and can be created from several standard topology patterns.

"Topology patterns and supported product features" on page 48
A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

Virtual hosts

Related tasks:

"Completing the database configuration" on page 615
Before you start the clusters in your network deployment environment, you must complete the database configuration by creating the database tables.

"Loading the database with system information" on page 581
You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

Configuring virtual hosts

Configuring a network deployment environment using the administrative console:

After you perform a custom installation and create the deployment manager and custom (managed node) profiles, you can create a network deployment environment using the administrative console.

The information in this section describes how to use the administrative console to create a network deployment configuration.

This section assumes that the following information is true:
- You have run the installer to create a Custom installation.
- You have created the deployment manager and custom (managed node) profiles.
- You are familiar with network deployment topologies and configurations and the administrative console.

Important: Consider using the Deployment Environment wizard to create your network deployment environment.

Any operation that you can perform from the administrative console can also be performed with wsadmin. Additionally, command assistance is available for a subset of administrative console actions. When available, command assistance displays the wsadmin scripting command for the last console action that you performed. For information about command assistance, see Administrative console actions with command assistance.

Creating and configuring servers and clusters:

You can use the administrative console to create and configure the servers and clusters for IBM Business Process Manager.

The information in this section describes how to create servers and clusters for IBM Business Process Manager manually using the administrative console.

Creating a cluster:

The following instructions explain how to create a cluster with one cluster member. The benefit of using the administrative console to create a cluster is that you can undo your changes as you go and you can use a graphical user interface.

Before you create a cluster using the administrative console, start the deployment manager.

The following procedure describes how to create a cluster with one cluster member using the administrative console. Note that the tasks performed from the administrative console that are described in this topic can also be performed using administrative scripting. For information on the **createCluster** parameters, see ClusterConfigCommands command group for the AdminTask object in the WebSphere Application Server information center.

For information on using the wsadmin tool to create cluster members, see Creating cluster members using scripting.

Consider to create a cluster even if a single server is currently sufficient for your high availability and scalability requirements because it is easier to add more servers to the cluster later.

1. Log in to the administrative console and navigate to **Servers** > **Clusters** > **WebSphere application server clusters**.
2. Click **New** to display the Create new cluster page.
3. From the Create new cluster page, enter basic cluster information:

a. Type a name for the cluster in the **Cluster name** field.

b. Select **Prefer local** if you want to enable host-scoped routing optimization. This option is enabled by default. When this option is enabled, if possible, EJB requests are routed to the client host. This option improves performance because client requests are sent to local enterprise beans.

   Note: If you enable the preferLocal optimization, the deployment manager must be running to affect the configuration. If the deployment manager is shut down, preferLocal optimization is not performed and requests might be dispersed across all the members of the cluster

c. Select **Configure HTTP session memory-to-memory** replication if you want a memory-to-memory replication domain created for this cluster.

   The replication domain is given the same name as the cluster and is configured with the default settings for a replication domain. When the default settings are in effect, a single replica is created for each piece of data and encryption is disabled. Also, the Web container for each cluster member is configured for memory-to-memory replication.

   To change these settings for the replication domain, click **Environment** > **Replication domains** > *replication_domain_name*. To modify the Web container settings, click **Servers** > **Clusters** > **WebSphere application server clusters** > *cluster_name* > **Clusters members** > **cluster_member_name**. Then, in the **Container settings** section, click **Web container settings** > > **Web container** > **Session management** > **Distributed environment** settings in the administrative console. If you change these settings for one cluster member, you might also need to change them for the other members of this cluster.

4. Click **Next** to go to the Create first cluster member page.

5. From the Create first cluster member page, enter information about the cluster member.

   a. Enter the member name in the **Member name** field

   a. From the **Select node** list, click the node in which you want to define the server.

   b. In the **Weight** field, enter the weight value for the cluster member.

   The weight value controls the amount of work that is directed to the application server. If the weight value for this server is greater than the weight values that are assigned to other servers in the cluster, then this server receives a larger share of the workload. The weight value represents a relative proportion of the workload that is assigned to a particular application server. The value can range from 0 to 20.

   - On a **z/OS system** weight is used to balance some of the workload types, but others are balanced by the z/OS system. For HTTP requests, weights are used to distribute HTTP traffic between the Web server plug-in and the controller handling the clustered application server. Assign a higher weight value to the application server that should receive the HTTP traffic.

   - For Web services calls, information is transferred from a servant in one application server to a controller in another application server. The application server that receives the call has the highest weight value.

   - Weight has no affect on Internet Inter-ORB Protocol (IIOP) requests. IIOP requests are distributed to the correct application server using the sysplex distributor.

   c. Select **Generate unique HTTP ports** (the default option) if you want to generate unique port numbers for every HTTP transport that is defined in the source server.

   When this option is selected, this cluster member does not have HTTP transports or HTTP transport channels that conflict with any of the other servers that are defined on the same node. If you clear the **Generate unique HTTP ports** check box, all of the cluster members will share the same HTTP ports.

6. From the section, **Select basis for first cluster member**, select from the following options:

   - Create the member using an application server template

   This is the typical way of creating a cluster. Select **defaultProcessServer**.

   If you select this option and click **Next**, a blank form is displayed which you can use to define additional cluster members. The server you have just created is listed at the bottom of the screen.

    a. Click **Next**.

    b. Check the details on the summary screen and click **Next**.

    c. Save your configuration changes.

       The cluster you have just created is displayed in the list.

    d. Customize the port numbers to suit your configuration.

- Create the member using an existing application server as a template

  This option is not supported.

- Create the member by converting an existing application server

  Note: Do not select this option. WebSphere Application Server added a new parameter named **resourcesScope** to the createCluster and createClusterMember commands that allows you to control how the resources will be processed when the first cluster member is added to a cluster. Those parameters are described in the technote titled New parameter named resourcesScope was added to the createCluster and createClusterMember commands.

  It is required to use the **resourcesScope** parameter with the value **cluster** for IBM Business Process Manager. The **resourcesScope** parameter is not currently exposed in the administrative console. Using the administrative console to create a cluster using an existing single server as the first cluster member is not supported on IBM Business Process Manager. Selecting this option will result in the following error message in the administrative console or the SystemOut.log of the deployment manager:

  ```
  resourcesScope parameter needs to be set to 'cluster' if creating a cluster from an existing server
  ```

  Using the command line to create a cluster using an existing single server as the first cluster member is valid. You must set the value of the **resourcesScope** to **Cluster**. For example,

  ```
  $AdminTask createCluster {-clusterConfig
  {-clusterName newcluster -preferLocal true}
  -convertServer {-serverNode NODE1Node01 -serverName testserver
  -resourcesScope cluster}}
  ```

  For information on the **createCluster** parameters, see ClusterConfigCommands command group for the AdminTask object in the WebSphere Application Server information center.

  For information on using the wsadmin tool to create cluster members, see Creating cluster members using scripting.

- None

  Works always.

7. Click **Next.**

8. Create additional cluster members.

   Before you create additional cluster members, check the configuration settings of the first cluster member. These settings are displayed at the bottom of the Create additional cluster members panel of the Create a new cluster wizard. For each additional member that you want to create:

   a. Specify a unique name for the member. The name must be unique within the node.

   b. Select the node to which you want to assign the cluster member.

   c. Specify the weight you want given to this member. The weight value controls the amount of work that is directed to the application server. If the weight value for the server is greater than the weight values that are assigned to other servers in the cluster, then the server receives a larger share of the workload. The value can range from 0 to 20.

   d. Select **Generate unique HTTP ports** if you want to generate unique port numbers for every HTTP transport that is defined in the source server.

   e. Click **Add member**. You can edit the configuration settings of any of the newly created cluster members other than the first cluster member, or you can create additional cluster members. Click

Previous to edit the properties of the first cluster member. The settings for the first cluster member become the settings for the cluster member template that is automatically created when you create the first cluster member.

9. When you finish creating cluster members, click **Next**.

10. View the summary of the cluster and then click **Finish** to create the cluster, click **Previous** to return to the previous wizard panel and change the cluster, or click **Cancel** to exit the wizard without creating the cluster.

11. To further configure a cluster, click **Servers** > **Clusters** > **WebSphere application server clusters** , and then click the name of the cluster. Only the **Configuration** and **Local Topology** tabs display until you save your changes.

12. Click **Review** to review your cluster configuration settings. Repeat the previous step if you need to make additional configuration changes.

13. If you do not want to make any additional configuration changes, select **Synchronize changes with Nodes** and then click **Save**. Your changes are saved and synchronized across all of your nodes.

   Note: If you click **Save** , but do not select **Synchronize changes with Nodes**, when you restart the cluster, the product does not start the cluster servers because it cannot find them on the node. If you want to always synchronize your configuration changes across your nodes, you can select Synchronize changes with Nodes as one of your console preferences.

14. Restart the cluster.

The cluster is created with your chosen server in the selected managed node as the first cluster member.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

⮕ Virtual hosts

Related tasks:

⮕ Configuring virtual hosts

Creating a new server:

Most installations require several servers to handle the application serving needs of the production environment. You can use the command-line tool or the administrative console to create the servers you need.

Determine if you want to include the new server in a cluster. If this server is going to be part of a cluster, you must create the server with the Create a new cluster wizard instead of the Create a new application server wizard.

Important: This task creates a managed server. If you want a stand-alone server, do not follow these steps. Instead, create a stand-alone server profile.

To create a new managed server, perform the following steps.

Follow the instructions in Creating application servers, selecting the template or a suitable user-defined template from the Select a server template page. **Restriction:** The "Start components as needed" capability is not supported.

Configure the components you need on the server. See **Configuring components** for more information.

Creating deployment environments using the command line:

You can use wsadmin to create a Process Server and Process Center deployment environment. The **createDeploymentEnvDef** and **generateDeploymentEnv** provide a command-line equivalent to creating the deployment environment using the deployment environment wizard.

Related concepts:

"Configuring a network deployment environment using the deployment environment wizard" on page 592
After performing a Custom installation and creating the deployment manager and custom (managed node) profiles, you can create a network deployment configuration.

"Overview: Deployment environment topologies and patterns" on page 33
A network deployment environment can have many topologies, and can be created from several standard topology patterns.

"Topology patterns and supported product features" on page 48
A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

⮕ Virtual hosts

Related tasks:

⮕ Configuring virtual hosts

Creating deployment environment definitions using the command line:

You can create a deployment environment definition using the **wsadmin** command. Running **createDeploymentEnvDef** provides the definition of the deployment environment.

You must be on the deployment manager to create the deployment environment definition.

Required security role for this task: When security and role-based authorization are enabled, you must use a userid and password with administrator or operator authority to perform this task.

IBM Business Process Manager supports a set of patterns, *Remote messaging and remote support* being the pattern to use for a network deployment production environment. If your deployment manager supports other products in addition to IBM Business Process Manager, you might want to use patterns for those products when creating a deployment environment definition. For information about patterns for other products, see the documentation for those products.

This task creates a deployment environment definition that is based on a specific pattern and uses the **wsadmin** command.

You can use the **wsadmin** command to create the same deployment environment as you can create from the administrative console. This capability allows you to run the administrative task to create a deployment environment definition with all the default values based on an existing configuration (the configuration that you created at profile creation time). The command also includes an optional property that imports a database design document. The database design document holds the database configuration for the topology you are creating.

When you generate the deployment environment, the information about whether to create tables is taken from the design document. Make sure that the **createTables** parameter is correctly set in the database design file that you specify.

Important: If you use the `createTables` parameter, the databases must already exist. Do not use `createTables` for a production environment where you want to customize the generated database scripts. Do not use `createTables` if you have a remote database server.

For DB2 and SQL Server databases, IBM Process Server and IBM Performance Data Warehouse should not use the same database as the rest of the components. However, if you are using an Oracle database, the IBM Process Server and IBM Performance Data Warehouse can use the same database instance, but should use different users.

For Oracle databases, make sure that the user name and schema name are exactly the same. The user must exist in the database before you generate the environment.

For SQL Server databases, make sure that the user name and schema exist before the configuration is done. The schema value must be the default schema for the user chosen.

For a production environment, set the same values for user name and schema name and do not select `createTables`. Create the required schemas manually and use the generated SQL files to create the tables.

A *deployment environment definition* describes the specific component, configuration (of clusters, nodes, and servers), resources and related configuration parameters that make up a deployment environment. This definition can also be referred to as an instance of a deployment environment configuration. A deployment environment configuration can be exported into a deployment environment definition. You can import a deployment environment definition to add a new deployment environment configuration to your system.

1. Open a command window. The **wsadmin** command can be found in either of the following directories:

   ```
   install_root/profiles/dmgr profile/bin
   install_root/bin
   ```

2. At the command prompt, enter the **wsadmin** command to enter the **wsadmin** environment.

3. Use the **createDeploymentEnvDef** command to create the deployment environment definition with a specific name for a particular runtime and pattern.

   Note: If administrative security is on, and you did not supply a user ID and password in the command, you are prompted for a user ID and password.

This example creates a deployment environment definition for a remote messaging and remote support pattern on the IBM Business Process Manager runtime, with myDepEnv on the host myDmgr with administrative security enabled. The example imports a database design document named `nd.topology.dbDesign`:

Note: The value you assign to *-topologyRuntime* varies depending on the IBM BPM configuration (Advanced or Standard) and the environment selection. For example:

- For IBM BPM Standard, **-topologyRuntime** can be:
  - **BPMSPS**, for IBM BPM Standard Process Server
  - **BPMSPC**, for IBM BPM Standard Process Center
- For IBM BPM Advanced, **-topologyRuntime** can be:
  - **BPMAPS**, for IBM BPM Advanced Process Server
  - **BPMAPC**, for IBM BPM Advanced Process Center

Note: If you disable administrative security, you do not need to provide a user ID and password.

After you have imported or created a deployment environment on a deployment manager, you can configure the deployment environment using the **generateDeploymentEnv** command.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Related concepts:

"Overview: Deployment environment topologies and patterns" on page 33
A network deployment environment can have many topologies, and can be created from several standard topology patterns.

"Topology patterns and supported product features" on page 48
A topology is the physical layout of the deployment environment. The product features and default usage depends on your choice of topology pattern.

Virtual hosts

Related tasks:

Configuring virtual hosts

"Generating deployment environments using the command line" on page 613
You can generate deployment environments using the **wsadmin** interface. This capability allows you to configure multiple deployment environments unattended on a deployment manager using a script.

Adding nodes to a deployment environment definition using the command line:

You can add nodes to a deployment environment definition using the **wsadmin** command.

The task assumes that the node has been federated to the deployment manager.

This command to add a node to the deployment environment definition will fail if the topology is already configured.

You must be on the deployment manager to add nodes.

Required security role for this task: When security and role-based authorization are enabled, you must use a userid and password with administrator or operator authority to perform this task.

This task adds a federated node to a deployment environment definition and uses the **wsadmin** command.

1. Open a command window. The **wsadmin** command can be found in either of the following directories:
   ```
   install_root/profiles/dmgr profile/bin
   install_root/bin
   ```
2. At the command prompt, enter the **wsadmin** command to enter the **wsadmin** environment.
3. Enter the **addNodeToDeploymentEnvDef** command to add the node to the deployment environment definition.

   Note: If administrative security is on, you will be prompted for a user ID and password, if you do not supply it in the command.

This example adds a node (**MyNode**) to deployment environment definition (**myDepEnv**) with administrative security enabled:

Attention: If you are adding a node to a single cluster topology pattern, the value for **-toplogyRole** must be set to **ADT**. Deployment environment topology patterns are specified when you create the deployment environment using either the **createDeploymentEnvDef** command or the Deployment Environment Configuration wizard.

Note: If you disable administrative security, you do not need to provide a user ID and password.

Important: If you use additional servers with unique ports, WebSphere Application Server does not automatically configure the virtual host for the server. Specifically, WebSphere Application Server does not automatically add the host alias ports to a virtual host. However, you can use the administrative console to add a new host alias for each of the ports that are used by the new server. For more information, see the WebSphere Application Server documentation about configuring virtual hosts.

Generating deployment environments using the command line:

You can generate deployment environments using the **wsadmin** interface. This capability allows you to configure multiple deployment environments unattended on a deployment manager using a script.

You must enter the commands on the deployment manager on which you are configuring deployment environments.

Required security role for this task: When security and role-based authorization are enabled, you must log in to the administrative console as an administrator or configurator to perform this task.

After you have imported or created deployment environments on a deployment manager, you can configure the deployment environments using the **generateDeploymentEnv** command.
1. Enter the **wsadmin** environment.
2. Enter the **generateDeploymentEnv** command for each topology you are configuring.

The following command configures the eastEnvironment topology on host myDmgr.

```
wsadmin -connType SOAP -host myDmgr -port 8879
> $AdminTask generateDeploymentEnv {-topologyName eastTopology}
> $AdminConfig save
```

Note: If administrative security is enabled, you are prompted for a user ID and password after the system processes the **wsadmin** command.

Save the configured deployment environments. From the command line, enter **$AdminConfig save**.

Validating the deployment environment definition from the command line:

You can validate the deployment environment definition using the **wsadmin** command.

The task assumes that the node has been federated to the deployment manager.

You must be on the deployment manager where you generated the deployment environment definition.

Required security role for this task: When security and role-based authorization are enabled, you must use a userid and password with administrator or operator authority to perform this task.

This task validates the deployment environment definition and uses the **wsadmin** command.
1. Open a command window. The **wsadmin** command can be found in either of the following directories:
   install_root/profiles/*dmgr profile*/bin
   install_root/bin
2. At the command prompt, enter the **wsadmin** command to enter the **wsadmin** environment.
3. Enter the **validateDeploymentEnvDef** command to validate the deployment environment definition.

   Note: If administrative security is on, you will be prompted for a user ID and password, if you do not supply it in the command.

This example validates the deployment environment definition (**myDepEnv**) with administrative security enabled:

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgrAdmin -password -dmgrPass
> $AdminTask validateDeploymentEnvDef { -topologyName topOne}
```

Note: If you disable administrative security, you do not need to provide a user ID and password.

Displaying deployment environment status using the command line:

You can display the current status of a deployment environment using the **wsadmin** command.

The admin client must connect to the deployment manager for which you are displaying the status.

Required security role for this task: When security and role-based authorization are enabled, you must use a userid and password with administrator or operator authority to perform this task.

This task displays the current status of a deployment environment and uses the **wsadmin** command.

1. Open a command window. The **wsadmin** command can be found in either of the following directories:

   install_root/profiles/*dmgr profile*/bin
   install_root/bin

2. At the command prompt, enter the **wsadmin** command to enter the command environment.

   Note: Make sure **wsadmin** connects to the correct deployment manager, when running in connected mode.

3. Use the **showDeploymentEnvStatus** command to show the current status of the deployment environment.

   Note: If administrative security is on, you will be prompted for a user ID and password, if you do not supply it in the command.
   The following table lists the results that might be returned.

   Note: Some of the states listed in the table are valid for configured topologies only. The states that are apply to configured topologies only are noted as such.

Table 104. States of a topology instance in order of least to most available

State	Description
Incomplete	The deployment environment is not missing any elements but is incomplete in some way. Incomplete state may mean the deployment environment is missing a required role, node, comp or dependencies . The warning message contains additional details.
Complete	This state is also known as *Not configured* and it means that the configuration is known and complete but has not yet been generated.
Configured	This means the configuration is in synch.
Partially configured	The deployment environment has been generated but deferred configuration has not been completed.
Unknown	The system cannot determine the current state of the deployment environment. A resync operation could be performed on this state.
Stopped	State applies to configured topologies only.All deployment targets in the topology are stopped.
Running	State applies to configured topologies only.The deployment environment is available and all functions are running.

Table 104. States of a topology instance in order of least to most available (continued)

State	Description
Partially started	State applies to configured topologies only.The deployment environment is available but at least one function is partially running.
Starting	State applies to configured topologies only.The deployment environment is starting.
Partially stopped	State applies to configured topologies only.The deployment environment is available but at least one function is stopped or partially stopped.
Stopping	State applies to configured topologies only.The deployment environment is stopping
Unavailable	State applies to configured topologies only.The deployment environment state is unavailable.

This example displays the status of a deployment environment (**MyDepEnv**) on the host (**myDmgr**) with administrative security enabled.

Note: If you are running the admin client from the deployment manager bin folder, you do not need to include the **-host** and **-port** parameters in the command.

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgradmin -password dmgrpass
> $AdminTask showDeploymentEnvStatus {-topologyName myDepEnv}
```

The **-connType** parameter specifies the type of connection to be used; the default argument is **SOAP**. Because **SOAP** is the default, you do not need to give it explicitly.

The **-host** parameter specifies the host used for the SOAP or RMI connection. The default value for **-host** is the local host. If the node is running on the local host, you do not need to specify **-host**.

Note: If you disable administrative security, you do not need to provide a user ID and password.

Completing the database configuration

Before you start the clusters in your network deployment environment, you must complete the database configuration by creating the database tables.

Creating the Process Center or Process Server database tables:

Run the SQL scripts to create the database tables for your IBM Process Center or IBM Process Server configuration.

You have performed all of the steps required for your configuration and have selected not to run the database scripts to initialize the databases during the configuration. You have also created the database that you want to create the Process Center or Process Server database tables in. For information about creating the database, see *Creating databases*

To run the Process Center or Process Server SQL scripts that were produced when you completed your configuration follow the steps below:

Note: The Process Server does not support schema. You must login as the user and then execute the SQL file.

1. Go to the directory that holds the SQL scripts for creating the database tables.

   By default, SQL scripts are output to *profile_root*/dbscripts/ProcessServer/*DB_product* at profile creation time. For example,

   profile_root/dbscripts/ProcessServer/DB2

2. Run the following command as a root user or administrator.

   • ▬ Linux ▬   ▬ UNIX ▬   **configProcessServerDB.sh createDB**

- **`Windows`** **configProcessServerDB.bat createDB**

Note: The **createDB** parameter is used to create a local database. Skip this parameter if you have already created the required database. The **createDB** parameter is not applicable for an Oracle database and must be excluded from the command.

3. For an SQL database, create the tables for Process Center or Process Server.

   From the directory navigated to in step 1, run **createTable_ProcessServer.sql** on the database you configured for the Process Center or Process Server (BPMDB).

   Make sure all SQL statements succeed without any errors.

4. For an SQL database, create the table procedures for Process Center or Process Server.

   From the directory navigated to in step 1, run **createProcedure_ProcessServer.sql** on the database you configured for the Process Center or Process Server (BPMDB).

   Make sure all SQL statements succeed without any errors.

5. Reset the connection.

You have created the database tables for your Process Center or Process Server configuration and have completed the database configuration.

You can now load the database with system information and start the servers or clusters on your Process Center or Process Server configuration.

Related tasks:

"Creating databases using the provided scripts" on page 555
If you did not create databases while creating profiles, database script files are generated for you to run afterward.

"Creating databases manually" on page 556
IBM Business Process Manager Standard requires a Process Server database and Performance Data Warehouse database.. You must make sure the databases are available before you install IBM Business Process Manager Standard using a typical installation.

Creating the Performance Data Warehouse database tables:

Run the SQL scripts to create the database tables for your IBM Performance Data Warehouse.

You have performed all of the steps required for your configuration and have selected not to run the database scripts to initialize the databases during the configuration. You have also created the database in which you want to create the Performance Data Warehouse database tables. For information about creating the database, see *Creating databases*.

To run the Performance Data Warehouse SQL scripts that were produced when you completed your configuration follow the steps below:

Note: The Performance Data Warehouse does not support schema. You must login as the user and then execute the SQL file.

1. Go to the directory that holds the SQL scripts for creating the database tables.

   By default, SQL scripts are output to *profile_root*/dbscripts/PerformanceDW/*DB_product* at profile creation time. For example,
   profile_root/dbscripts/PerformanceDW/DB2

2. Run the following command as the root user or administrator.

   - **`Linux`** **`UNIX`** **configPerformanceDWDB.sh createDB**
   - **`Windows`** **configPerformanceDWDB.bat createDB**

Note: The **createDB** parameter is used to create a local database. The **createDB** parameter is not applicable for an Oracle database and must be excluded from the command.

3. For an SQL database, create the tables for the Performance Data Warehouse.

   From the directory navigated to in step 1, run **createTable_PerformanceDW.sql** on the database you configured for the Performance Data Warehouse (PDWDB).

   Make sure all SQL statements succeed without any errors.

You have created the database tables for your Performance Data Warehouse and have completed the database configuration.

You can now load the database with system information and start the servers or clusters on your Performance Data Warehouse.

Related tasks:

"Creating databases using the provided scripts" on page 555
If you did not create databases while creating profiles, database script files are generated for you to run afterward.

"Creating databases manually" on page 556
IBM Business Process Manager Standard requires a Process Server database and Performance Data Warehouse database.. You must make sure the databases are available before you install IBM Business Process Manager Standard using a typical installation.

Loading the database with system information

You must run the **bootstrapProcessServerData** command before you try to start or use the IBM Business Process Manager server.

When you run the **bootstrapProcessServerData** command, configuration data for the BPM applications is loaded into the database. This data is required for the BPM applications to run correctly

Note: This command runs automatically when you create a stand-alone profile and choose to create the database during profile creation.

- If you create a stand-alone profile and delay the database table setup, you must run the bootstrap command after the database and its tables have been created and before the server is started for the first time.

- In a network deployment (ND) environment, you must run this command after a server or cluster of servers is created. **For a cluster, you need to run the command only once**, specifying a single cluster member. Run this command after the database and its tables have been created, after the profile and deployment environment have been created, and before the first server is started. You do not need to rerun the command if you add another cluster member.

Run the bootstrap utility from the command line. The bootstrap utility is found in the stand-alone or deployment manager profile directory. For example:

```
<install_root>/profiles/Dmgr01/bin
C:\<install_root>\profiles\ProcCtr01/bin
```

Run the bootstrap utility using the following syntax:

- `Linux` `UNIX` **bootstrapProcessServerData.sh [-dbJDBCClasspath** *classpath*] **-clusterName** *cluster_name*

- `Linux` `UNIX` **bootstrapProcessServerData.sh [-dbJDBCClasspath** *classpath*] **-nodeName** *node_name* **-serverName** *server_name*

- `Windows` **bootstrapProcessServerData.bat [-dbJDBCClasspath** *classpath*] **-clusterName** *cluster_name*

- `Windows` **bootstrapProcessServerData.bat [-dbJDBCClasspath** *classpath*] **-nodeName** *node_name* **-serverName** *server_name*

where:

- **-dbJDBCClasspath** is the location of the database JDBC driver. The value specified for the **dbJBBCClasspath** must be surrounded by quotation marks if there are spaces in the path. The default location is *WAS_INSTALL_ROOT*/jdbcdrivers.
- **-clusterName** is the name of the application target cluster. For an ND environment, you must specify either this parameter or the **-nodeName** and **-serverName** parameters. Do not specify for stand-alone.
- **-nodeName** is the name of the node. For an ND environment, you must specify either this parameter and the **-serverName** parameter, or the **-clusterName** parameter. Do not specify for stand-alone.
- **-serverName** is the name of the server. For an ND environment, you must specify either this parameter and the **-nodeName** parameter, or the **-clusterName** parameter. Do not specify for stand-alone.

The parameters are case-sensitive.

You have loaded the database with system information prior to successfully starting the Process Server or Process Center.

Important: For SQL Server, if the bootstrap utility fails, check that when you created the Process Server and Performance Data Warehouse databases, the **COLLATE** attribute indicated case insensitive (**CI**), not case sensitive (**CS**). That is, the create command should look like this:

```
CREATE DATABASE database_name COLLATE SQL_Latin1_General_CP1_CI_AS
```

Failure to do so can result in an error like the following:

```
org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'message.routingCache'
defined in class path resource [registry.xml]: Instantiation of bean failed; nested exception is
org.springframework.beans.BeanInstantiationException: Could not instantiate bean class
[com.lombardisoftware.bpd.runtime.engine.message.DefaultMessageRoutingCache]: Constructor threw exception;
nested exception is org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCallback;
bad SQL grammar [select "value" from lsw_system where "key"=?]; nested exception is
com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.

Caused by: com.microsoft.sqlserver.jdbc.SQLServerException: Invalid object name 'lsw_system'.
at com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.getNextResult(SQLServerStatement.java:1454)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.doExecutePreparedStatement(SQLServerPreparedStatement.java:388)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement$PrepStmtExecCmd.doExecute(SQLServerPreparedStatement.java:338)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1416)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeCommand(SQLServerStatement.java:185)
at com.microsoft.sqlserver.jdbc.SQLServerStatement.executeStatement(SQLServerStatement.java:160)
at com.microsoft.sqlserver.jdbc.SQLServerPreparedStatement.executeQuery(SQLServerPreparedStatement.java:281)
at org.apache.commons.dbcp.DelegatingPreparedStatement.executeQuery(DelegatingPreparedStatement.java:205)
at org.springframework.jdbc.core.JdbcTemplate$1.doInPreparedStatement(JdbcTemplate.java:648)
at org.springframework.jdbc.core.JdbcTemplate.execute(JdbcTemplate.java:591)
[...]
```

Bootstrap data onto a server that is part of an ND environment but not part of a cluster:

```
bootstrapProcessServerData -nodeName node01 -serverName server1
```

Bootstrap data onto a cluster that hosts the Process Center or Process Server:

```
bootstrapProcessServerData -clusterName myAppCluster
```

Bootstrap data onto the default stand-alone profile server:

```
bootstrapProcessServerData
```

Bootstrap data onto the default stand-alone profile server with a custom JDBC driver location:

```
bootstrapProcessServerData -dbJDBCClasspath c:/DB2/java
```

You can start the stand-alone server or first cluster member.

Configuring components

You can configure individual components for IBM Business Process Manager.

Components can be configured in one of three ways:

1. Some IBM Business Process Manager components can be configured at profile creation time. This is particularly the case for a stand-alone server, for which most of the components may be configured at profile creation time.
2. For a network deployment cell, components may be configured by building and generating a Deployment Environment. By setting up a Deployment Environment, you build server clusters and configure the IBM Business Process Manager components on them.

   Note: Deployment Environments are not available for a stand-alone server.
3. The IBM Business Process Manager servers and clusters (and the components that run on them) can be configured individually, using either the administrative console or administrative scripts.

The information in this section describes how to configure individual components for IBM Business Process Manager.

Configuring a Process Center

You can use the administrative console to configure an application server or cluster as a Process Center. By configuring a server or cluster as a Process Center, you enable the server or cluster to support a shared development environment where Process Designer and Integration Designer can share assets such as process applications and their associated services.

This task describes how to use the administrative console to configure a server or cluster as a Process Center.

For information about configuring a server or cluster as a Process Center using wsadmin scripting, including parameter definitions and examples, see the configureProcessCenter command in the Reference documentation.

Note:

If you are using a stand-alone server profile, or have previously configured Process Center support for the server or cluster, you can review your current configuration but cannot change it in the administrative console.

1. Launch the administrative console
2. From the administrative console, select a path for configuring a server or a cluster.

   To configure a server, click **Servers** > **Server Types** > **WebSphere application servers** > *server_name* > **Process Center**.

   To configure a cluster, click **Servers** > **Clusters** > **WebSphere application server clusters** > *cluster_name* > **Process Center**
3. Set the parameters to configure the server or cluster as a Process Center.

   For a description of all parameters, go to the panel and select **Page help** or, see Process Center in *Administrative console help* section of the Reference documentation.
4. Save the configuration.

You have configured the server or cluster as a Process Center.

Configure any other Business Process Manager components and products that are part of your installation.

When configuring Business Process Manager components using the administrative console, you will need to configure Performance Data Warehouse after you have configured Process Center.

A `TeamWorksConfiguration.running.xml` file is generated from several local configuration XML files when each server starts. This file is located at `BPM_ROOT\profiles\profile_name\config\cells\cellName\nodes\nodeName\servers\serverName\process-center\TeamWorksConfiguration.running.xml`.

Check the contents of `TeamWorksConfiguration.running.xml`. You might need to edit the configuration files that are used to generate the `TeamWorksConfiguration.running.xml` file.

If changes are required for the URL, IP addresses, and ports, then update the `99Local.xml` and the `100Custom.xml` configuration files and restart the server(s). You must update these files when you add a web server to your cluster because information from these files are used for generating application URLs. Make sure that you make identical changes for these files on all affected nodes, the `99Local.xml` and `100Custom.xml` files are local to the systems on which they reside. They are not replicated automatically between nodes. For more information about updating the xml files, see the Managing configuration settings and Customizing the Process Server/Process Center cluster to work with a web server topics.

Configuring a Process Server

You can use the administrative console to configure an application server or cluster as a Process Server. By configuring a server or cluster as a Process Server, you enable the server or cluster to support process applications and their associated services.

This task describes how to use the administrative console to configure a server or cluster as a Process Server.

For information about configuring a server or cluster as a Process Server using wsadmin scripting, including parameter definitions and examples, see the configureProcessServer command in the Reference documentation.

Note:

If you are using a stand-alone server profile, or have previously configured Process Server support for the server or cluster, you can review your current configuration but cannot change it in the administrative console.

1. Launch the administrative console
2. From the administrative console, select a path for configuring a server or a cluster.

   To configure a server, click **Servers** > **Server Types** > **WebSphere application servers** > *server_name* > **Process Server**.

   To configure a cluster, click **Servers** > **Clusters** > **WebSphere application server clusters** > *cluster_name* > **Process Server**

3. Set the parameters to configure the server or cluster as a Process Server.

   For a description of all parameters, go to the panel and select **Page help** or, see Process Server in *Administrative console help* section of the Reference documentation.

4. Save the configuration.

You have configured the server or cluster as a Process Server.

Configure any other Business Process Manager components and products that are part of your installation.

When configuring Business Process Manager components using the administrative console, you will need to configure Performance Data Warehouse after you have configured Process Server.

A `TeamWorksConfiguration.running.xml` file is generated from several local configuration XML files when each server starts. This file is located at *BPM_ROOT*\profiles*profile_name*\config\cells*cellName*\nodes\ *nodeName*\servers*serverName*\process-center\TeamWorksConfiguration.running.xml.

Check the contents of `TeamWorksConfiguration.running.xml`. You might need to edit the configuration files that are used to generate the `TeamWorksConfiguration.running.xml` file.

If changes are required for the URL, IP addresses, and ports, then update the `99Local.xml` and the `100Custom.xml` configuration files and restart the server(s). You must update these files when you add a web server to your cluster because information from these files are used for generating application URLs. Make sure that you make identical changes for these files on all affected nodes, the `99Local.xml` and `100Custom.xml` files are local to the systems on which they reside. They are not replicated automatically between nodes. For more information about updating the xml files, see the Managing configuration settings and Customizing the Process Server/Process Center cluster to work with a web server topics.

Configuring the Business Performance Data Warehouse component on a server or cluster

You can use the administrative console to configure the Business Performance Data Warehouse component on a server or cluster.

Before you configure the Performance Data Warehouse component on a server or cluster, you must first configure the corresponding Process Server or Process Center. In a network deployment environment the two components may be on the same server or cluster or split across multiple servers or clusters.

This task describes how to use the administrative console to configure the Performance Data Warehouse component on a server or cluster.

For information about configuring the Performance Data Warehouse component on a server or cluster using wsadmin scripting, including parameter definitions and examples, see the configurePerfDW command in the Reference documentation.

Note:

If you are using a stand-alone server profile, or have previously configured Performance Data Warehouse support for the server or cluster, you can review your current configuration but cannot change it in the administrative console.

1. Launch the administrative console
2. From the administrative console, select a path for configuring a server or a cluster.

   To configure a server, click **Servers** > **Server Types** > **WebSphere application servers** > *server_name* > **Performance Data Warehouse Server**.

   To configure a cluster, click **Servers** > **Clusters** > **WebSphere application server clusters** > *cluster_name* > **Performance Data Warehouse Server**

3. Set the parameters to configure the Performance Data Warehouse component on a server or cluster.

   For a description of all parameters, go to the panel and select **Page help** or, see Performance Data Warehouse in *Administrative console help* section of the Reference documentation.

4. Save the configuration

You have configured the Performance Data Warehouse component on a server or cluster.

Configure any other Business Process Manager components and products that are part of your installation.

Configuring Common Event Infrastructure

You can configure Common Event Infrastructure resources, or change existing resources, using the server AdminTask object

Use the administrative console to configure CEI when you are installing it in a network deployment environment or in a cluster or in a stand-alone server configuration.

You can also use the wsadmin command to configure CEI, or you can use the command to alter an existing CEI configuration. In either case, you would change the configuration of CEI by using the server AdminTask object to run administrative commands.

After changing CEI configuration, you must restart the server or cluster.

Common Event Infrastructure components:

Common Event Infrastructure components are installed as a set of applications, services, and resources on the server.

When you configure Common Event Infrastructure, a number of components are created and deployed on your server.

Common Event Infrastructure service
A service installed into the server, that enables applications and clients to use Common Event Infrastructure. You can view the configuration of the Common Event Infrastructure service in the administrative console, as follows:

- For a server, click **Servers** > **Application Servers** > *server_name* > **Business Integration** > **Common Event Infrastructure** > **Common Event Infrastructure Service**.

- For a cluster, click **Servers** > **Clusters** > *cluster_name* > **Business Integration** > **Common Event Infrastructure** > **Common Event Infrastructure Service**.

If the check box labeled Enable the event infrastructure server is selected, then the service is installed and running or it will start after you restart your server or cluster. If it is cleared, then the service is not installed or will be uninstalled after you restart your server or cluster

Event service settings
A set of properties used by the event service that enable event distribution and persistence using the data store. Typically, no configuration is necessary for this resource, but you might need to create additional event service settings if you want to set up multiple event services in the same cell. To view the event service settings, click **Service integration** > **Event service** > **Event service settings**.

Event messaging configuration
The resources that support asynchronous event transmission to the event service using the Java Messaging Service (JMS). The default messaging configuration uses the server embedded messaging. You can optionally configure an external JMS provider for event messaging.

Event database
The event database is used to persistently store events received by the event service. You can configure an external event database on the following products: DB2, Oracle, and SQLServer.

Event filter plug-in
A filter plug-in is used to filter events at the source using XPath event selectors. To configure the filter properties, click **Service Integration** > **Common Event Infrastructure** > **Event Emitter Factories** > **Event Filter Settings**.

Emitter factory
An emitter factory is an object used by event sources to create emitters; an emitter is used to send events to the event service. The properties of an emitter factory affect the behavior of any emitter

that is created using that emitter factory. To view the available emitter factories, click **Service Integration** > **Common Event Infrastructure** > **Event Emitter Factories**.

Event service transmission

An event service transmission is an object defining properties that determine how emitters access the event service synchronously using EJB calls; these properties are used by emitter factories when creating new emitters. You can view or change the available event service transmissions from the emitter factory settings.

JMS transmission

A JMS transmission is an object that defines properties that determine how emitters access the event service asynchronously using a JMS queue; these properties are used by emitter factories when creating new emitters. You can view or change the available JMS transmissions from the emitter factory settings.

Event group

An event group is a logical collection of events used to categorize events according to their content. When querying events from the event service or subscribing to event distribution, an event consumer can specify an event group to retrieve only the events in that group. Event groups can also be used to specify which events should be stored in the persistent data store. To view the available event groups in the administrative console, click **Service integration** > **Common Event Infrastructure** > **Event service** > **Event services** > *event_service* > **Event groups**.

Configuring the Common Event Infrastructure server:

Configure Common Event Infrastructure by using the server administrative console.

Open the Common Event Infrastructure Server panel of the administrative console:

If you are configuring a server, select **Servers** > **Server Types** > **WebSphere application servers** > *server_name* > **Business Integration** > **Common Event Infrastructure** > **Common Event Infrastructure Server**.

If you are configuring a cluster, click **Servers** > **Clusters** > **WebSphere application server clusters** > *cluster_name* > **Business Integration** > **Common Event Infrastructure** > **Common Event Infrastructure Server**.

1. Enable the deployment of the Common Event Infrastructure enterprise application by selecting the check box labeled **Enable the event infrastructure server**. If the enable check box is cleared, Common Event Infrastructure has not been configured, or a previous configuration has been disabled but the server has not been restarted. An information message shows you whether this deployment target has Common Event Infrastructure configured.

   Note:
   - If you select the check box to enable the Common Event Infrastructure server and the server has not yet been configured, the parameters shown are used to configure it unless you change them.
   - The messaging service is created under a unique schema under the common database.
   - When the server or cluster on which Common Event Infrastructure has been configured is restarted, the changes take effect.

2. Optional: You can configure the event database only by using scripting. However, it is not recommended that you use an event database because of its negative impact on performance. The event database is used only by the Common Base Event browser to view events. Refer to "Configuring the event database."

3. Select whether the Common Event Infrastructure bus is to be **Local** on the server, or **Remote** and reside on another server. If you choose **Remote**, select the remote location from the menu or click **New** to create a new remote bus.

4. Configure Common Event Infrastructure support for messaging.

- Click **Edit** for a database configuration panel with a more extensive list of options than the ones listed on the panel.
- Use the fields on the panel to enter the information:

  a. **Database name** – enter the name of the database you use to store messages.

  b. **Schema** – enter a name for the schema, or accept the default name.

  c. **Username** and **Password** – these are for authenticating into the messaging database.

  d. **Server** – the name of the server where the messaging database is located.

  e. **Provider** – choose a provider for your database from the menu.

5. Create a messaging authentication alias for the Common Event Infrastructure bus.

   a. Select **Additional Properties** > **JMS Authentication Alias**.

   b. Enter the user ID and password you use for secure communications across the System Integration Bus. You can accept the default configured values of **CEI** for both the user ID and password if security is disabled. If security has been enabled, enter the user ID and password used for the bus authentication. In a production environment, select your own user ID and password to secure the system.

   c. Click **OK**.

6. Click **OK** or **Apply**.

7. Restart your server or cluster.

All the major parts of Common Event Infrastructure are now configured and running on your server or cluster. The configuration includes the event data store, the messaging engine, and the event application. This single panel can be used in place of many commands and steps you would otherwise use to configure Common Event Infrastructure.

After you have restarted your server or cluster, you will be able to store service component events that are emitted from your applications. You can now change the runtime properties of the Common Event Infrastructure server by selecting the **Common Event Infrastructure Destination** panel. You can choose whether to start the Common Event Infrastructure server at startup, and specify the emitter factory JNDI name where the events are sent.

Configuring event messaging:

You can modify the messaging configuration used for JMS transmission of events to the event service.

You will create the messaging infrastructure for Common Event Infrastructure when you use the administrative console panel to configure Common Event Infrastructure on a server. Generally, the messaging configuration will use the default messaging provider and create a single JMS queue for asynchronous transmission of events to the event service. You can, if necessary, modify this messaging configuration.

Configuring additional JMS queues:

If you are using the default event messaging configuration, you can add additional JMS queues for transmission of events to the event service.

To configure additional JMS queues to use the default messaging configuration, you can set up multiple JMS queues that are routed to the service integration bus queue destination. The Common Event Infrastructure service integration bus queue destination depends upon the scope at which the event service is deployed:

Scope	Service integration bus queue destination
Server	*node.server*.CommonEventInfrastructureQueueDestination

Scope	Service integration bus queue destination
Cluster	*cluster*.CommonEventInfrastructureQueueDestination

Configuring event messaging using an external JMS provider:

If you do not want to use the default embedded messaging configuration for event transmission, you can configure asynchronous message transport to use an external Java Messaging Service (JMS) provider.

Before you can configure event messaging using an external JMS provider, you must first create a JMS queue and connection factory using the appropriate interfaces for your JMS provider. You must also create a listener port or activation specification.

To configure event messaging using an external JMS provider:

From the wsadmin tool, run the **deployEventServiceMdb** administrative command in batch or interactive mode. The parameters of the **deployEventServiceMdb** command are as follows:

applicationName
    The application name of the event service message-driven bean to be deployed. This parameter is required.

nodeName
    The name of the node where the event service message-driven bean is to be deployed. If you specify a node name, you must also specify a server name. The node name is an optional parameter; the default value is the current node. Do not specify this parameter if you are deploying the application in a cluster.

serverName
    The name of the server where the event service message-driven bean is to be deployed. This parameter is required if you are deploying the application at server scope; otherwise it is optional. Do not specify a server name if you are deploying the application in a cluster.

clusterName
    The name of the cluster where the event service message-driven bean is to be deployed. Specify this parameter only if you are deploying the application in a cluster.

listenerPort
    The name of the listener port used by the event service message-driven bean to publish events. The specified listener port must exist. You must specify either a listener port or an activation specification, but not both.

activationSpec
    The JNDI name of the activation specification used by the event service message-driven bean to publish events. The specified activation specification must exist. You must specify either a listener port or an activation specification, but not both.

qcfJndiName
    The JNDI name of the JMS queue connection factory to be used by the event service message-driven bean. This parameter is required if you specify an activation specification; otherwise it is optional. If you specify a queue connection factory and a listener port, the queue connection factory must match the one configured for the listener port.

The **deployEventServiceMdb** administrative command deploys the message-driven bean for the event service, configured for the specified listener port or activation specification. It also creates an emitter factory and JMS transmission using the external JMS configuration. Applications can use either the default emitter factory (which is configured to use the default messaging configuration) or the new emitter factory (which uses the external JMS provider).

If you want to set up more than one JMS queue to the event service, you can run this command multiple times, specifying different enterprise application names and JMS queues. Each time you run the script, it deploys an additional message-driven bean and configures new resources to use the specified JMS queue.

Configuring the JMS authentication alias:

If WebSphere security is enabled and you want to use asynchronous JMS messaging to submit events to the event service, you must configure the JMS authentication alias.

To configure the JMS authentication alias:

From the wsadmin tool, run the **setEventServiceJmsAuthAlias** administrative command in batch or interactive mode. The parameters of the **setEventServiceJmsAuthAlias** command are as follows:

userName
> The name of the user to be used for the JMS authentication alias. This parameter is required.

password
> The password of the user to be used for the JMS authentication alias. This parameter is required.

nodeName
> The name of the node where you want to update or create the JMS authentication alias. If you specify a node name, you must also specify a server name. Do not specify a node name if you are configuring the authentication alias in a cluster.

serverName
> The name of the server where you want to update or create the JMS authentication alias. This parameter is required only if you specify a node; it is not valid if you are configuring the authentication alias in a cluster.

clusterName
> The name of the cluster where you want to update or create the JMS authentication alias. Specify this parameter only if you are configuring the authentication alias in a cluster; if you specify a cluster name, do not specify a node or server name.

The JMS authentication alias used by the event service objects is updated at the specified scope; if the authentication does not exist, it is created using the specified values.

Configuring the event database:

You can configure the event data source using commands that are specific for each supported database product.

The event database is required to support persistence of events. If you did not use the Common Event Infrastructure configuration panel in the administrative console, you still have the option of creating the event database by using the commands described here.

Event database limitations:

Some limitations apply to configurations of the event database using certain database software.

Refer to the following table to see which limitations might apply to your environment.

Table 105. Event database limitations

Database type	Limitations
Oracle	• The Oracle 11 JDBC thin driver imposes some size restrictions for string values if you are using a Unicode character set. You may receive an Oracle ORA-01461 error when events containing large values (such as a long message attribute) are stored in the event database. For more information about this restriction, refer to the Oracle 11 documentation. To avoid this problem, use the Oracle 11 OCI driver or the Oracle 11 thin driver. • Oracle database software treats a blank string as a NULL value. If you specify a blank string as an event attribute value, that string is converted to a NULL when it is stored in an Oracle event database.
SQL Server	• The SQL Server database must be configured to use mixed authentication mode. Trusted connections are not supported. • The XA stored procedures must be installed. These stored procedures are provided with the JDBC driver from Microsoft Corporation. • The sqljdbc.dll file must be available in a directory specified on the PATH statement. This file is provided with the JDBC driver from Microsoft Corporation. • The Distributed Transaction Coordinator (DTC) service must be started.

Configuring a DB2 event database (Linux, UNIX, and Windows systems):

You can configure an external event database using DB2 Universal Database on a Linux, UNIX, or Windows system.

To configure a DB2 event database on a Linux, UNIX, or Windows system:
1. Start the wsadmin tool.
2. Use the AdminTask object to run the **configEventServiceDB2DB** administrative command in batch or interactive mode. The minimum required parameters of the **configEventServiceDB2DB** command are as follows:

   createDB
   Indicates whether the administrative command creates and run the database configuration scripts. Specify **true** or **false**. If this parameter is set to **false**, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

   nodeName
   The name of the node that contains the server where the event service data source is created. If you specify a node name, you must also specify a server name. You must specify one of the following:
   • Node name and server name
   • Cluster name

   serverName
   The name of the server where the event service data source is created.

clusterName

The name of the cluster where the event service data source is created. If you specify a cluster name, do not specify node and server names.

jdbcClassPath

The path to the JDBC driver. Specify only the path to the driver file; do not specify the file name.

dbHostName

The host name of the server where the database is installed.

dbUser

The DB2 user ID to use when creating the event database. The specified user ID must have sufficient privileges to create and drop databases.

dbPassword

The DB2 password to use.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **configEventServiceDB2DB** administrative command.

The administrative command creates the required data source at the specified scope; if you specified **true** for the createDB parameter, the command also runs the generated database configuration script to create the database.

The generated database configuration scripts are stored by default in the *profile_root*/databases/event/ *node*/*server*/dbscripts/db2 directory. (In a Network Deployment environment, these scripts are stored under the deployment manager profile directory.) If you specified a value for the optional outputScriptDir parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

Configuring a DB2 database on a z/OS system:

You can configure an event database on a z/OS system using DB2 database software.

To configure the DB2 database from a remote client, you must have the DB2 Connect product installed with the latest fix packs.

To configure the event database:

1. `Linux` `UNIX` `Windows` If you are configuring the z/OS event database from a Linux, UNIX, or Windows client system, follow these steps to create and catalog the database:

   a. On the z/OS system, use the DB2 administration menu to create a subsystem.

   b. Optional: Create the storage group you want to use for the event database. You can also use an existing storage group (for example, **sysdeflt**).

   c. Enable the 4 K, 8 K, and 16 K buffer pools you want to use for the event database.

   d. Grant the necessary permissions to the user ID you want the data source to use. This user ID must have rights to access the database and storage group you created; it must also have permission to create new tables, table spaces, and indexes for the database.

   e. Catalog the remote database. Run the following commands, either in a script or in a DB2 command-line window:

   ```
   catalog tcpip node zosnode remote hostname server IP_port
       system db_subsystem
   catalog database db_name as db_name at node zosnode authentication DCS
   ```

   For more information about how to catalog a node and its databases, refer to the DB2 Connect documentation.

   f. Verify that you can establish a connection to the remote subsystem. You can run the following command to perform the verification:

   ```
   db2 connect to subsystem user userid using password
   ```

g. Bind to the host database. Run the following commands:

```
db2 connect to db_name user userid using password
db2 bind db2_root/bnd/@ddcsmvs.lst blocking all sqlerror continue message
    mvs.msg grant public
db2 connect reset
```

For more information about binding a client to a host database, refer to the DB2 Connect documentation.

2. On the WebSphere system, start the wsadmin tool.
3. Use the AdminTask object to run the **configEventServiceDB2ZOSDB** administrative command in batch or interactive mode. The minimum required parameters of the **configEventServiceDB2ZOSDB** command are as follows:

createDB

> `Linux` `UNIX` `Windows` Indicates whether the administrative command creates and runs the database configuration scripts. This parameter applies only if you are running the administrative command from a Linux, UNIX, or Windows client system. Specify **true** or **false**.
>
> If this parameter is set to **false**, or if you are running the command on the z/OS system, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

nodeName

> The name of the node that contains the server where the event service data source is created. If you specify a node name, you must also specify a server name. You must specify one of the following:
> - Node name and server name
> - Cluster name

serverName

> The name of the server where the event service data source is created.

clusterName

> The name of the cluster where the event service data source is created. If you specify a cluster name, do not specify node and server names.

jdbcClassPath

> The path to the JDBC driver. Specify only the path to the driver file; do not specify the file name.

dbHostName

> The host name of the server where the database is installed.

dbUser

> The DB2 user ID to use when creating the event database. The specified user ID must have sufficient privileges to create and drop databases.

dbPassword

> The DB2 password to use.

dbPort

> The DB2 instance port.

dbSubSystemName

> The name of the database subsystem.

storageGroup

> The storage group for the event database and the event catalog database.

eventDBName

> The event database name to be created.

eventCatalogDBName
> The event catalog database to be created.

bufferPool4K
> The name of the 4K buffer pool.

bufferPool8K
> The name of the 8K buffer pool.

bufferPool16K
> The name of the 16K buffer pool.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **configEventServiceDB2ZOSDB** administrative command.

The administrative command creates the required data source at the specified scope; if you are running the command on a Linux, UNIX, or Windows DB2 client and you specified **true** for the createDB parameter, the command also runs the generated database configuration script to create the database. On a z/OS system, you must use the SQL Processor Using File Input (SPUFI) facility to run the generated DDL files. The DDL files are stored in the *profile_root*/databases/event/*node*/*server*/db2zos/ddl directory.

The generated database configuration scripts are stored by default in the *profile_root*/databases/event/ *node*/*server*/dbscripts/db2zos directory. (In a Network Deployment environment, these scripts are stored under the deployment manager profile directory.) If you specified a value for the optional outputScriptDir parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

After you have finished configuring the database, you can use the server administrative console to test the database configuration. To perform this task, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Configuring an Oracle event database:

You can configure an external event database using Oracle Database on a Linux, UNIX, or Windows system.

Before you configure an Oracle event database, you must first create the database. The Oracle SID must already exist before you run the event database configuration command. The default SID for the event database is **event**.

To configure an Oracle event database:
1. Start the wsadmin tool.
2. Use the AdminTask object to run the **configEventServiceOracleDB** administrative command in batch or interactive mode. The minimum required parameters of the **configEventServiceOracleDB** command are as follows:

   createDB
   > Indicates whether the administrative command should create and run the database configuration scripts. Specify **true** or **false**. If this parameter is set to **false**, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

   nodeName
   > The name of the node that contains the server where the event service data source should be created. If you specify a node name, you must also specify a server name. You must specify one of the following:
   > - Node name and server name
   > - Cluster name

serverName

The name of the server where the event service data source should be created.

clusterName

The name of the cluster where the event service data source should be created. If you specify a cluster name, do not specify node and server names.

jdbcClassPath

The path to the JDBC driver. Specify only the path to the driver file; do not specify the file name.

oracleHome

The ORACLE_HOME directory. This parameter is required only if you specified **true** for the createDB parameter.

dbPassword

The password to use for the schema user ID created during the database configuration (the default user ID is **ceiuser**. This password is used to authenticate the Oracle database connection.

sysUser

The Oracle SYSUSER user ID. This user ID must have SYSDBA privileges.

sysPassword

The password for the specified SYSUSER user ID.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **configEventServiceOracleDB** administrative command.

The administrative command creates the required data source at the specified scope; if you specified **true** for the createDB parameter, the command also runs the generated database configuration script to create the database.

The generated database configuration scripts are stored by default in the *profile_root*/databases/event/ *node*/*server*/dbscripts/oracle directory. (In a Network Deployment environment, these scripts are stored under the deployment manager profile directory.) If you specified a value for the optional outputScriptDir parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

Configuring a SQL Server event database:

You can configure an external event database using Microsoft SQL Server Enterprise on a Windows system.

To configure a SQL Server event database:

1. On the SQL Server database server system, create the directory used to contain the database files. By default, the files are written to the c:\program files\ibm\event\ceiinst1\sqlserver_data directory. If you need to specify a different location, you must edit the generated database configuration script to modify the value of the ceiInstancePrefix parameter, and then run the script manually.
2. On the server system, start the wsadmin tool.
3. Use the AdminTask object to run the **configEventServiceSQLServerDB** administrative command in batch or interactive mode. The minimum required parameters of the **configEventServiceSQLServerDB** command are as follows:

createDB

Indicates whether the administrative command should create and run the database configuration scripts. Specify **true** or **false**. If this parameter is set to **false**, the scripts are created but are not run. You must then run the database configuration scripts to complete the database configuration.

nodeName

The name of the node that contains the server where the event service data source should be created. If you specify a node name, you must also specify a server name. You must specify one of the following:

- Node name and server name
- Cluster name

serverName

The name of the server where the event service data source should be created. If you specify a server name, you must also specify a node name.

clusterName

The name of the cluster where the event service data source should be created. If you specify a cluster name, do not specify node and server names.

dbServerName

The server name of the SQL Server database. This parameter is required only if you specified **true** for the createDB parameter.

dbHostName

The host name of the server where the SQL Server database is running.

dbPassword

The password to use for the user ID created to own the event database tables (the default user ID is ceiuser). The WebSphere data source uses this password to authenticate the SQL Server database connection.

saUser

A user ID with privileges to create and drop databases and users. This parameter is required only if you specified **true** for the createDB parameter.

saPassword

The password for the specified SA user.

Other parameters might be required for your environment. For a complete list of parameters and usage information, refer to the help for the **configEventServiceSQLServerDB** administrative command.

The administrative command creates the required data source at the specified scope; if you specified **true** for the createDB parameter, the command also runs the generated database configuration script to create the database.

The generated database configuration scripts are stored by default in the *profile_root*/databases/event/ *node*/*server*/dbscripts/dbscripts/sqlserver directory. (In a Network Deployment environment, these scripts are stored under the deployment manager profile directory.) If you specified a value for the optional outputScriptDir parameter, the scripts are stored in that location instead. You can use these scripts to manually configure the event database at any time.

Manually running database configuration scripts:

You can manually run the scripts generated by the database configuration administrative commands at any time.

Database configuration is a two-step process. The database configuration administrative command first generates a database-specific script for your environment; this generated script then configures the event database and data sources. If you specify **true** for the createDB parameter when running the administrative command, both steps happen automatically.

However, if you specify **false** for the createDB parameter, you must complete the database configuration by manually running the generated script on the target system. You might need to run the script manually in any of the following situations:

- You need to configure the event database on a different system from the one where you ran the administrative command.
- You need to re-create the event database at a later time.
- You need to modify the default options used by the generated script before running it.

Manually creating a DB2 event database on a Linux, UNIX, or Windows system:

Manually generate a database configuration script for a DB2 event database on a Linux, UNIX, or Windows server using the **cr_event_db2** command.

Manually run the generated database configuration script for a DB2 event database on a Linux, UNIX, or Windows system by completing the following steps:

1. On the server system, go to the directory containing the generated script. The default location is the *profile_root/databases/event/node/server*/dbscripts/db2 directory. If you specified a value for the **outputScriptDir** parameter of the database configuration administrative command, the scripts are stored in that location instead.
2. Using an ASCII text editor, make any required modifications to the configuration script. The name of the script varies depending upon the operating system in use:

   - Windows **cr_event_db2.bat**
   - Linux UNIX **cr_event_db2.sh**
3. Run the database creation script using the following syntax (remember to specify the file extension, if applicable):

   ```
   cr_event_db2 [client|server] db_user [db_password]
   ```

   The parameters are as follows:

   client|server
   Indicates whether the database is a client or server. You must specify either **client** or **server**.

   db_user
   The database user ID. This parameter is required.

   db_password
   The password for the database user. If you do not specify a password for a client database, you are prompted for it.

   For example, the following command would create the DB2 event database for a client database, using the user ID **db2admin** and the password **mypassword**:

   ```
   cr_event_db2 client db2admin mypassword
   ```
4. Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Manually creating a DB2 event database on a z/OS system:

Manually generate a database configuration script for a DB2 event database on a z/OS system using a Linux, UNIX, or Windows client system using the **cr_event_db2zos** command.

Manually run the generated database configuration script for a DB2 event database on a z/OS system using a Linux, UNIX, or Windows client system by completing the following steps:

1. On the server system, go to the directory containing the generated script. The default location is the *profile_root*/databases/event/node/server/dbscripts/db2 directory. If you specified a value for the **outputScriptDir** parameter of the database configuration administrative command, the scripts are stored in that location instead.

2. Using an ASCII text editor, make any required modifications to the configuration script. The name of the script varies depending upon the operating system in use:

   - ▆ Windows ▆ **cr_event_db2zos.bat**

   - ▆ Linux ▆ ▆ UNIX ▆ **cr_event_db2zos.sh**

3. Run the database creation script using the following syntax (remember to specify the file extension, if applicable):

   cr_event_db2zos [dbName=*db_name*] *db_user* [*db_password*]

   The parameters are as follows:

   db_name

   > The database name to use. This parameter is optional; if you do not specify a database name, a name is generated.

   db_user

   > The database user ID to use. This parameter is required.

   db_password

   > The password for the database user. If you do not specify the password, the DB2 database prompts you for it.

   For example, the following command would create a DB2 event database called **event**, using the user ID **db2admin** and the password **mypassword**:

   cr_event_db2zos dbName=client db2admin mypassword

4. Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Manually creating an Oracle event database:

Manually generate a database configuration script for an Oracle event database using the **cr_event_oracle** command.

Manually run the generated database configuration script for an Oracle event database by completing the following steps:

1. On the server system, go to the directory containing the generated script. The default location is the *profile_root*/databases/event/node/server/dbscripts/db2 directory. If you specified a value for the **outputScriptDir** parameter of the database configuration administrative command, the scripts are stored in that location instead.

2. Using an ASCII text editor, make any required modifications to the configuration script. The name of the script varies depending upon the operating system in use:

   - ▆ Windows ▆ **cr_event_oracle.bat**

   - ▆ Linux ▆ ▆ UNIX ▆ **cr_event_oracle.sh**

3. Run the database creation script using the following syntax (remember to specify the file extension, if applicable):

```
cr_event_oracle password sys_user
  sys_password [sid=sid]
  [oracleHome=oracle_home]
```

The parameters are as follows:

password
> The password for the schema user ID. This parameter is required.

sys_user
> The user ID that has SYSDBA privileges in the Oracle database (typically the sys user). This parameter is required.

sys_password
> The password for the specified sys user ID. If this user ID does not use a password, type **none**.

sid=*sid*
> The Oracle system identifier (SID). This parameter is optional.

oracleHome=*oracle_home*
> The Oracle home directory. This parameter is optional; if you do not specify a value, a generated path is used.

For example, the following command would create the Oracle event database using the schema user ID **auser** and the sys user ID **sys**:

```
cr_event_oracle auser sys syspassword sid=event oracleHome=c:\oracle
```

4. Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Manually creating a SQL Server event database:

Manually generate a database configuration script for a SQL Server event database using the **cr_event_mssql** command.

Manually run the generated database configuration script for a SQL Server event database by completing the following steps:

1. On the server system, go to the directory containing the generated script. The default location is the *profile_root/databases/event/node/server*/dbscripts/db2 directory. If you specified a value for the **outputScriptDir** parameter of the database configuration administrative command, the scripts are stored in that location instead.
2. Using an ASCII text editor, make any required modifications to the cr_event_mssql.bat script.
3. Run the database creation script using the following syntax:
```
cr_event_mssql user_id password [server=server] sauser=sa_user
  sapassword=sa_password
```

The parameters are as follows:

user_id
> The SQL Server login user ID that will own the created tables. This user ID must be created in SQL Server so that a JDBC connection can be made to the database. (The JDBC drivers do not support trusted connections.)

password
> The password for the new login user ID that is created.

server=_server_
> The name of the server that contains the SQL Server database. This parameter is optional; the default value is the local host.

sauser=_sa_user_
> The sa user ID. This user ID must have sufficient privileges to create databases and user logins.

sapassword=_sa_password_
> The sa password, if using mixed authentication mode. If the sa user ID does not have a password set, specify **sapassword=** with no value. Omit this parameter if you are using a trusted connection.

For example, the following command would create the SQL Server event database using the login user ID **userid**:

```
cr_event_mssql userid apassword server=myserver sauser=sa sapassword=sapassword
```

4. Restart the server. For a federated node, you must also stop and restart the node agent using the **stopNode** and **startNode** commands.

After you finish configuring the database, you can use the administrative console to test the database configuration. To do this, navigate to the appropriate JDBC data source and select the **Test Connection** option.

Upgrading the event database from a previous version:

If you have migrated from a previous version of Common Event Infrastructure and you are using event persistence, you might need to upgrade an existing event database.

Upgrading the event database is required if you are migrating from Common Event Infrastructure version 5.1 or earlier.

The database upgrade process upgrades the schema and metadata of the existing event database to the current version while preserving existing event data.

The database upgrade script upgrades the schema and metadata of the existing event database to the current version.

Unsupported versions: If your event database uses a version of database software that is no longer supported by Common Event Infrastructure 6.0, you must first migrate the database to a supported version using the appropriate procedure for the database software. You can then follow the event database upgrade process to upgrade the database.

Upgrading a DB2 event database from a previous version:

If you have an existing DB2 event database from Version 5.1 of Common Event Infrastructure on a Linux, UNIX, or Windows system, you must upgrade it to the current version.

To upgrade a DB2 event database on a Linux or UNIX system:
1. Make a backup copy of the existing event database.
2. Go to the *profile_root*/bin directory.
3. Run the DB2 upgrade script for your operating system:

   - | Windows | Windows systems:

     ```
     eventUpgradeDB2 runUpgrade=[true|false] dbUser=user
        [dbName=name] [dbPassword=pw]
        [dbNode=node] [scriptDir=dir]
     ```

   - | Linux | | UNIX | Linux and UNIX systems:

```
eventUpgradeDB2.sh runUpgrade=[true|false] dbUser=user
  [dbName=name] [dbPassword=pw]
  [dbNode=node] [scriptDir=dir]
```

The typical required parameters are as follows:

runUpgrade
> Indicates whether you want the upgrade script to automatically run the generated DDL scripts to complete the database upgrade. This parameter is required. Specify **false** if you want to manually perform the database upgrade at a later time or on a different system.

dbUser
> Specifies the DB2 user ID to use. This parameter is required.

dbName
> Specifies the DB2 database name. The default name for the event database is **event**. This parameter is required if you specified **runUpgrade=true**.

dbPassword
> Specifies the password for the specified DB2 user ID. This parameter is optional; if you do not specify a password, DB2 prompts you to type it.

dbNode
> Specifies the database node name. This parameter is required if you are running the upgrade script from a DB2 client system.

scriptDir
> Specifies the directory you want to contain the generated DDL scripts. This parameter is optional; if you do not specify a directory, the scripts are stored in the .\eventDBUpgrade\db2 directory.

To see a complete list of parameters and usage information, run the **eventUpgradeDB2** script with no parameters.

The upgrade script generates the required DDL scripts for upgrading the event database. If you specified **runUpgrade=true**, the DDL scripts are automatically run, completing the upgrade.

The following example upgrades an existing DB2 database on a Windows system:
```
eventUpgradeDB2 runUpgrade=true dbUser=db2inst1 dbName=event
```

If you specified **runUpgrade=false**, you must manually run the DDL scripts on the database system to complete the database upgrade.

Upgrading a DB2 for z/OS event database from a previous version:

If you have an existing DB2 event database from Version 5.1 of Common Event Infrastructure on a z/OS system, you must upgrade it to the current version.

To upgrade a DB2 event database on a z/OS system:
1. Make a backup copy of the existing event database.
2. Go to the *profile_root*/bin directory.
3. Run the DB2 for z/OS upgrade script for your client operating system:
   - Windows Windows systems:

     ```
     eventUpgradeDB2ZOS runUpgrade=[true|false] dbUser=user
       [dbName=name] [dbPassword=pw]
       [scriptDir=dir] storageGroup=group
       bufferPool4K=4kbufpool bufferPool8k=8kbufpool
       bufferPool16K=16kbufpool
     ```
   - Linux UNIX Linux and UNIX systems:

```
eventUpgradeDB2ZOS.sh runUpgrade=[true|false] dbUser=user
  [dbName=name] [dbPassword=pw]
  [scriptDir=dir] storageGroup=group
  bufferPool4K=4kbufpool bufferPool8k=8kbufpool
  bufferPool16K=16kbufpool
```

The typical required parameters are as follows:

runUpgrade

Indicates whether you want the upgrade script to automatically run the generated DDL scripts to complete the database upgrade. This parameter is required. Specify false if you want to manually upgrade the database at a later time or on a different system.

z/OS systems: This parameter is ignored on a native z/OS system. Automatically running the generated DDL scripts is supported only on a client system.

dbUser

Specifies the DB2 user ID to use. This parameter is required.

dbName

Specifies the DB2 database name. The default name for the event database is **event**. This parameter is required if you specified **runUpgrade=true**.

dbPassword

Specifies the password for the specified DB2 user ID. This parameter is optional; if you do not specify a password, DB2 prompts you to type it.

scriptDir

Specifies the directory you want to contain the generated DDL scripts. This parameter is optional; if you do not specify a directory, the scripts are stored in the .\eventDBUpgrade\db2zos directory.

storageGroup

Specifies the name of the storage group. This parameter is required.

bufferPool4K

Specifies the name of the 4K buffer pool. This parameter is required.

bufferPool8K

Specifies the name of the 8K buffer pool. This parameter is required.

bufferPool16K

Specifies the name of the 16K buffer pool. This parameter is required.

To see a complete list of parameters and usage information, run the **eventUpgradeDB2ZOS** script with no parameters.

The upgrade script generates the required DDL scripts for upgrading the event database. If you specified **runUpgrade=true** on a client system, the DDL scripts are automatically run, completing the upgrade.

The following example upgrades a DB2 for z/OS event database from a Windows client system:

```
eventUpgradeDB2ZOS runUpgrade=true dbUser=db2inst1 dbName=event
  storageGroup=sysdeflt bufferPool4K=BP9 bufferPool8K=BP8K9 bufferPool16K=BP16K9
```

If you specified **runUpgrade=false**, or if you ran the upgrade script on the z/OS system, you must manually run the generated DDL scripts on the z/OS system using the SQL Processor Using File Input (SPUFI) facility. This step completes the database upgrade.

Upgrading an Oracle event database from Version 5:

If you have an existing Oracle event database from Version 5.1 of Common Event Infrastructure, you must upgrade it to the current version.

To upgrade an Oracle event database:

1. Make a backup copy of the existing event database.
2. Go to the *profile_root*/bin directory.
3. Run the Oracle upgrade script for your operating system:
   - Windows systems:
     ```
     eventUpgradeOracle runUpgrade=[true|false] schemaUser=schemauser
       [oracleHome=dir] [dbName=name]
       [dbUser=sysuser] [dbPassword=pw]
       [scriptDir=dir]
     ```
   - Linux and UNIX systems:
     ```
     eventUpgradeOracle.sh runUpgrade=[true|false] schemaUser=schemauser
       [oracleHome=dir] [dbName=name]
       [dbUser=sysuser] [dbPassword=pw]
       [scriptDir=dir]
     ```

   The typical required parameters are as follows:

   runUpgrade
   > Indicates whether you want the upgrade script to automatically run the generated DDL scripts to complete the database upgrade. This parameter is required. Specify **false** if you want to manually upgrade the database at a later time or on a different system.

   schemaUser
   > Specifies the Oracle user ID that owns the database tables. This parameter is required.

   oracleHome
   > Specifies the Oracle home directory. This parameter is required if you specified **runUpgrade=true**.

   dbName
   > Specifies the Oracle database name. The default name for the event database is **event**. This parameter is required if you specified **runUpgrade=true**.

   dbUser
   > Specifies the Oracle sys user ID. This parameter is required if you specified **runUpgrade=true**.

   dbPassword
   > Specifies the password for the sys user ID. Do not specify this parameter if the sys user ID has no password.

   scriptDir
   > Specifies the directory you want to contain the generated DDL scripts. This parameter is optional; if you do not specify a directory, the scripts are stored in the .\eventDBUpgrade\oracle directory.

   To see a complete list of parameters and usage information, run the **eventUpgradeOracle** script with no parameters.

The upgrade script generates the required DDL scripts for upgrading the event database. If you specified **runUpgrade=true**, the DDL scripts are automatically run, completing the upgrade.

The following example upgrades an existing Oracle database on a Windows system:
```
eventUpgradeOracle runUpgrade=true schemaUser=cei
  dbName=event dbUser=sys
```

If you specified **runUpgrade=false**, you must manually run the DDL scripts on the database system to complete the database upgrade.

Configuring the SMTP server
Define the Simple Mail Transfer Protocol (SMTP) server that the Process Server runtime environment should use to send e-mail notifications. After a new install, you must manually configure the SMTP server to enable mail notifications.

This topic provides the steps that are necessary to change the SMTP server settings to point to the right SMTP server. The default SMTP server settings are:

```
<server merge="mergeChildren">

        <!-- email properties -->
        <email>
            <!-- SMTP server that mail should be sent to -->
            <smtp-server>smtp.yourcompany.com</smtp-server>

        </email>
```

1. Stop the server(s).
2. Open the 100Custom.xml file using a text editor.
   - For a network deployment environment:
     - Process Server and Process Center clusters:

       DMGR-profile-root\config\cells*cell-name*\clusters*ClusterName*\process-center\config\100Custom.xml
     - Process Server and Process Center cluster members:

       DMGR-profile-root\config\cells*cell-name*\nodes*node-name*\servers*serverName*\process-center\config\100Custom.xml
   - For a standalone environment:

     profile-root\config\cells*cell-name*\nodes*node-name*\servers*serverName*\process-center\config\100Custom.xml
3. Add the following to the 100Custom.xml file:

```
<properties>
  <server merge="mergeChildren">

  <!-- email properties -->
  <email merge="mergeChildren">
   <!-- SMTP server that mail should be sent to -->
   <smtp-server merge="replace">SMTP.YOURCOMPANY.COM</smtp-server>
  </email>
  </server>
 </properties>
```

   Where *SMTP.YOURCOMPANY.COM* is the name of the actual SMTP server.
4. Save your changes and restart the server(s).

Advanced configuration: customizing your environment

After you install and configure the runtime environment on your system in IBM Business Process Manager, you might want to customize your configuration. For example, you might want to customize settings, set up an additional security provider, set up user accounts, and change and encrypt passwords.

Customizing the settings used by Process Server to connect to Process Center

After you install and configure IBM Business Process Manager, you can change the Process Server from an offline server to a Process Center connected server, and vice versa.

Make sure that you have installed and configured the Process Server component of IBM Business Process Manager.

Use the procedure in this topic to customize the Process Server connection settings for either a stand-alone server environment or network deployment environment.

Typically, you manage configuration customization by editing the 100Custom.xml file. (If you must change the configuration, for example to change the address of the Process Center, update the Authentication Alias Configuration and run the administrative task.) For an example of customizing and editing the 100Custom.xml file, see "Changing passwords after installation" on page 648.

For example, you can perform the following tasks:

- Update the host and port name of the connection details.
- Change Process Server from an offline server to a Process Center connected server, and vice versa. For example, if you configured Process Server as an offline server, you can change the configuration to a Process Center connected server. You can administer and manage a Process Center connected Process Server directly from the Process Center administrative console. For information about administering a Process Server through the Process Center user interface, see Administering IBM Business Process Manager.
- Change the connection URL.

If your Process Center is connected to a Process Server, you can manage it directly through the Process Center user interface.

If there are multiple application targets in your network configuration and you want to configure new users for each target for process center-process server communication, you must create an authentication alias for each new user that doesn't already exist in the configuration file.

To customize the settings used by Process Server to connect to a Process Center, complete the following steps:

1. Stop the deployment manager and the Process Server cluster or server if they are running.
2. Update the configuration.
    - **If you are running V7.5.1.1:** Update the Authentication Alias Configuration page and run the administrative task using the steps provided here.
        a. Open the administrative console.
        b. Click **Security** > **Business Integration Security**.
        c. Click the authentication alias that you want to configure, or select the check box associated with the authentication alias and click the **Edit** button. For each default IBM Business Process Manager user account, default authentication aliases are provided for external components to connect to the Process Server. For example, BPMPrimaryAdmin_Auth_Alias is the authentication alias for the default administrative user. Provide user name and password credentials.

            Note: You must be a member of the administrator or configuration role in order to make changes to the authentication alias configurations.
        d. Update the values provided here and run the administrative task.
            - **repositoryServerURL**: `http://localhost:9190/ProcessCenter`
            - **repositoryServerHeartbeat**: `10`

                Note: An offline server is indicated with **-1** as a value.
                For a server:

                ```
                AdminTask updateBPMConfig {-nodeName Node_Name -serverName Server_Name -repositoryServerURL http://localhost
                AdminConfig save
                ```

                For a cluster:

                ```
                AdminTask updateBPMConfig {-clusterName Cluster_Name -repositoryServerURL http://localhost:9190/ProcessCente
                AdminConfig save
                ```

                Note: You can also update the **user-auth-alias** and **designated-user-auth-alias** values.

                The **user-auth-alias** value specifies the authentication alias for a user to connect from the process server to the Process Center. The user and password set in this alias must be present in the Process Center.

                The **designated-user-auth-alias** value specifies the authentication alias for a user to access and deploy snapshots to the runtime process server and access that process server from the

Process Inspector, which is located in IBM Process Designer. This authentication alias must be defined in both that process server and the Process Center and the passwords must match.

The **updateBPMconfig** admin task updates the XML configuration files including the 100Custom.xml file to provide a single file that has all of the changes applied to the environment. The admin task creates the 100Custom.xml file if it does not exist. If you have multiple versions of the custom file, such as 101Custom.xml, 102Custom.xml, and so on, then only the 100Custom.xml file is updated.

- **If you are running V7.5.1:** Edit the connection values in the 99Local.xml file.

The directory path location of the 99Local.xml file depends on whether your environment is stand-alone, clustered network deployment, or single server network deployment.

For a stand-alone server environment

> *stand-alone-profile-root*\config\cells*cell-name*\nodes*stand-alone-node-name*\
> servers*server_name*\process-server\config\system\99Local.xml

For a network deployment cluster environment

> – In the network deployment Process Server cluster: *DMGR-profile-root*\config\cells\
>   *cell-name*\clusters*cluster-name*\process-server\config\system\99Local.xml
> – Each Process Server cluster member at: *DMGR-profile-root*\config\cells*cell-name*\
>   nodes*custom-node-name*\servers*cluster-member-name*\process-server\config\system\
>   99Local.xml

For a network deployment single server environment

> *DMGR-profile-root*\config\cells*cell-name*\nodes*custom-node-name*\servers\
> *server_name*\process-server\config\system\99Local.xml

See the following example:

Before the update:

> <repository-server-url>/ProcessCenter</repository-server-url>
> <repository-server-interval>-1</repository-server-interval>

After the update:

> <repository-server-url>http://localhost:9190/ProcessCenter</repository-server-url>
> <repository-server-interval>10</repository-server-interval>

Important: An offline server is indicated with **-1** as a value, for example, **<repository-server-interval>-1</repository-server-interval>**.

3. Restart the deployment manager.
4. If you have a network deployment environment, synchronize the node agents from the administrative console. Deployment manager and node agent synchronization are not needed for a stand-alone server. For information on how to synchronize node agents, see the **syncNode** command in the WebSphere Application Server information center.
5. Restart the Process Server cluster or server.

The settings used to connect Process Server to Process Center have been customized.

Related tasks:

"Changing passwords after installation" on page 648
To change the passwords that you specified when you installed IBM Business Process Manager, use the Process Admin Console to change the password and the WebSphere Application Server administrative console to change the password in the corresponding authentication alias.

Related information:

WebSphere Application Server information center

Customizing the Process Server/Process Center cluster to work with a web server

After you install and configure IBM Business Process Manager, you must customize your environment so that it can function properly as a cluster with a web server.

Make sure that the Process Server has been installed and configured, and that you have created and configured the servers and clusters. You cannot start the cluster until you complete the procedures in this topic.

Note: The **updateBPMconfig** admin task was added in V7.5.1.1. This admin task updates the XML configuration files including the 100Custom.xml file to provide a single file that has all of the changes applied to the environment. The admin task creates the 100Custom.xml file if it does not exist. If you have multiple versions of the custom file, such as 101Custom.xml, 102Custom.xml, and so on, then only the 100Custom.xml file is updated.

The procedures in this topic describe how to:

- Configure the 100Custom.xml file in both the Process Server and Process Center to point to a web server. In network deployment environments, this file exists multiple times. For more information on how to create or find the 100Custom.xml file, see Managing IBM Business Process Manager configuration settings. Although this topic talks about 100Custom.xml, other 100-level files can contain configuration information. If you must change the configuration, for example to change the address of the Process Center, update the authentication alias configuration and run the administrative task. Each authentication alias in the 100Custom.xml file must contain a valid user name and password.
- Change the Process Server from offline mode to online mode for use with a web server.
- Set the Process Center to Process Server communication to work with HTTP over SSL or HTTP Secure (HTTPS).

1. Configure the 100Custom.xml file in both the Process Server and the Process Center to point to a web server. Before you directly edit the 100Custom.xml configuration file, create a backup file for it. Add a .bak extension to keep the file safe.

   a. Stop the deployment manager.

   b. Open the 99Local.xml file and locate the <authoring-environment> section.

   c. Copy and paste that section into the 100Custom.xml file.

   d. Update the text that you pasted into the 100Custom.xml file with a merge="mergeChildren" qualifier. If there is no web server, then all URLs in the 100Custom.xml files point to the cluster member host name and port. If there is no web server, the cluster server member xml files do not need to be changed. If you have a web server, change all values to point to the web server host name and port with the actual host name and port of the web server. If you are using the default port for http (80) or https (443), the port is an optional part of the URL. Ensure that you do not leave any tokenized value in the 100Custom.xml file. Here is an example of an entry after changes have been made:

   ```
   <authoring-environment merge="mergeChildren">
     <!--  Prefix for serving images in the Authoring Environment  -->
     <images-prefix merge="replace">https://server1.ibm.com:9447/teamworks</images-prefix>

     <!-- Prefix for URLs that refer to the portal -->
     <portal-prefix merge="replace">https://server1.ibm.com:9447/portal</portal-prefix>

     <!-- Prefix for URLs that refer to the repository view -->
     <repository-prefix merge="replace">https://server1.ibm.com:9447/ProcessCenter</repository-prefix>

     <servlet-prefix merge="replace">https://server1.ibm.com:9447/teamworks</servlet-prefix>

     <!-- Prefix for URLs that refer to the web api -->
     <webapi-prefix merge="replace">https://server1.ibm.com:9447/webapi></webapi-prefix>
   </authoring-environment>
   ```

   e. If you do not plan to continue with the rest of this procedure, start the deployment manager. Otherwise, continue to the next step.

f. Save the updates. In the administrative console, click **Full Synchronize** to synchronize the information to all the nodes.

g. Restart the server.

2. To configure the Process Server and the Process Center to point to a web server:

a. Stop the deployment manager.

b. Update the configuration.

- **If you are running V7.5.1.1 and later:** Run the administrative task.

1) Update the value provided here and run the administrative task. For a server:

```
AdminTask.updateBPMConfig( [ "-nodeName", Node_Name, "-serverName", Server_Name, "-appServerName", APPSERVER_U
AdminConfig.save()
```

For a cluster:

```
AdminTask.updateBPMConfig( [ "-clusterName", Cluster_Name, "-appServerName", YOUR_APPSERVER_URL,  "-appServerP
AdminConfig.save()
```

Note: If you have a web server, change all values to point to the web server host name and port with the actual host name and port of the web server. If you are using the default port for http (80), port is optional.

- **If you are running V7.5.1.0:** Edit the connection values in the `100Local.xml` file located on the Process Server cluster machine.

    – In the network deployment Process Server cluster: `DMGR-profile-root\config\cells\cell-name\clusters\ClusterName\process-center\config\system\100Local.xml`

    – Each Process Server cluster member at: `DMGR-profile-root\config\cells\cell-name\nodes\node-name\servers\serverName\process-center\config\system\100Local.xml`

See the following example:

```
<images-prefix>http://w2k8-64bitp.austin.ibm.com:9080/teamworks</images-prefix>
<portal-prefix>http://w2k8-64bitp.austin.ibm.com:9080/portal</portal-prefix>
<deploy-snapshot-using-https>true</deploy-snapshot-using-https>
 --><server-name>Process Server</server-name>
<server-description>Process Server Production Cluster</server-description>
<server-host>w2k8-64bitps1.austin.ibm.com</server-host>
<server-port>80</server-port>
<environment-type>Production</environment-type>
```

In the previous example:

```
<images-prefix>http://w2k8-64bitp.austin.ibm.com:9080/teamworks</images-prefix>
```

- is the prefix for serving images in the authoring environment.

```
<portal-prefix>http://w2k8-64bitp.austin.ibm.com:9080/portal</portal-prefix>
```

- is the prefix for URLs that refer to Process Portal.

```
<server-host>w2k8-64bitps1.austin.ibm.com</server-host>
```

- is the host name of the Process Server cluster web server.

```
<server-port>80</server-port>
```

- is the Process Server web server http port. If you are using https, specify the https port.

c. If you do not plan to continue with the rest of this procedure, start the deployment manager. Otherwise, continue to the next step.

d. Synchronize the node agents from administrative console. For more information, see the **syncNode** command in the WebSphere Application Server information center.

3. To change the Process Server from offline mode to online mode for use with a web server:

a. Stop the deployment manager.

b. On the Process Server cluster machine, verify that the authentication alias have valid user names and passwords, and run the administrative task.

   1) Open the administrative console.

   2) Click **Security** > **Global Security**, then expand **Java Authentication and Authorization Service** and click **J2C authentication data**.

   3) Verify the following authentication aliases and change them if needed:

     • The alias that is configured in the `100custom.xml` file under **<repository-server-user-auth-alias>BPMRuntimeServer_Auth_Alias</repository-server-user-auth-alias>** must contain a valid user name and password for authenticating to Process Center.

     • The alias that is configured in `100custom.xml` file under **<repository-server-designated-user-auth-alias>BPMAuthor_Auth_Alias</repository-server-designated-user-auth-alias>** must contain a valid user name and password in IBM Process Server, which is used by IBM Process Center for online deployment.

    Note: You must be a member of the administrator or configuration role in order to make changes to the authentication alias configurations.

   4) Update the values provided here and run the administrative task. For a cluster:

     • **repositoryServerURL**: `http://`*`YOUR_APPSERVER_URL`*`/ProcessCenter`

     • **repositoryServerHeartbeat**: `10`

     • **clusterName**: *`YOUR_APPSERVER_URL`*

    ```
AdminTask.updateBPMConfig( [ "-clusterName", Cluster_Name, "-repositoryServerURL", " http://APPSERVER_URL/Prc
AdminConfig.save()
```

c. If you do not plan to continue with the rest of this procedure, start the deployment manager. Otherwise, continue to the next step.

d. Synchronize the node agents from the administrative console.

  For information on how to synchronize node agents, see **syncNode** command in the WebSphere Application Server information center.

4. To set the Process Center to Process Server communication to work with HTTP over SSL or HTTP Secure (HTTPS).

  a. Import the Process Server SSL certificate into Process Center:

   1) On the Process Center administrative console, click **Security** > **SSL certificate and key management** > **Manage endpoint security configurations** > *Click on any node* > **Key stores and certificates** > **CellDefaultTrustStore** > **Signer certificates** > **Retrieve from port**.

   2) Enter an alias, the host name, and secure port of the Process Server web server. The default IHS web server secure port is 443. If there is no web server, the host name and port values are the same as the host name and port values of the Process Center cluster member.

   3) Click **Retrieve signer information**, click **Apply** and **Save**.

   4) Restart the cluster.

  b. Import the Process Center SSL certificate into Process Server:

   1) On the Process Server administrative console, go to **Security** > **SSL certificate and key management** > **Manage endpoint security configurations** > *Click on any node* > **Key stores and certificates** > **CellDefaultTrustStore** > **Signer certificates** > **Retrieve from port**.

   2) Enter an alias, the host name, and secure port of the Process Server web server. The default IHS web server secure port is 443. If there is no web server, the host name and port values are the same as the host name and port values of the Process Center cluster member.

   3) Click **Retrieve signer information**, click **Apply** and **Save**.

   4) Restart the cluster.

  c. Update the value provided here and run the administrative task. For a cluster:

  ```
AdminTask.updateBPMConfig( [ "-clusterName", Cluster_Name, "-repositoryServerURL". "https://APPSERVER_URL/Proces
AdminConfig.save()
```

d. Start the deployment manager.

e. Restart the cluster.

Related tasks:

"Changing passwords after installation" on page 648
To change the passwords that you specified when you installed IBM Business Process Manager, use the Process Admin Console to change the password and the WebSphere Application Server administrative console to change the password in the corresponding authentication alias.

Related information:

➦ WebSphere Application Server information center

Configuring Oracle Real Application Cluster (RAC) for use with IBM Business Process Manager

Oracle Real Application Clusters (RAC) is an option of an Oracle database that brings together two or more computers to form a clustered database that behaves as a single system. In a RAC database, Oracle processes that are running in separate nodes access the same data from a shared disk storage.

During the initial installation and configuration of IBM Business Process Manager it is not possible to use Oracle RAC as the target database. You must specify the SID of the single instance of the Oracle database during configuration. The JDBC URL that is automatically generated and supported for accessing the Oracle database is **jdbc:oracle:thin:@<hostname>:<port number>:<DBName>**. The JDBC URL must be edited as a post configuration step to use Oracle RAC.

There are two scenarios to consider:

- If you are using a Standalone profile you need to create the profile using SID as the value for the database name.

- If you are setting up a Network Deployment environment, the profiles and the clusters need to be configured using SID.

IBM Business Process Manager always creates the JDBC URL in the following format: **jdbc:oracle:thin:@[HOST][:PORT]:SID**. The format can be edited to one of the following to use Oracle RAC.

- `jdbc:oracle:thin:@//[HOST][:PORT]/SERVICE`

  Where **SERVICE** is the Oracle service name.

  Note: This new syntax can be used with the SCAN feature available on Oracle 11g R2 and can not be used without the SCAN feature.

- General format to specify a URL for an Oracle RAC:

  ```
  jdbc:oracle:thin:@(DESCRIPTION=
  (LOAD_BALANCE=on)
  (ADDRESS=(PROTOCOL=TCP)(HOST=host1)(PORT=1521))
  (ADDRESS=(PROTOCOL=TCP)(HOST=host2)(PORT=1521))
  (CONNECT_DATA=(SERVICE_NAME=service_name)))
  ```

If a Oracle RAC node failover occurs, IBM Business Process Manager stops processing and you might need to restart all of the IBM Business Process Manager nodes. If you experience process-recovery issues, refer to Process recovery issues in certain situations in IBM Business Process Manager V7.5.

1. Specify the RAC URL as the Oracle URL when defining the relevant Oracle JDBC data sources.

   a. In the administrative console, select **Resources** > **JDBC** > **Data sources**.

   b. Edit all of the data sources using Oracle with the JDBC URL to use one of the formats above.

   Refer to Configuring a data source using the administrative console.

2. Update the Authentication Alias Configuration page and run the administrative task to update the URL to use the Oracle RAC. For more information, see Modifying authentication aliases.

3. In a network deployment environment, you must synchronize the nodes that contain Process Center, Process Server, or Performance Data Warehouse cluster members.
   a. In the administrative console, click **System administration** > **Nodes**.
   b. Select all of the nodes and click **Full Resynchronize**.
   c. Stop and restart all of the clusters and servers.

Configuring an LDAP security provider

To use an additional Lightweight Directory Access Protocol (LDAP) security provider in IBM Business Process Manager, you must add the provider to the federated repository.

By default, IBM Business Process Manager is configured with a single federated repository that contains only the internal IBM BPM security provider. Additionally, you can configure an LDAP security provider, such as Microsoft Active Directory, with WebSphere federated repository.

To configure an LDAP security provider in IBM Business Process Manager, complete the following steps:

1. Log on to WebSphere administrative console as tw_user. The default password for the tw_user account is tw_user.
2. Click **Security** > **Global security**.
3. Under **User account repository**, click **Available realm definitions** > **Federated repositories**.
4. Click **Configure**.
5. Under **Related items**, click **Manage repositories**.
6. Click **Add** and specify parameters for the provider that you want to add. For example, to add Microsoft Active Directory, specify values such as the following examples:

Table 106. Parameters for adding a provider

Parameter	Example values
Repository identifier	**SALOMLDAP // change to suit**
Directory type	**Microsoft Windows Active Directory**
Primary host name	**10.1.5.18**
Bind distinguished name	**cn=LDAP_USER,CN=Users,DC=COMPANYQA,DC=com**
Bind password	**pwsaaswp**

7. Click **OK** and then **Save**.
8. Click **Add Base entry to Realm** and specify values such as the following examples:

Table 107. Parameters for adding a base entry to a realm

Base entry name	Example values
Distinguished name of a base entry that uniquely identifies this set of entries in the realm	**cn=Users,DC=COMPANYQA,DC=com**
Distinguished name of a base entry in this repository	**cn=Users,DC=COMPANYQA,DC=com**

9. Click **OK** and then **Save**.
10. On the Global Security page, click **Set as current** and then click **Apply**.
11. Shut down all IBM BPM servers. For a network deployment environment, you can shut down all the servers that are members of the same cluster at the same time by stopping the cluster.
12. Make sure that no duplicate users exist in the IBM BPM internal security provider and the security provider that you just added. If duplicate users exist, errors will occur when you run IBM Business Process Manager product components.

13. Start all IBM BPM servers. For a network deployment environment, you can start all the servers in a cluster (cluster members) in one action by starting the cluster.

   If you have configured a server cluster for your runtime environment, stop and restart all servers in the cluster.

Related tasks:

"Setting up user accounts for stand-alone configurations"
You can use an internal security provider in IBM Business Process Manager to create and maintain users and groups. Additionally, you can combine the internal security provider with an external security provider such as Lightweight Directory Access Protocol (LDAP).

Setting up user accounts for stand-alone configurations

You can use an internal security provider in IBM Business Process Manager to create and maintain users and groups. Additionally, you can combine the internal security provider with an external security provider such as Lightweight Directory Access Protocol (LDAP).

The internal security provider in IBM Business Process Manager includes several default users and groups. When you use the internal security provider with an external provider, the users and groups from both providers are available for selection.

Important: Do not remove the default IBM BPM administrator account, tw_admin, or the default administrator group, tw_admins. You cannot administer servers and users without these default accounts. To change the password for the tw_admin account, see Managing default users and groups.

The best way to manage security in IBM BPM is by using groups. For example, to grant administrative access to IBM BPM, add preexisting groups of users from your external security provider to tw_admins, which is a IBM BPM group whose members have administrative access to IBM BPM product components by default. Then, when changes are required, you can add or remove individual users from the groups that exist in your external security provider. This practice ensures that the security maintenance that you perform in your external provider does not require additional work in IBM BPM.

To set up your user accounts for stand-alone configurations when you initially configure IBM BPM servers, complete the following steps:

1. To access the Process Admin Console, in your web browser, go to **http://[host_name]:[port]/ ProcessAdmin** . Log on using the default administrative account (tw_admin). By default, the password will be set to the administrative password used during profile creation.

2. Optional: Create the different types of users and groups that your users will need in IBM BPM.

3. Add members to the default IBM BPM groups or groups that you have created. You can add users and groups from any configured external provider, such as LDAP, and internal IBM BPM users and groups.

4. Add the users and groups who need access to the repository, and grant administrative access to the appropriate users. Then establish who can access each process application and toolkit. The best way to grant access to the repository is to add members to the default group, tw_authors.

Related tasks:

"Configuring an LDAP security provider" on page 647
To use an additional Lightweight Directory Access Protocol (LDAP) security provider in IBM Business Process Manager, you must add the provider to the federated repository.

Changing passwords after installation

To change the passwords that you specified when you installed IBM Business Process Manager, use the Process Admin Console to change the password and the WebSphere Application Server administrative console to change the password in the corresponding authentication alias.

If you are using the internal security provider, use the Process Admin Console to maintain user accounts. If you configured a different type of user registry, you must maintain user accounts in the user interface for that user registry.

In addition to changing the **tw_admin** and **tw_user** passwords, you can also change all of the out-of-the-box passwords in your network deployment environment.

Note: If you do not have BPMAdmin_Auth_Alias defined in your environment, your profiles were most likely created before you upgraded to IBM Business Process Manager version 7.5.1. In this case, you must follow the process for changing the password in the IBM Business Process Manager version 7.5.0 information center.

Related tasks:

"Encrypting passwords" on page 657
When you install IBM Business Process Manager product components, passwords in the installed files are generally encrypted by default. If you need to change these passwords, you can use the IBM BPM EncryptPassword utility to encrypt the new passwords.

"Customizing the settings used by Process Server to connect to Process Center" on page 640
After you install and configure IBM Business Process Manager, you can change the Process Server from an offline server to a Process Center connected server, and vice versa.

Changing the tw_admin password in a stand-alone environment:

To change the **tw_admin** password in a stand-alone environment that you specified when you installed IBM Business Process Manager, use the Process Admin Console to change the password and the WebSphere Application Server administrative console to change the password in the corresponding authentication alias.

If you do not have BPMAdmin_Auth_Alias defined in your environment, your profiles were most likely created before you upgraded to IBM Business Process Manager version 7.5.1. In this case, follow the process for changing the password in the IBM Business Process Manager version 7.5.0 information center.

Important: If you are using the internal security provider, use the Process Admin Console to maintain user accounts. If you configured a different type of user registry, you must maintain user accounts in the user interface for that user registry.

For example, if you used federated repositories with the file-based repository, which is the default repository in a WebSphere Application Server Network Deployment environment, use the WebSphere Application Server administrative console to make changes to user accounts.

For example, if you use federated repositories with one or more LDAP repositories, use the LDAP user interface to make changes to user accounts. If **tw_admin** is a user from **LDAP** for example, then the password must be changed in the LDAP.

1. Log on to the Process Admin Console using the administrative account. The administrative account has the user name and password that was specified when the product was installed.
2. Change the password for the **tw_admin** account in the Process Admin Console.
3. Log in to the WebSphere Application Server administrative console.
4. Change the BPMAdmin_Auth_Alias password.
   a. In the WebSphere Application Server administrative console, click **Security** > **Global security**.
   b. On the right side, under **Authentication**, click **JAASConfiguration** > **J2C Authentication data**.
   c. Navigate to **BPMAdmin_Auth_Alias**.
   d. Change the **BPMAdmin_Auth_Alias** password.
5. Modify the **tw_admin** password for the roles associated with the **tw_admin** administrative user name using one of the following methods:

- Go to **Enterprise Applications** > **IBM_BPM_Teamworks_<*node and server location*>** > **User RunAs roles**, for example: **Enterprise Applications** > **IBM_BPM_Teamworks_Node01_Server01** > **User RunAs roles**. Change the password for the roles with which the **tw_admin** administrative user name is associated and apply the change. For example, if the **twem** and **twuser** roles are associated with the **tw_admin** user name, change the password for those roles.
  - Run the **util\Security\bpmModifyMapRunAsRole.py** utility to update the password for the administrative user for the system applications. See Defining RunAs roles user assignments for system applications for more information.
6. Log out of the WebSphere Application Server administrative console.
7. Restart the server.

Related information:

⇥ Changing the default administrative password in IBM BPM v7.5

Changing the tw_user password in a stand-alone environment:

To change the **tw_user** password that you specified when you installed IBM Business Process Manager in a stand-alone environment, use the Process Admin Console to change the password and the WebSphere Application Server administrative console to change the password in the corresponding authentication alias.

If you do not have BPMAdmin_Auth_Alias defined in your environment, your profiles were most likely created before you upgraded to IBM Business Process Manager version 7.5.1. In this case, you must follow the process for changing the password in the IBM Business Process Manager version 7.5.0 information center.

Important: If you are using the internal security provider, use the Process Admin Console to maintain user accounts. If you configured a different type of user registry, you must maintain user accounts in the user interface for that user registry.

For example, if you used federated repositories with the file-based repository, which is the default repository in a WebSphere Application Server Network Deployment environment, use the WebSphere Application Server administrative console to make changes to user accounts.

For example, if you use federated repositories with one or more LDAP repositories, use the LDAP user interface to make changes to user accounts. If **tw_user** is a user from **LDAP** for example, then the password must be changed in the LDAP.
1. Log on to the Process Admin Console using the administrative account. The administrative account has the user name and password that was specified when the product was installed.
2. Change the password for the **tw_user** account.
   a. In the Process Admin Console, click **User Management** > **User Management**.
   b. Input **tw_user** in the **Retrieve Profile** field and click **Retrieve**.
   c. Click **tw_user** in the **Internal IBM BPM Users** section.
   d. In the **Internal IBM BPM User Details** section, type in the new **Password**.
   e. Click **Update**.
3. Log in to the WebSphere Application Server Administrative console.
4. Change the **BPMUser_Auth_Alias** password.
   a. In the WebSphere Application Server administrative console, click **Security** > **Global security**.
   b. On the right side, under **Authentication**, click **JAASConfiguration** > **J2C Authentication data**.
   c. Change the **BPMUser_Auth_Alias** password.

5. Go to **Enterprise Applications** > **IBM_BPM_PerformanceDW_<*node and server location*>** > **User RunAs roles**, for example: **Enterprise Applications** > **IBM_BPM_PerformanceDW_Node01_Server01** > **User RunAs roles**.

6. Change the password for the roles with which the **tw_user** user name is associated and apply the change. For example, if the **twem** and **twuser** roles are associated with the **tw_user** user name, change the password for those roles.

7. Log out of the WebSphere Application Server Administrative console.

Related information:

⏏ Changing the default administrative password in IBM BPM v7.5

Changing the tw_admin password in an ND cluster environment:

To change the **tw_admin** password in an ND cluster environment that you specified when you installed IBM Business Process Manager, use the WebSphere Application Server administrative console to change the password.

If you do not have BPMAdmin_Auth_Alias defined in your environment, your profiles were most likely created before you upgraded to IBM Business Process Manager version 7.5.1. In this case, follow the process for changing the password in the IBM Business Process Manager version 7.5.0 information center.

Important: If you are using the internal security provider, use the Process Admin Console to maintain user accounts. If you configured a different type of user registry, you must maintain user accounts in the user interface for that user registry.

For example, if you used federated repositories with the file-based repository, which is the default repository in a WebSphere Application Server Network Deployment environment, use the WebSphere Application Server administrative console to make changes to user accounts.

For example, if you use federated repositories with one or more LDAP repositories, use the LDAP user interface to make changes to user accounts. If **tw_admin** is a user from **NDCluster LDAP** for example, then the password must be changed in the NDCluster LDAP.

1. Log in to the WebSphere Application Server administrative console.

2. Navigate to **Users and Groups** > **Manage Users** and locate the **tw_admin** user account.

3. Change the password for the **tw_admin** user account.

4. Change the **BPMAdmin_Auth_Alias** password.

   a. In the WebSphere Application Server administrative console, click **Security** > **Global security**.

   b. On the right side, under **Authentication**, click **JAASConfiguration** > **J2C Authentication data**.

   c. Navigate to **BPMAdmin_Auth_Alias**.

   d. Change the **BPMAdmin_Auth_Alias** password.

5. Modify the **tw_admin** password for the roles associated with the **tw_admin** administrative user name using one of the following methods:

   • Go to **Enterprise Applications** > **IBM_BPM_Teamworks_<*node and server location*>** > **User RunAs roles**, for example: **Enterprise Applications** > **IBM_BPM_Teamworks_Node01_Server01** > **User RunAs roles**. Change the password for the roles with which the **tw_admin** administrative user name is associated and apply the change. For example, if the **twem** and **twuser** roles are associated with the **tw_admin** user name, change the password for those roles.

   • Run the **util\Security\bpmModifyMapRunAsRole.py** utility to update the password for the administrative user for the system applications. See Defining RunAs roles user assignments for system applications for more information.

     Important: You must run the **bpmModifyMapRunAsRole.py** utility twice in a clustered environment. For example,

```
        wsadmin.bat -port port number -lang jython -user username -password admin -f C:\WAS_INSTALL_LOCATION\util\Security\bp
        wsadmin.bat -port port number -lang jython -user username -password admin -f C:\WAS_INSTALL_LOCATION\util\Security\bp
```

6. In a network deployment environment, you must synchronize the nodes that contain Process Center, Process Server, or Performance Data Warehouse cluster members.

   a. In the administrative console, click **System administration** > **Nodes**.

   b. Select all of the nodes and click **Full Resynchronize**.

   c. Stop and restart all of the clusters and servers.

7. Restart the cluster members.

Related information:

➡ Changing the default administrative password in IBM BPM v7.5

Changing the tw_user password in an ND cluster environment:

To change the **tw_user** password that you specified when you installed IBM Business Process Manager in an ND cluster environment, use the Process Admin console to change the password and the WebSphere Application Server administrative console to change the password in the corresponding authentication alias.

If you do not have BPMAdmin_Auth_Alias defined in your environment, your profiles were most likely created before you upgraded to IBM Business Process Manager version 7.5.1. In this case, you must follow the process for changing the password in the IBM Business Process Manager version 7.5.0 information center.

Important: If you are using the internal security provider, use the Process Admin Console to maintain user accounts. If you configured a different type of user registry, you must maintain user accounts in the user interface for that user registry.

For example, if you used federated repositories with the file-based repository, which is the default repository in a WebSphere Application Server Network Deployment environment, use the WebSphere Application Server administrative console to make changes to user accounts.

For example, if you use federated repositories with one or more LDAP repositories, use the LDAP user interface to make changes to user accounts. If **tw_user** is a user from **NDCluster LDAP** for example, then the password must be changed in the NDCluster LDAP.

1. Log on to the Process Admin Console using the administrative account. The administrative account has the user name and password that was specified when the product was installed.

2. Change the password for the **tw_user** account.

   a. In the Process Admin Console, click **User Management** > **User Management**.

   b. Input **tw_user** in the **Retrieve Profile** field and click **Retrieve**.

   c. Click **tw_user** in the **Internal IBM BPM Users** section.

   d. In the **Internal IBM BPM User Details** section, type in the new **Password**.

   e. Click **Update**.

3. Log in to the WebSphere Application Server administrative console.

4. Change the **BPMUser_Auth_Alias** password.

   a. In the WebSphere Application Server administrative console, click **Security** > **Global security**.

   b. On the right side, under **Authentication**, click **JAASConfiguration** > **J2C Authentication data**.

   c. Change the **BPMUser_Auth_Alias** password.

5. Go to **Enterprise Applications** > **IBM_BPM_PerformanceDW_<*node and server location*>** > **User RunAs roles**, for example: **Enterprise Applications** > **IBM_BPM_PerformanceDW_Node01_Server01** > **User RunAs roles**.

6. Change the password for the roles with which the **tw_user** user name is associated and apply the change. For example, if the **twem** and **twuser** roles are associated with the **tw_user** user name, change the password for those roles.

7. In a network deployment environment, you must synchronize the nodes that contain Process Center, Process Server, or Performance Data Warehouse cluster members.

   a. In the administrative console, click **System administration** > **Nodes**.

   b. Select all of the nodes and click **Full Resynchronize**.

   c. Stop and restart all of the clusters and servers.

8. Restart the cluster members.

Related information:

➦ Changing the default administrative password in IBM BPM v7.5

Modifying the database password after configuration

You can reconfigure the database password, as needed, once your database configuration is completed.

You reconfigure the database password using the administrative console. There are three processes you need to complete to change the database password.

- Update the authentication alias for all data sources
- Update the Messaging Engine data store authentication alias
- Manually update the Process Server and Performance Data Warehouse password in the 98database.xml file

Important: Before you make any configuration changes, it is suggested that you backup the config folder. The config folder is located beneath the Deployment Manager at *<Dmgr-profile>*/config. Contact your WebSphere Application Server administrator for more information.

Updating the data source authentication alias:

You need to update the data source authentication alias before you change a database password.

Make sure you do all the following items before you begin this procedure.

- For a Network deployment environment, make sure all sending of requests is stopped. You must also stop all the clusters.
- For a Standalone profile, make sure all sending of requests is stopped.
- You must start the DMGR and all node agents after stopping the network deployment clusters and servers before changing the alias.
- All locations for the 98Database.xml file are listed in "Updating the password in the configuration file" on page 588.

To modify the data source authentication alias, complete the following steps:

1. Login to the administrative console.

2. Select **Resources** > **JDBC** > **Data sources**.

   Note: Make sure you write down the values that are configured for **Component-managed authentication alias** and **Authentication alias for XA recovery**. To identify the datasource for the component, see the *ConsolidateJAASAuthAliases.py script* topic in the related information section.

3. Select the desired datasource.

4. Select the Related Items section and then select **JAAS - J2C authentication data**.

5. Select the appropriate **Component-managed authentication alias** value noted in step 2. Edit the password, and then click **OK** to save the change.

6. Go back to the **JAAS - J2C authentication data** panel and select the **Authentication alias for XA recovery** value noted in step 2. Edit the password, and then click **OK** to save the change.
7. Save the updates and click **Full Synchronize** to synchronize the information to all the nodes.

Updating the Messaging Engine data store authentication alias:

You need to update the Messaging Engine data store authentication alias before you modify the database password.

Make sure you do all the following items before you begin this procedure.
- For a Network deployment environment, make sure all sending of requests is stopped. You must also stop all the clusters.
- For a Standalone profile, make sure all sending of requests is stopped.
- You must start the DMGR and all node agents after stopping the network deployment clusters and servers before changing the alias.
- All locations for the 98Database.xml file are listed in "Updating the password in the configuration file" on page 588.

To update the Messaging Engine data store authentication alias, complete the following steps:
1. Login to the administrative console.
2. Select **Service Integration** > **Buses**.
3. To update the authentication alias for each Bus, do the following steps:
   a. Select **Buses** > **[Bus_NAME]** > **Messaging engines** > **[MESSAGING_ENGINE]** > **Data Store** .

      Remember: Make sure you write down and save the **Authentication Alias** value in the Data Store.
   b. Select the Related Items section and click **JAAS - J2C authentication data**.
   c. Select the value noted in step a. Edit the password, and then click **OK** to save the change.
   d. Save the updates and click **Full Synchronize** to synchronize the information to all the nodes.

Updating the password in the configuration file:

You need to update some configuration files in order to modify your database password.

IBM Process Center, IBM Process Server, and IBM Performance Data Warehouse store their database information in the 98Database.xml file that you need to update with the new password. To locate where the password is to be updated, open the 98Database.xml file in an editor and then search for an entry that looks similar to the following:

<encrypted-password>7Yu5Kz3XaYL5aUjbWfISHg==:VyP9e8ytnTz6623hERR4lg==</encrypted-password>

The above entry needs to be updated with the password value generated by using the encryption tool. Use the method described in the *Encrypting passwords* topic in the related links section.

Attention: Make sure the DMGR and the node agents are operating before you begin your password update. Once you have updated the 98Database.xml file with the new password, you must click **Full Synchronize** to percolate your changes to all nodes. And then, you must restart your environment to make sure the new password is activated in your network deployment environment or your stand-alone environment.

Table 1 identifies the locations of the 98Database.xml configuration file in a stand-alone environment.

Table 108. 98Database.xml Configuration file locations for a stand-alone environment

Deployment type	File location
Process Center	Process Center **\<STANDALONE-PROFILE-DIR\>\config\cells\\<cell name\>\nodes\\<node name\>\servers\\<server name\>\process-center\config\system** Performance Data Warehouse **\<STANDALONE-PROFILE-DIR\>\config\cells\\<cell name\>\nodes\\<node name\>\servers\\<server name\>\performance-data-warehouse\config\system**

| Process Server | Process Server

\<STANDALONE-PROFILE-DIR\>\config\cells\\<cell name\>\nodes\\<node name\>\servers\\<server name\>\process-server\config\system

Performance Data Warehouse

\<STANDALONE-PROFILE-DIR\>\config\cells\\<cell name\>\nodes\\<node name\>\servers\\<server name\>\performance-data-warehouse\config\system |

Table 2 identifies the locations of the 98Database.xml configuration file in a network deployment environment.

Table 109. 98Database.xml Configuration file locations for a network deployment environment

Deployment type	File location
Process Center	**1 or 2 cluster topologies** Process Center **<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\process-center\config\system** For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\process-center\config\system** Performance Data Warehouse **<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\performance-data-warehouse\config\system** For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\performance-data-warehouse\config\system**
	3 or 4 cluster topologies Process Center **<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\process-center\config\system** For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\process-center\config\system** Performance Data Warehouse **<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<SUPPORT-CLUSTER-NAME>\performance-data-warehouse\config\system** For each node used as part of the Support Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\performance-data-warehouse\config\system**

Process Server	**1 or 2 cluster topologies**
	Process Server
	<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\process-server\config\system
	For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\process-server\config\system**
	Performance Data Warehouse
	<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\performance-data-warehouse\config\system
	For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\performance-data-warehouse\config\system**
	3 or 4 cluster topologies
	Process Server
	<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<APPLICATION-CLUSTER-NAME>\process-server\config\system
	For each node used as part of the Application Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\process-server\config\system**
	Performance Data Warehouse
	<DMGR-PROFILE-DIR>\config\cells\<cell name>\clusters\<SUPPORT-CLUSTER-NAME>\performance-data-warehouse\config\system
	For each node used as part of the Support Cluster - **<DMGR-PROFILE-DIR>\config\cells\<cell name>\nodes\<node name>\servers\<server name>\performance-data-warehouse\config\system**

Encrypting passwords

When you install IBM Business Process Manager product components, passwords in the installed files are generally encrypted by default. If you need to change these passwords, you can use the IBM BPM EncryptPassword utility to encrypt the new passwords.

To encrypt passwords, complete the following steps:

1. From a command prompt, go to the **[IBM_BPM_home]/BPM/Lombardi/lib** directory.
2. Run the **java -cp utility.jar com.lombardisoftware.utility.EncryptPassword <password>** command, where **<password>** is the password that you want to encrypt.

   For example, run the following command:

   ```
   java -cp utility.jar com.lombardisoftware.utility.EncryptPassword JohnDoe
   ```

   The result is the following code:

   ```
   taVCmTLTWWgkTvfvavu/7g==:sROM4ZbvqRDFYMobWXqvwg==
   ```

   Important: If you are running the utility from a different directory, replace **utility.jar** with the exact path to the **utility.jar** file.
3. If the utility fails with a **ClassNotFound** exception, set your Java home to the following directory before you run the utility again: **[IBM_BPM_home]/AppServer/java/bin/java**

   For example:

4. Replace the existing encrypted password in your IBM BPM configuration files, such as `PROFILE_HOME\config\cells\`*cell_name*`\nodes\`*node_name*`\servers\`*server_name*`\process-center\ config100Custom.xml`.

   Restriction: You cannot use the EncryptPassword utility to decrypt passwords that are already encrypted.

Related tasks:

"Changing passwords after installation" on page 648
To change the passwords that you specified when you installed IBM Business Process Manager, use the Process Admin Console to change the password and the WebSphere Application Server administrative console to change the password in the corresponding authentication alias.

Installing samples manually

You can manually import the advanced samples in a network deployment environment or in a stand-alone server environment.

If you are using a profile, you can check the log file of the Process Center profile that you have created. An error states that advanced samples could not be found during the **importAdvancedSamples** task. This is expected and you can manually install the advanced samples.

To import the samples, you provide the process server database with the appropriate data by running the **importAdvancedSamples** script.

Choose one of the following options to run the **importAdvancedSamples** script:

- If you are using a profile, call the **importAdvancedSamples** script. The script evaluates the appropriate parameter values and imports the advanced samples into the database. The following example imports the advanced samples into a stand-alone profile server environment:

   `Windows`   importAdvancedSamples.bat

   `Linux`   `UNIX`   importAdvancedSamples.sh

- If you are not using a profile, you can call the **importAdvancedSamples** script but you must provide the appropriate parameter values. You can use the following case-sensitive parameters:

   profileName
   You must set the profile name if you are not using the default profile.

   dbJDBCClasspath
   The directory location with the database JDBC driver. Though this parameter is optional, if it is not specified the following location is used: WAS_INSTALL_ROOT/jdbcdrivers.

   nodeName
   The node name parameter is optional, however it must be specified with the **serverName** parameter in a network deployment environment.

   serverName
   The server name parameter is optional, however it must be specified with the **nodeName** parameter in a network deployment environment.

   clusterName
   Providing the name of the cluster is optional.

   Important: Any path value you specify that contains spaces must be surrounded by quotes.

The following example imports the advanced samples into a process center in a network deployment environment:

```
importAdvancedSamples -nodeName node01 -serverName server1
```

The following samples are imported, if they are available:

- advanced-hiring-sample.twx
- procurement-sample.twx

Starting the First steps console

After you install IBM Business Process Manager Standard, you can use the First steps console to verify the installation, start the Profile Management Tool, access product documentation, or direct elements such as servers and administrative consoles related to individual profiles.

A generic version of the console and a version for each profile in your installation are available. Options on each console are displayed dynamically, depending on features you install and the availability of elements on each operating system. Options might include verifying your installation, starting or stopping the server or deployment manager, accessing the administrative console, starting the Profile Management Tool, and accessing the product documentation.

You will usually want to start the version for the profile. The following sections provide detailed information on starting a First steps console based on its version and the platform used on the system:

- "Starting the First steps console for a profile on Linux, UNIX, and Windows platforms"
- "Starting the generic version of the First steps console" on page 660

Restrictions: The First steps console might not start if you use Mozilla as your default browser and it is installed in a location containing a space in the path name. To rectify this problem, perform one of these actions:

- Install Mozilla into a location without a space in the path name.
- Alter the registry key to remove the space.
- Temporarily set Internet Explorer as the default browser and then set Mozilla as the default browser. This approach automatically removes the space from the registry key.

Starting the First steps console for a profile on Linux, UNIX, and Windows platforms

`Linux` `UNIX` `Windows` Start a First steps console for a profile by performing the following steps:

1. Open a command window.
2. Change to the following directory (where *install_root* represents the installation location of the IBM Business Process Manager or WebSphere Enterprise Service Bus profile (*install_root*/profiles/*profile_name*/):
   - For IBM Business Process Manager profiles:
     - `Linux` `UNIX` *profile_root*/firststeps/bpmStd
     - `Windows` *profile_root*\firststeps\bpmStd
   - For WebSphere Enterprise Service Bus profiles:
     - `Linux` `UNIX` *profile_root*/firststeps/esb
     - `Windows` *profile_root*\firststeps\esb
3. Issue the **firststeps** command to start the console:
   - `Linux` `UNIX` ./firststeps.sh
   - `Windows` firststeps.bat

Fast path:

You can also start a version of the First steps console associated with a profile by performing one of the following tasks:

- Check the First steps console check box on the Profile creation complete or Profile augmentation complete panel at the end of the profile creation or augmentation process.

- `Windows` Click **Start** > **Programs** > **IBM** > **BPM Standard 7.5** > **Profiles** > *profile_name* > **First steps**.

Starting the generic version of the First steps console

Start the generic version of the First steps console by performing the following steps.

1. Open a command window.
2. Change to the following directory:

   - `Linux` `UNIX` *install_root*/firststeps/bpmStd

   - `Windows` *install_root*\firststeps\bpmStd

   The variable *install_root* represents the location of the IBM Business Process Manager installation on Linux, UNIX, and Windows systems.

3. Issue one of the following commands to start the console:

   - `Linux` `UNIX` ./firststeps.sh

   - `Windows` firststeps.bat

Fast path:

`Windows` You can also start the generic version of the console on Windows platforms by clicking **Start** > **Programs** > **IBM** > **BPM Standard 7.5** > **First steps**.