WebSphere® Application Server for z/OS, Version 6.1

Installing your application serving environment

GA22-7957-04

Note

Before using this information, be sure to read the general information under “Notices” on page 255.

Compilation date: April 30, 2006

Contents

How to send your comments

Your feedback is important in helping to provide the most accurate and highest quality information.

- To send comments on articles in the WebSphere Application Server Information Center
  1. Display the article in your Web browser and scroll to the end of the article.
  2. Click on the **Feedback** link at the bottom of the article, and a separate window containing an e-mail form appears.
  3. Fill out the e-mail form as instructed, and click on **Submit feedback** .
- To send comments on PDF books, you can e-mail your comments to: **wasdoc@us.ibm.com** or fax them to 919-254-0206.

  Be sure to include the document name and number, the WebSphere Application Server version you are using, and, if applicable, the specific page, table, or figure number on which you are commenting.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Installing on the z/OS operating system

You can install WebSphere Application Server on z/OS systems. Order the WebSphere Application Server for z/OS product to get started.

Installing an application serving environment under z/OS is a two-step process:

1. Load the WebSphere Application Server for z/OS product code using one of two delivery mechanisms:
   - IBM SystemPac or ServerPac
   - MVS Custom-Build Product Delivery Offering (CBPDO)
2. Create the WebSphere Application Server for z/OS runtime environments.

WebSphere Application Server for z/OS overview

The WebSphere Application Server for z/OS product lets you create unlimited numbers of stand-alone application servers using the customization dialog. However, the real strength of the product is the ability to create and manage Network Deployment environments, with managed application server nodes across a sysplex. You can manage all of the application servers as a group or cell by using the administrative console of the deployment manager. A deployment manager manages the configuration for all the managed nodes in its cell and deploys applications to any managed node in the cell. Use the customization dialog to create deployment managers and managed server nodes, and to federate existing stand-alone application servers into your cell.

The main reason to use managed nodes in a cell versus using the same number of stand-alone application servers is the centralized administration that the deployment manager provides for the cell. Another advantage is the ability to create clusters in a managed node that the deployment manager can then manage with simple workload balancing.

How to buy IBM WebSphere Application Server for z/OS software

See your IBM representative for licensing information, or visit the How to buy WebSphere Application Server software Web site.

Overview and new features for installing an application serving environment

Use the links provided in this topic to learn about the installation features.

"What is new for installers"

This topic provides an overview of new and changed features in installation, migration, coexistence, and interoperability support.

"Planning for installation" on page 36

This topic introduces the two methods available to install the product and helps you plan your product code placement and naming before you install the product.

"Planning for product configuration" on page 51

This topic describes the planning involved before you configure the product.

What is new for installers

Installation is an easier, more consistent, and functionally rich experience across platforms, installable components, and types of installations.

New in Version 6.1! indicates new features or changes implemented at the Version 6.1 level. Unmarked items are Version 6.0 improvements that apply also to Version 6.1, which should interest anyone migrating to Version 6.1 from Version 5.x.

Platform support is described on the Detailed system requirements page. A quick summary of new platform support is provided in the Support announcement.

Improved ease of use

Simplified setup and prerequisites

Redesigned product features simplify operating system setup and reduce prerequisites.

See "Preparing the base operating system" on page 41 and "Hardware and software requirements" on page 9.

Better indicators of installation status

Improved logging and status indicators let you easily determine the success or failure of an installation.

New customization tool

New in Version 6.1! A workstation-based Profile Management tool is now available as an alternative to the host-based ISPF Customization Dialog. This tool uses the same worksheets and overall customization flow as the Customization Dialog, and also provides new function, such as the ability to create a complete Network Deployment cell in one pass, including an application server.

See "Using the Profile Management tool" on page 225.

Improved Customization Dialog menus

The Customization Dialog main menus have been reorganized for clarity. They now show the configuration options more clearly and the product data set panel has been updated to simplify setup.

Ability to create managed nodes directly

You now can directly configure managed nodes (with a node agent but no application servers) and add them to a Network Deployment cell. Create new application servers in the node using scripting or the administrative console. It is no longer necessary to create a stand-alone application server (formerly called a base application server) to build a Network Deployment cell.

Improved installation documentation

New installation documentation helps you retrieve relevant information.

Improved capabilities

zFS file system option for performance gains

New in Version 6.1! Customization tools allow the customization file system to be created as either an HFS or zFS file system. The zFS file system has significant performance advantages.

Independent product SDK

The Java Software Development Kit is now shipped, installed and serviced as part of WebSphere Application Server. There is no longer a separate FMID for the SDK. This makes the WebSphere Application Server for z/OS product independent of any other SDK installed on the same z/OS systems. The shipped SDK resides in *app_server_root*/java.

Security enabled by default; installation requires a user ID

New in Version 6.1! Security now is enabled by default at installation time.

Changed product installation root

Additional directory structure changes are introduced.

The default product directory in Version 6.1 is /usr/lpp/zWebSphere/V6R1. The default product directory in Version 6.0 is /usr/lpp/zWebSphere/V6R0.

See “Directory conventions” on page 12.

Migration, upgrade, and change management

Tivoli License Management (ITLM)

New in Version 6.1! This product is enabled for management with IBM Tivoli License Management (ITLM). Tivoli software meets today’s business requirements for software license management and asset tracking.

For more information about ITLM, see http://www.ibm.com/software/tivoli/products/license-mgr/.

Query product version information

You can query product version information.

Incremental cell upgrade

You can incrementally upgrade nodes in a cell.

Removed use of static application environments

The Workload Management dynamic application environment SPE (APAR OW54622, included in z/OS Version 1.5 and above) is a prerequisite for this product running under z/OS Version 1.4. You no longer need to set up or update static application environments in WLM.

Simplified service

The installation program installs the system files (shared binary files), which you do not update until you install a service fix. Creation and configuration of application servers do not change the system files.

Easier removal and reinstallation

Removal and reinstallation of application server environments is simplified and faster. Application server environments are much easier to install and much easier to uninstall than in previous version. Uninstalling an application server profile does not change the shared system files of the product.

Now you can leave profiles installed when uninstalling the product and its core product files.

z/OS Version 1.6 required

Please note this change from Version 6.0 to Version 6.1. WebSphere Application Server requires a minimum z/OS level of Version 1 Release 6. Because Version 1 Release 6 always supports WLM application environments, the customization steps and documentation for static application environments have been removed.

New sample jobs

Please note this change from Version 6.0 to Version 6.1. New sample jobs are provided in the SBBOJCL product data set for:

- Creating z/OS log streams for the error log and transaction XA partner log
- Created a scheduler database in DB2

The jobs include comments for customization, and replace the corresponding panels in the Customization Dialog.

LE and SSL data set requirements

Please note this change from Version 6.0 to Version 6.1. The customization process assumes that the Language Environment load module libraries (SCEERUN and SCEERUN2) and the System SSL load module library (SGSKLOAD or SIEAKNKE) are in the system link list. This reduces the amount of customer information that must be provided during customization.

If they are not in the link list on the target z/OS system, be sure to follow the customization instructions for adding them to JCL and scripts as appropriate.

Administrative console port changes

Please note this change from Version 6.0 to Version 6.1. The stand alone application server now uses a separate virtual host and HTTP transport chains for the administrative console. As a result, the default ports for the stand alone administrative console are now the same as for the administrative console in a Network Deployment cell:

- 9060 (non-SSL)
- 9443 (SSL)

Ports 9080 and 9043 continue to be the default HTTP ports for ordinary applications in a stand alone application server.

Java 1.5 SDK included

Please note this change from Version 6.0 to Version 6.1. WebSphere Application Server includes IBM 31-bit SDK for z/OS, Java 2 Technology Edition, V5 (5655-N98) See IBM 31-bit SDK for z/OS, V5 - Prerequisites to installing the product for z/OS prerequisites that must be installed to support this level of Java.

Changes to SBBOLPA

Please note this change from Version 6.0 to Version 6.1. The modules that are loaded into common storage during node startup have been merged into a single load module, and moved from SBBOLPA to SBBOLOAD. The only module remaining in SBBOLPA is the CTRACE support module, BBORTS61.

Preparation for 64-bit support

Please note this change from Version 6.0 to Version 6.1. A new load module library, SBBGLOAD, is shipped in this release; it contains 64-bit product code, which is not currently enabled. The new library must be APF authorized, and is added to server procedure JCL and scripts during customization. Additional logic is added to the cataloged procedures generated during customization to enable the product with the option of providing 64-bit support in the future.

HTTP transport chains replace HTTP transports

Please note this change from Version 6.0 to Version 6.1. HTTP transport chains are used at initial customization, in place of the HTTP transports used by default in previous releases.

The SAF key ring default name and digital certificate names used during customization have been modified to provide better security isolation, and to reduce the number of browser challenges. See “Digital certificates” on page 64.

Changes to certificate and key ring names

Please note this change from Version 6.0 to Version 6.1. The SAF key ring default name and digital certificate names used during customization have been modified to provide better security isolation, and to reduce the number of browser challenges. See “Digital certificates” on page 64.

Web serving capability

When you install IBM HTTP Server, its plug-in is installed

New in Version 6.1! The IBM HTTP Server installation now includes the Web server plug-in for IBM HTTP Server, eliminating an extra step. You can enter information in just one place when setting up the Web server environment.

Separate installer for Web server plug-ins

New in Version 6.1! The separate plug-ins installer installs the Web server plug-ins on a machine on which WebSphere Application Server is not installed, without incurring the overhead of running the WebSphere Application Server installation.

Configure the Web server more easily

New in Version 6.1! New convenience scripts have been added to the installation package, enabling you to configure another instance of the Web server post-installation. Version 6.0 required running the Plug-ins installer again in order to accomplish this.

Use the administrative console to define Web servers

New in Version 6.1! The stand alone application server administrative console can be used to create Web server definitions. The customization job BBOWCFGW has been removed. The Web server definition function has been incorporated into the BBOWWPFA job.

Flexibility in when to define Web servers

New in Version 6.1! Web servers on non-z/OS systems can be defined during customization (using the customization tools), or after customization (using the administrative clients).

Installable components in the integrated platform

MQ Series integrated JMS removed from product installation

WebSphere Application Server no longer uses MQ Series to support the embedded JMS function contained in the application server. The MQ Series product is not included any more.

You can use either a separate WebSphere MQ license or the new service integration bus architecture for messaging services. If you set up the integrated JMS under Version 5.0 or Version 5.1, you can continue to use any JMS servers on nodes which have not yet migrated to Version 6.1.

New application server address space

Each application server controller now uses a new address space, called the control region adjunct (CRA), to attach to service integration buses for messaging.

Installing the client

Application server clients and plug-in code no longer are shipped in the product hierarchical file system. Instead, they are provided among the installation discs. An additional HFS for client and plug-in code no longer is required.

More usability in selecting application clients to install

New in Version 6.1! J2EE clients and pluggable clients are no longer exclusive features placed on the same feature panel. They are separate setup types that have separate feature panels.

Improved update installer for Application Server Toolkit (AST)

New in Version 6.1! In Version 6, Application Server Toolkit (AST) used the Eclipse Update Manager to retrieve and apply maintenance. In Version 6.1, it uses the Rational Product Updater (RPU), which is built on top of the Update Manager. RPU is able to update the JRE that is bundled with AST, whereas Update Manager knew only how to update Eclipse plug-ins. RPU runs as a separate process from the code it updates, whereas Update Manager runs within AST itself.

IPv6 support

This product version introduces selective support for Internet Protocol Version 6 (IPv6). IPv6 is the next evolution in Internet Protocol beyond the IPv4 standard currently in use in most networks today. IPv6 client workload now can be distributed to back end application servers, and proxy server support is extended to new IPv6 clients and servers.

The key IPv6 enhancement is the expansion of the IP address space from 32 bits to 128 bits, enabling virtually unlimited IP addresses. This addressing capability, along with new functions enabling end-to-end security, improved mobility support, simplified address configuration and management, make IPv6 a critical component in the evolution of e-business and the next generation internet.

The new IPv6 network support extends Load Balancer to IPv6 clients, enabling their workload to be distributed to back end WebSphere application servers for increased flexibility. With proxy server support for IPv6 clients and upstream server connections, you get added network reach and flexibility. For details, see the Edge Component documentation.

IPv6 addresses are supported in JMS connection factories, but require square brackets to distinguish the IPv6 address from the rest of the providerEndpoints string. This is because the colon separator (:) used by the providerEndpoints string is also used as the separator in IPv6 network addresses. The standard format for provider endpoints is host:port:chain, and an IPv6 address might appear as follows:

```
2002:914:fc12:179:9:20:141:42
```

Therefore, an IPv6 provider endpoint setting might be written as:

```
2002:914:fc12:179:9:20:141:42:7276:BootstrapBasicMessaging
```

An IPv6 address can be of variable length; therefore, it is impossible to know where the IPv6 address ends and the port begins. The solution is to surround the IPv6 address with square brackets, as described in RFC2732 on the Internet Engineering Task Force Web site (http://www.ietf.org). For example, the previous providerEndpoints string becomes:

```
[2002:914:fc12:179:9:20:141:42]:7276:BootstrapBasicMessaging
```

IPv6 enablement is introduced using the HTTP transport. Support is not yet available using IIOP.

IPv6 is now supported for Tivoli Access Manager, Version 6.0. IPv6 was not supported on Tivoli Access Manager Version 5.1 or previous versions.

See also the following Web pages:

- Security evaluations for IBM products
- IPv6 enablement at IBM - Overview
- IPv6 enablement at IBM - Products

Packaging

WebSphere Application Server is a family of similar products designed to deliver the right combination of economy and application serving power for your business demands. The contents of the product packages are similar, but each package has unique elements. Use this topic to familiarize yourself with the various products so that you can determine the offering that best fits your needs.

The WebSphere software platform for e-business starts with a foundation that is formed from Web application serving and integration. IBM WebSphere Application Server software provides the core software to deploy, integrate, and manage your e-business applications.

WebSphere Application Server software supports custom-built applications that are either based on integrated WebSphere software platform products or on other third-party products. Such applications can range from dynamic Web presentations to sophisticated transaction processing systems.

IBM WebSphere Application Server for z/OS is a full-function version of the Network Deployment product. WebSphere Application Server for z/OS can support e-business on any scale.

The management interface for all WebSphere Application Server product offerings includes a scripting interface and the administrative console, which is a Web application.

The scripting interface and the administrative console of the deployment manager are enhanced for managing all of the Application Server nodes in a distributed network cell. From the single point of control of the deployment manager, you can define servers, Web server processes, and clusters. You can start and stop servers and cluster members and deploy applications anywhere in the cell.

Product offerings

IBM WebSphere Application Server products provide a next-generation application server on an industry-standard foundation. Each product addresses a distinct set of scenarios and needs. WebSphere Application Server includes the following product offerings:

- WebSphere Application Server
- WebSphere Application Server Network Deployment
- WebSphere Application Server for z/OS

WebSphere Application Server: This product offering addresses the basic programming and run-time needs of desktop developers and single-server production scenarios. The run-time environment addresses standards-based programming for Web and component-based programming, as well as Web services.

The administration model is a single-server environment without clustering, and without centralized administration of multiple server instances.

The development environment offering is the Application Server Toolkit and a trial version of a Rational Tool product.

WebSphere Application Server Network Deployment: This product offering addresses WebSphere Application Server processes that run in multiple-server production scenarios. Use the Network Deployment product to create:

- Cells

  The cell profile is an ease-of-use enhancement that provides a deployment manager profile that contains a federated application server on the same machine or partition.
- Deployment managers

  The deployment manager provides centralized administration of managed Application Server nodes and custom nodes as a single cell. The deployment manager provides basic clustering and caching support, including failover support and workload balancing. Use the Edge components in the Network Deployment package to further extend the environment with sophisticated caching, load balancing, and enhanced security functions.

- Application servers

  The application server profile in the Network Deployment product can run in a deployment manager cell as a managed node or on its own as a stand-alone application server.

- Custom

  A custom profile has an empty node that you must add to a deployment manager cell to customize and use. Use the administrative console of the deployment manager to create servers and clusters on the custom managed node. Consider the custom node as a production-ready shell, ready for customization to contain your production-level servers and applications.

WebSphere Application Server for z/OS: This product offering addresses WebSphere Application Server processes that run in multiple-server production scenarios in a UNIX space in z/OS regions. Use WebSphere Application Server for z/OS to create:

- Deployment managers

  The deployment manager provides centralized administration of federated application server nodes and managed nodes as a single cell. The deployment manager provides basic clustering and caching support, including failover support and workload balancing.

- Application servers

  The application server profile in WebSphere Application Server for z/OS can run in a deployment manager cell as a federated node or on its own as a stand-alone application server.

- Custom

  A custom or managed node is an empty node until you add it to the deployment manager cell for customization. Use the administrative console of the deployment manager to create servers and clusters on the managed node. Consider the managed node as a production-ready shell, ready for customization to contain your production-level servers and applications.

Launchpad console

The launchpad is the focal point for installing from the product compact disc, with important guidance for creating typical topology scenarios. Diagrams help you identify the types of systems that you can create. The launchpad is in the root directory of the product CD.

The launchpad works on installable components in the primary packet of compact discs. Select and launch any of the components on the product disc or view their HTML documentation on the disc.

Use the **launchpad** command if the program does not start automatically when you insert the product disc:

If you cannot start the launchpad, see the appropriate locale-specific readme file in the root directory on the disc. You can also launch the installation programs directly, using the information in the following table:

Table 1. Installable components on the product disc and the supplements disc

Launch command	Documentation	Name of installable component
`WAS/install`	`WAS/readme/readme_en.html` `WAS/docs/InstallGuide_en.html`	The name of the component varies per product offering:
`plugin/install`	`plugin/readme/readme_en.html` `plugin/docs/InstallGuide_en.html`	Web server plug-ins for WebSphere Application Server

Table 1. Installable components on the product disc and the supplements disc (continued)

Launch command	Documentation	Name of installable component
`AppClient/install`	`AppClient/readme/readme_en.html` `AppClient/docs/InstallGuide_en.html`	Application Client for WebSphere Application Server Available on 32-bit operating system platforms only.
`InstallationFactory/compressed file` Decompress the file onto your system to install the Installation Factory.	`InstallationFactory/readme/readme_en.html` `InstallationFactory/docs/InstallGuide_en.html`	IBM Installation Factory for WebSphere Application Server
`UpdateInstaller/install`	`UpdateInstaller/readme/readme_en.html` `UpdateInstaller/docs/InstallGuide_en.html`	IBM Update Installer for WebSphere Software

Other directories on the product and supplements discs contain components that are automatically installed during one or more of the installation procedures for the installable components.

The IBM HTTP Server tolerates a non-root installation, but you cannot install the Global Services Kit 7 (GSKit 7) that is required for HTTPS security. You can install IBM HTTP Server as a non-root user, then log on as root to install the GSKit program from the directory on the product disc.

Hardware and software requirements

This topic describes hardware and software prerequisites for installing WebSphere Application Server for z/OS.

See the Supported hardware and software Web page for the complete up-to-date listings on what is supported.

If there is a conflict between the information provided in the information center and the information on the Supported hardware and software pages, the information at the Web site takes precedence. Prerequisites information in the information center is provided as a convenience only.

WebSphere Application Server for z/OS Version 6 Release 1 requires z/OS or z/OS.e Version 1 Release 6 and runs on any hardware that supports the required operating system software.

For detailed hardware and software requirements for installing WebSphere Application Server for z/OS, see “Driving system requirements.”

For detailed hardware and software requirements for customizing and running WebSphere Application Server for z/OS application serving environments, see “Target system requirements” on page 10.

Driving system requirements

This article describes prerequisites for installing WebSphere Application Server for z/OS.

Hardware requirements

The hardware requirements for this product are any hardware that supports z/OS Version 1 Release 6 or later. However, there are significant performance advantages for those applications doing floating point arithmetic if the machine has binary floating point hardware, such as S/390 Parallel Enterprise Server-Generation 5 and later systems.

You should plan on three 3390-3 DASD volumes (or equivalent storage) for the product target and distribution libraries and the product HFS, and an additional 3390-3 DASD volume (or equivalent storage) for CustomPac dialogs and work data sets (if you install using a ServerPac or SystemPac) or for SMP/E work data sets and relfile storage (if you install using a Custom-Build Product Delivery Offering).

Software requirements

The z/OS system used to install WebSphere Application Server for z/OS must run the following software. Consult the Program Directory and PSP bucket for any additional required corrective service not listed here.

- z/OS Version 1 Release 6 or later, or z/OS.e Version 1 Release 6 or later, with fixes for Language Environment APARs IR54653 and IO00462.
- z/OS UNIX System Services (z/OS UNIX) with an HFS or zFS file system configured. For details, see *z/OS UNIX System Services Planning*.
- IBM Developer Kit for OS/390 Java 2 Technology Edition Version 1.1 or another Java Software Development Kit (level 1.2 or above). This is required to provide the jar command used before SMP/E APPLY processing. You must mount the Java SDK file system during APPLY processing and ensure that the directory containing the jar command is in the PATH of the user ID used to perform the SMP/E APPLY.

Target system requirements

This article describes prerequisites for configuring and running WebSphere Application Server for z/OS application serving environments.

Hardware requirements

The hardware requirements for this product are any hardware that supports z/OS Version 1 Release 4 or later. However, there are significant performance advantages for those applications doing floating point arithmetic if the machine has binary floating point hardware, such as S/390 Parallel Enterprise Server-Generation 5 and later systems.

The LPAR in which the WebSphere Application Server for z/OS runtime and initial application servers run requires a minimum of 512 MB of real storage. You might need to increase the real storage size depending on the size and number of application servers you deploy. In addition, you might want to increase your JES spool space if you use WebSphere Application Server for z/OS tracing options to the SYSPRINT DD data set.

In addition to the DASD volumes used to hold the product code and file system, you will need additional DASD storage to hold configuration data for application serving environments. The amount of storage depends on the number of environments and the size and complexity of the applications being deployed.

WebSphere Application Server for z/OS is a heavy user of auxiliary storage. You might want to add additional paging volumes before configuring application serving environments.

WebSphere Application Server for z/OS V6.1 offers full support for zAAPs. zAAPs are designed to operate asynchronously with the general purpose processors when executing Java™ programming under control of the IBM JVM. The IBM JVM processing cycles can be executed on the configured zAAPS, with no anticipated modifications to the Java applications.

The zAAPs may be purchased and installed on z9 – 109, z990 and z890 servers (and follow-on models only). In order to exploit a zAAP, the operating system level must be z/OS version 1 release 6 (or z/OS.e version 1 release 6).

IBM System z® Application Assist Processors (zAAPs) are attractively priced specialized processing units that provide a strategic z/OS Java™ execution environment for customers who desire the powerful integration advantages and traditional qualities of service of the IBM mainframe platform. zAAP can help you strategically integrate and run new Java-based Web applications with core z/OS business applications and backend database systems, increase overall system productivity, and ultimately lower the overall cost of computing for running Java technology-based workloads on the platform.

For more details, see the zAAP Redbook at http://www.redbooks.ibm.com/abstracts/sg246386.html?Open .

Software requirements

You must install, enable, and configure the following z/OS elements, features, and components on each z/OS target system. Consult the WebSphere Application Server for z/OS Program Directory and PSP bucket for any additional required corrective service not listed here. In some cases, this corrective service must be installed on each target system for WebSphere Application Server to start.

All of the z/OS sources referenced are available at this Web site: http://www.ibm.com/servers/eserver/zseries/zos/bkserv/

- z/OS (or z/OS.e) Version 1 Release 6 or later, configured as a sysplex (in the case of a single z/OS system, as a monoplex). For details, see *z/OS MVS Setting Up a Sysplex*.
- z/OS UNIX System Services (z/OS UNIX) with an HFS or zFS file system. For details, see *z/OS UNIX System Services Planning*.
- eNetwork Communications Server (TCP/IP) or equivalent. In this manual, we refer to eNetwork Communications Server; but you can substitute an equivalent product. For details, see *z/OS Communications Server: IP Migration*.
- Resource recovery services (RRS). For details, see *z/OS MVS Programming: Resource Recovery*.
- System logger. For details, see *z/OS MVS Setting up a Sysplex*.
- A security product such as z/OS Security Server (RACF). In this manual, we refer to Security Server in examples; but you can substitute an equivalent security product. For details, see *z/OS Security Server RACF Migration*.

Additional software might be required to support particular product functions.

Table 2. Software requirements for optional functions

If you plan to use . . .	Then you need . . .	Notes . . .
WebSphere Application Server for z/OS IMS Connect Version 8.1 support	IMS/TM 6.1.0	the connector is called IMS Connector for Java V2.2
WebSphere Application Server for z/OS CICS Transaction Gateway 5.0.1 support	CICS/TS V2.2	CTG V6.0 (CICS Transaction Gateway)
DB2	DB2 Version 7.1	DB2 Version 7.1 and above are supported for use with configuration options and applications which may require a database.

Table 2. Software requirements for optional functions (continued)

If you plan to use . . .	Then you need . . .	Notes . . .
DB2 SQLJ in J2EE application components	DB2 Version 7.1 with PTFs satisfying APARs PQ84404 PQ86525 PQ89043 PQ80841 UQ88238 PQ88082 PQ87786 PQ88082 UQ86911 PQ51847 DB2 Version 8.1 with PTFs satisfying APARs PK00615 PQ84577 PQ87786 PQ88082 UQ86912	DB2 Version 7.1 and above are supported for use with configuration options and applications which may require a database. DB2 V7.1 is no longer a prerequisite for WebSphere Application Server for z/OS, though some functions might still require it. For example, session persistence requires an existing database. Check your configuration. DB2 SQLJ in J2EE application components, DB2 V7.1 PTFUQ59527 DB2 is no longer a prerequisite for Websphere Application Server for z/OS, though some functions might still require it..
Connectors	WebSphere Application Server for z/OS supports any resource adapter that is designed to use the 1.0 level of the J2EE Connector Architecture (JCA). Install the listed IBM connectors if you want to access your IMS or CICS legacy data. For details on how to define and install a specific connector, see the documentation for that particular connector product.	
	For the CICS Transaction Gateway ECI connector: • CICS Transaction Gateway Version 5.0.1 • CICS Transaction Server Version 1.3 • WebSphere Studio Application Developer IE Version 5.0	
	For the IMS Connector for Java: • IMS Connect for z/OS Version 2.1 • IMS Version 8 • WebSphere Studio Application Developer IE Version 5.0.1	
	For the IMS JDBC Connector: • IMS Version 8	

Directory conventions

References in the product information to *app_server_root*, *profile_root*, and other directories imply specific directory locations. This topic describes the conventions in use for WebSphere Application Server for z/OS.

smpe_root

Refers to the root directory for product code installed with SMP/E.

The corresponding product variable is smpe.install.root.

The default is /usr/lpp/zWebSphere/V6R1.

configuration_root

Refers to the mount point for the configuration file system (formerly, the configuration HFS) in WebSphere Application Server for z/OS.

The configuration_root contains the various app_server_root directories and certain symbolic links associated with them. Each different configuration_root normally requires its own cataloged procedures under z/OS.

The default is /WebSphere/V6R1.

app_server_root

Refers to the top directory for a WebSphere Application Server node.

The node may be of any type—application server, deployment manager, or unmanaged for example. Each node has its own app_server_root. Before Version 6.0 of the product information, this was referred to as the ″WAS_HOME″ directory. Corresponding product variables are was.install.root and WAS_HOME.

The default varies based on node type. Common defaults are *configuration_root*/Appserver and *configuration_root*/DeploymentManager.

profile_root

Refers to the home directory for a particular instantiated WebSphere Application Server profile.

Corresponding product variables are server.root and user.install.root.

In general, this is the same as *app_server_root*/profiles/*profile_name*. On z/OS, this will be always be *app_server_root*/profiles/default because only the profile name ″default″ is used in WebSphere Application Server for z/OS.

How do I install an application serving environment?

Plan your installation of WebSphere Application Server	Documentation
Review the Release Notes	Documentation
Review the software and hardware prerequisites	Detailed system requirements page
Review the Release Notes	Documentation
Review the software and hardware prerequisites	Documentation
Configure communication with the Web server	
Configure the HTTP channel within the application server	
Enable your Web server to use HTTPS in addition to HTTP	
Establish virtual hosts	
Configure Edge Side Include (ESI) for performance	

Troubleshoot Web server plug-in configurations	
Apply maintenance	Documentation
Review the software and hardware prerequisites	Documentation

Release notes

The Release Notes summarize late-breaking news and product limitations. Ideally, review them prior to installing the product, as addendums to the installation and configuration instructions.

Review the **Release Notes**.

IBM WebSphere Application Server release notes

Links are provided to the product support Web site, to product documentation, and to last minute updates, limitations, and known problems for the WebSphere® Application Server Version 6.x products.

transition: Release notes for the Version 6.1 beta program can be found in the beta package.

- "Accessing last-minute updates, limitations, and known problems"
- "Accessing hardware and software requirements"
- "Accessing product documentation"
- "Accessing the product support Web site" on page 15
- "Contacting IBM Software Support" on page 15

Accessing last-minute updates, limitations, and known problems

Release notes are available on the product support Web site.

- Version 6.1 for z/OS®

Accessing hardware and software requirements

- The hardware and software requirements for the WebSphere Application Server products are provided on the Detailed system requirements page.

Accessing product documentation

The following documentation is installed with this product.

The readme file.

Several readme files are provided:

- The readme_en.html file in the root directory of the product CD-ROM is a shadow of the Launchpad application. It provides links to the installation procedures and documentation for all of the installable components on the product CD. Translated versions of the file are in the launchpad directory.
- The readme.html file in the WAS directory points to this more detailed readme in the WAS directory and to the Getting Started Guide in the WAS/docs directory. This file is not translated.

- The readme_en.html in the readme directory of each installable component, such as this file. The Launchpad links to this file for a description of installing the WebSphere Application Server - Express product. This file is also available from the readme directory in the root directory of the product CD-ROM.

Getting Started Guide

The InstallGuide_en.pdf and InstallGuide_en.html files in the WAS/docs directory provide detailed instructions for installing the product. The information in the guide is derived from information in the online information center. The guide is not translated on the product CD; however, translated versions of the information center are available online.

Help files

The help files provide detailed instructions for completing tasks and for specifying settings in the graphical systems management tools. Use the help menus, links, or buttons provided in the tool interfaces to access the help files.

Documentation on the Web

For the entire documentation set for all WebSphere Application Server products, including the information center and Adobe Acrobat PDF versions of the information, go to the WebSphere Application Server library page. All product documentation is in the information center, including versions of the installed help files.

Accessing the product support Web site

- To search for the latest troubleshooting tips, downloads, fixes, and other support-related information, go to the WebSphere Application Server support page.
- To search for z/OS support-related information only, go to the WebSphere Application Server for z/OS support page.

Contacting IBM® Software Support

If you encounter a problem with this product, first try the following actions:

- Following the steps described in the product documentation
- Looking for related documentation in the online help
- Looking up error messages in the message reference

If you cannot resolve your problem by any of the preceding methods, contact IBM Technical Support.

Purchase of IBM WebSphere Application Server entitles you to one year of telephone support under the Passport Advantage® program. For details about Passport Advantage, visit the Passport Advantage page.

The number for Passport Advantage members to call for WebSphere Application Server support is 1-800-426-7378. Have the following information available when you call:

- Your contract or Passport Advantage number.
- Your WebSphere Application Server version and revision level, plus any installed fixes.
- Your operating system name and version.
- Your database type and version.
- Basic topology data: how many machines are running, how many application servers, and so on.
- Any error or warning messages that are related to your problem.

Task overview: installing

This article describes the process of installing and configuring WebSphere Application Server for z/OS.

This article introduces the context of installing and customizing IBM WebSphere Application Server for z/OS, including the tasks you need to perform before and after installing. The product is provided in both U.S. English and Japanese.

To create a complete, customized WebSphere Application Server for z/OS application serving environment, you need to install the product code, prepare the z/OS operating system and subsystems, run the Customization Dialog or Profile Management tool, follow the customized instructions and run the jobs from the dialog, and bring up your servers.

Note: See “Building a practice WebSphere Application Server for z/OS cell” on page 84 for steps you can follow to set up a practice version of WebSphere Application Server for z/OS if you want to just get the feel for it or see the basics.

Perform the following tasks to create a running version of the product on your machine.

1. Plan data set names and layout for product code installation, as described in “Planning for installation” on page 36.
2. Install WebSphere Application Server for z/OS, as described in “Installing the product and additional software” on page 40. You must first load the WebSphere Application Server for z/OS code onto your system, using SMP/E or a preloaded product offering, before you make it usable through customization.
3. Prepare your z/OS target systems to run WebSphere Application Server for z/OS, as described in “Preparing the base operating system” on page 41.
4. Choose your application serving environment and decide on its initial characteristics, as described in “Planning for product configuration” on page 51.
5. Configure WebSphere Application Server for z/OS, as described in “Configuring the product after installation” on page 131. Once you have the WebSphere Application Server for z/OS code installed on your system, you are ready to make it your own by customizing it.
6. To create additional application serving environments, repeat the steps in “Planning for product configuration” on page 51 and “Configuring the product after installation” on page 131.
7. Tune for performance. For best performance on any platform, see Tuning the application serving environment.

You are ready to deploy and run applications using the WebSphere Application Server for z/OS product.

Packaging

WebSphere Application Server is a family of similar products designed to deliver the right combination of economy and application serving power for your business demands. The contents of the product packages are similar, but each package has unique elements. Use this topic to familiarize yourself with the various products so that you can determine the offering that best fits your needs.

The WebSphere software platform for e-business starts with a foundation that is formed from Web application serving and integration. IBM WebSphere Application Server software provides the core software to deploy, integrate, and manage your e-business applications.

WebSphere Application Server software supports custom-built applications that are either based on integrated WebSphere software platform products or on other third-party products. Such applications can range from dynamic Web presentations to sophisticated transaction processing systems.

IBM WebSphere Application Server for z/OS is a full-function version of the Network Deployment product. WebSphere Application Server for z/OS can support e-business on any scale.

The management interface for all WebSphere Application Server product offerings includes a scripting interface and the administrative console, which is a Web application.

The scripting interface and the administrative console of the deployment manager are enhanced for managing all of the Application Server nodes in a distributed network cell. From the single point of control of the deployment manager, you can define servers, Web server processes, and clusters. You can start and stop servers and cluster members and deploy applications anywhere in the cell.

Product offerings

IBM WebSphere Application Server products provide a next-generation application server on an industry-standard foundation. Each product addresses a distinct set of scenarios and needs. WebSphere Application Server includes the following product offerings:

- WebSphere Application Server
- WebSphere Application Server Network Deployment
- WebSphere Application Server for z/OS

WebSphere Application Server: This product offering addresses the basic programming and run-time needs of desktop developers and single-server production scenarios. The run-time environment addresses standards-based programming for Web and component-based programming, as well as Web services.

The administration model is a single-server environment without clustering, and without centralized administration of multiple server instances.

The development environment offering is the Application Server Toolkit and a trial version of a Rational Tool product.

WebSphere Application Server Network Deployment: This product offering addresses WebSphere Application Server processes that run in multiple-server production scenarios. Use the Network Deployment product to create:

- Cells

  The cell profile is an ease-of-use enhancement that provides a deployment manager profile that contains a federated application server on the same machine or partition.
- Deployment managers

  The deployment manager provides centralized administration of managed Application Server nodes and custom nodes as a single cell. The deployment manager provides basic clustering and caching support, including failover support and workload balancing. Use the Edge components in the Network Deployment package to further extend the environment with sophisticated caching, load balancing, and enhanced security functions.
- Application servers

  The application server profile in the Network Deployment product can run in a deployment manager cell as a managed node or on its own as a stand-alone application server.
- Custom

  A custom profile has an empty node that you must add to a deployment manager cell to customize and use. Use the administrative console of the deployment manager to create servers and clusters on the custom managed node. Consider the custom node as a production-ready shell, ready for customization to contain your production-level servers and applications.

WebSphere Application Server for z/OS: This product offering addresses WebSphere Application Server processes that run in multiple-server production scenarios in a UNIX space in z/OS regions. Use WebSphere Application Server for z/OS to create:

- Deployment managers

The deployment manager provides centralized administration of federated application server nodes and managed nodes as a single cell. The deployment manager provides basic clustering and caching support, including failover support and workload balancing.

- Application servers

  The application server profile in WebSphere Application Server for z/OS can run in a deployment manager cell as a federated node or on its own as a stand-alone application server.

- Custom

  A custom or managed node is an empty node until you add it to the deployment manager cell for customization. Use the administrative console of the deployment manager to create servers and clusters on the managed node. Consider the managed node as a production-ready shell, ready for customization to contain your production-level servers and applications.

Launchpad console

The launchpad is the focal point for installing from the product compact disc, with important guidance for creating typical topology scenarios. Diagrams help you identify the types of systems that you can create. The launchpad is in the root directory of the product CD.

The launchpad works on installable components in the primary packet of compact discs. Select and launch any of the components on the product disc or view their HTML documentation on the disc.

Use the **launchpad** command if the program does not start automatically when you insert the product disc:

If you cannot start the launchpad, see the appropriate locale-specific readme file in the root directory on the disc. You can also launch the installation programs directly, using the information in the following table:

Table 3. Installable components on the product disc and the supplements disc

Launch command	Documentation	Name of installable component
`WAS/install`	`WAS/readme/readme_en.html` `WAS/docs/InstallGuide_en.html`	The name of the component varies per product offering:
`plugin/install`	`plugin/readme/readme_en.html` `plugin/docs/InstallGuide_en.html`	Web server plug-ins for WebSphere Application Server
`AppClient/install`	`AppClient/readme/readme_en.html` `AppClient/docs/InstallGuide_en.html`	Application Client for WebSphere Application Server Available on 32-bit operating system platforms only.

`InstallationFactory/ compressed file`
Decompress the file onto your system to install the Installation Factory.
	`InstallationFactory/readme/readme_en.html`
	`InstallationFactory/docs/ InstallGuide_en.html`
		IBM Installation Factory for WebSphere Application Server

`UpdateInstaller/ install`	`UpdateInstaller/readme/readme_en.html` `UpdateInstaller/docs/InstallGuide_en.html`	IBM Update Installer for WebSphere Software

Other directories on the product and supplements discs contain components that are automatically installed during one or more of the installation procedures for the installable components.

The IBM HTTP Server tolerates a non-root installation, but you cannot install the Global Services Kit 7 (GSKit 7) that is required for HTTPS security. You can install IBM HTTP Server as a non-root user, then log on as root to install the GSKit program from the directory on the product disc.

Hardware and software requirements

This topic describes hardware and software prerequisites for installing WebSphere Application Server for z/OS.

See the Supported hardware and software Web page for the complete up-to-date listings on what is supported.

If there is a conflict between the information provided in the information center and the information on the Supported hardware and software pages, the information at the Web site takes precedence. Prerequisites information in the information center is provided as a convenience only.

WebSphere Application Server for z/OS Version 6 Release 1 requires z/OS or z/OS.e Version 1 Release 6 and runs on any hardware that supports the required operating system software.

For detailed hardware and software requirements for installing WebSphere Application Server for z/OS, see “Driving system requirements” on page 9.

For detailed hardware and software requirements for customizing and running WebSphere Application Server for z/OS application serving environments, see “Target system requirements” on page 10.

Driving system requirements

This article describes prerequisites for installing WebSphere Application Server for z/OS.

Hardware requirements

The hardware requirements for this product are any hardware that supports z/OS Version 1 Release 6 or later. However, there are significant performance advantages for those applications doing floating point arithmetic if the machine has binary floating point hardware, such as S/390 Parallel Enterprise Server-Generation 5 and later systems.

You should plan on three 3390-3 DASD volumes (or equivalent storage) for the product target and distribution libraries and the product HFS, and an additional 3390-3 DASD volume (or equivalent storage) for CustomPac dialogs and work data sets (if you install using a ServerPac or SystemPac) or for SMP/E work data sets and relfile storage (if you install using a Custom-Build Product Delivery Offering).

Software requirements

The z/OS system used to install WebSphere Application Server for z/OS must run the following software. Consult the Program Directory and PSP bucket for any additional required corrective service not listed here.

- z/OS Version 1 Release 6 or later, or z/OS.e Version 1 Release 6 or later, with fixes for Language Environment APARs IR54653 and IO00462.
- z/OS UNIX System Services (z/OS UNIX) with an HFS or zFS file system configured. For details, see *z/OS UNIX System Services Planning*.
- IBM Developer Kit for OS/390 Java 2 Technology Edition Version 1.1 or another Java Software Development Kit (level 1.2 or above). This is required to provide the jar command used before SMP/E APPLY processing. You must mount the Java SDK file system during APPLY processing and ensure that the directory containing the jar command is in the PATH of the user ID used to perform the SMP/E APPLY.

Target system requirements

This article describes prerequisites for configuring and running WebSphere Application Server for z/OS application serving environments.

Hardware requirements

The hardware requirements for this product are any hardware that supports z/OS Version 1 Release 4 or later. However, there are significant performance advantages for those applications doing floating point arithmetic if the machine has binary floating point hardware, such as S/390 Parallel Enterprise Server-Generation 5 and later systems.

The LPAR in which the WebSphere Application Server for z/OS runtime and initial application servers run requires a minimum of 512 MB of real storage. You might need to increase the real storage size depending on the size and number of application servers you deploy. In addition, you might want to increase your JES spool space if you use WebSphere Application Server for z/OS tracing options to the SYSPRINT DD data set.

In addition to the DASD volumes used to hold the product code and file system, you will need additional DASD storage to hold configuration data for application serving environments. The amount of storage depends on the number of environments and the size and complexity of the applications being deployed.

WebSphere Application Server for z/OS is a heavy user of auxiliary storage. You might want to add additional paging volumes before configuring application serving environments.

WebSphere Application Server for z/OS V6.1 offers full support for zAAPs. zAAPs are designed to operate asynchronously with the general purpose processors when executing Java™ programming under control of the IBM JVM. The IBM JVM processing cycles can be executed on the configured zAAPS, with no anticipated modifications to the Java applications.

The zAAPs may be purchased and installed on z9 – 109, z990 and z890 servers (and follow-on models only). In order to exploit a zAAP, the operating system level must be z/OS version 1 release 6 (or z/OS.e version 1 release 6).

IBM System z® Application Assist Processors (zAAPs) are attractively priced specialized processing units that provide a strategic z/OS Java™ execution environment for customers who desire the powerful integration advantages and traditional qualities of service of the IBM mainframe platform. zAAP can help you strategically integrate and run new Java-based Web applications with core z/OS business applications and backend database systems, increase overall system productivity, and ultimately lower the overall cost of computing for running Java technology-based workloads on the platform.

For more details, see the zAAP Redbook at http://www.redbooks.ibm.com/abstracts/sg246386.html?Open .

Software requirements

You must install, enable, and configure the following z/OS elements, features, and components on each z/OS target system. Consult the WebSphere Application Server for z/OS Program Directory and PSP bucket for any additional required corrective service not listed here. In some cases, this corrective service must be installed on each target system for WebSphere Application Server to start.

All of the z/OS sources referenced are available at this Web site: http://www.ibm.com/servers/eserver/zseries/zos/bkserv/

- z/OS (or z/OS.e) Version 1 Release 6 or later, configured as a sysplex (in the case of a single z/OS system, as a monoplex). For details, see *z/OS MVS Setting Up a Sysplex*.
- z/OS UNIX System Services (z/OS UNIX) with an HFS or zFS file system. For details, see *z/OS UNIX System Services Planning*.

- eNetwork Communications Server (TCP/IP) or equivalent. In this manual, we refer to eNetwork Communications Server; but you can substitute an equivalent product. For details, see *z/OS Communications Server: IP Migration*.
- Resource recovery services (RRS). For details, see *z/OS MVS Programming: Resource Recovery*.
- System logger. For details, see *z/OS MVS Setting up a Sysplex*.
- A security product such as z/OS Security Server (RACF). In this manual, we refer to Security Server in examples; but you can substitute an equivalent security product. For details, see *z/OS Security Server RACF Migration*.

Additional software might be required to support particular product functions.

Table 4. Software requirements for optional functions

If you plan to use . . .	Then you need . . .	Notes . . .
WebSphere Application Server for z/OS IMS Connect Version 8.1 support	IMS/TM 6.1.0	the connector is called IMS Connector for Java V2.2
WebSphere Application Server for z/OS CICS Transaction Gateway 5.0.1 support	CICS/TS V2.2	CTG V6.0 (CICS Transaction Gateway)
DB2	DB2 Version 7.1	DB2 Version 7.1 and above are supported for use with configuration options and applications which may require a database.
DB2 SQLJ in J2EE application components	DB2 Version 7.1 with PTFs satisfying APARs PQ84404 PQ86525 PQ89043 PQ80841 UQ88238 PQ88082 PQ87786 PQ88082 UQ86911 PQ51847 DB2 Version 8.1 with PTFs satisfying APARs PK00615 PQ84577 PQ87786 PQ88082 UQ86912	DB2 Version 7.1 and above are supported for use with configuration options and applications which may require a database. DB2 V7.1 is no longer a prerequisite for WebSphere Application Server for z/OS, though some functions might still require it. For example, session persistence requires an existing database. Check your configuration. DB2 SQLJ in J2EE application components, DB2 V7.1 PTFUQ59527 DB2 is no longer a prerequisite for Websphere Application Server for z/OS, though some functions might still require it..
Connectors	WebSphere Application Server for z/OS supports any resource adapter that is designed to use the 1.0 level of the J2EE Connector Architecture (JCA). Install the listed IBM connectors if you want to access your IMS or CICS legacy data. For details on how to define and install a specific connector, see the documentation for that particular connector product.	
	For the CICS Transaction Gateway ECI connector: • CICS Transaction Gateway Version 5.0.1 • CICS Transaction Server Version 1.3 • WebSphere Studio Application Developer IE Version 5.0	
	For the IMS Connector for Java: • IMS Connect for z/OS Version 2.1 • IMS Version 8 • WebSphere Studio Application Developer IE Version 5.0.1	

Table 4. Software requirements for optional functions (continued)

If you plan to use . . .	Then you need . . .	Notes . . .
	For the IMS JDBC Connector: • IMS Version 8	

Skill requirements

In assembling your project team, you should consider the skills you need to implement WebSphere Application Server for z/OS. This article discusses the recommended skill set necessary to support the following configurations:

- Basic configurations
- Production environments

Documentation to support the z/OS skills described in this article can be found at the following Web site: z/OS Internet Library

For basic configurations:

Below are the recommended skills necessary to support a basic configuration:

- z/OS UNIX® System Services and the hierarchical file system (HFS) - to set up a functional HFS and UNIX environment
- eNetwork Communications Server (TCP/IP) or equivalent - to configure connectivity for WebSphere Application Server for z/OS clients and servers
- Resource recovery services (RRS) - to implement resource recovery services and to support two-phase commit transactions
- Security Server (RACF®), or the security product you use - to authenticate WebSphere Application Server for z/OS clients and servers, and authorize access to resources
- Secure Sockets Layer (SSL) - to enable security if desired (recommended)
- SMP/E and JCL
- System logger - to set up log streams for RRS and the WebSphere Application Server for z/OS error log
- Webserver - to support HTTP clients if desired
- Workload management (WLM)
- Java™ and WebSphere Application Server tooling - to support application development and deployment

Depending on the needs of the applications you deploy, you might also need skills to configure the resource managers your applications require, such skills might include CICS®, DB2®, IMS™, and MQ.

For production environments:

As you move your system toward a production environment, you need to have the following system skills available:

- Automatic restart management (ARM)
- System Automation, if you have it installed, or whichever automation you prefer to use
- Sysplex if you plan to use WebSphere Application Server for z/OS in a cell that spans systems
- Sysplex Distributor (part of eNetwork Communications Server), if you plan to create a high availability environment
- RMF™ or other performance measurement systems

Steps for creating your implementation plan

We assume that you have a z/OS system on which you will implement WebSphere Application Server for z/OS.

To get started, plan to build your initial WebSphere Application Server for z/OS application serving environment servers on one system, then replicate them on other systems as you expand into a cell. This procedure first guides you through initial planning and implementation of WebSphere Application Server for z/OS on a monoplex. Then, it guides you through setting up your application development and client environments. Finally, the procedure guides you through planning for optional advanced system configurations.

Perform the following steps to implement your plan, checking off each item as you complete it:

1. Determine the skills that you need. See "Skill requirements" on page 22 for more information.
2. Determine WebSphere Application Server for z/OS system requirements. See "Hardware and software requirements" on page 9 for more information.
3. Understand the security options, and prepare for securing your system. See Security planning overview for more information.
4. Implement Workload Management in goal mode on each z/OS system if necessary. See "Workload management (WLM)" on page 68 for more information.
5. Implement Resource Recovery Services (if not already implemented) on each z/OS system. See "Preparing Resource Recovery Services" on page 46 for more information.
6. Plan for your performance and monitoring systems. See Monitoring end user response time for more information.
7. Plan and define your problem diagnosis procedures. See "Problem diagnostic plan strategy" on page 129 for more information.
8. Consider automatic restart management before you install WebSphere Application Server for z/OS. See "Automatic restart management" on page 126 for more information.
9. Plan your product data set and HFS naming conventions. See "Planning for installation" on page 36 for more information.
10. Install the WebSphere Application Server for z/OS product. See "Installing the product and additional software" on page 40 for more information.
11. Prepare your z/OS target systems to run WebSphere Application Server for z/OS. See "Preparing the base operating system" on page 41 for more information.
12. Learn about configuring application serving environments. See "Planning for product configuration" on page 51 for more information.
13. Set up a simple stand-alone application server to verify system readiness and gain experience with a basic application serving environment. See "Building a practice WebSphere Application Server for z/OS cell" on page 84 for more information.
14. Plan and define your system backup procedures. See Backing up the WebSphere Application Server for z/OS system for more information.
15. Plan and define your software service procedures.
16. (Optional) Plan for testing and production systems. See Testing and production phases for more information.
17. Plan and configure the application serving environments that you want. See "Planning for product configuration" on page 51 and "Configuring the product after installation" on page 131.
18. Enable global security (strongly recommended). See Task overview: Securing resources for more information.
19. Develop and deploy applications. See How do I develop and deploy applications? for more information.
20. Review WebSphere Application Server for z/OS requirements for application development and client environments.
21. (Optional) Implement Sysplex Distributor, and set up a high-availability environment. See Setting up a high availability environment for more information.
22. (Optional) Expand your application serving environments as needed.
23. Tune system performance. See Tuning the z/OS operating system and "Skill requirements" on page 22 for more information.

Once you have identified the elements you want to incorporate in your implementation plan, you are ready to install and configure the product.

Directory conventions

References in the product information to *app_server_root*, *profile_root*, and other directories imply specific directory locations. This topic describes the conventions in use for WebSphere Application Server for z/OS.

smpe_root

Refers to the root directory for product code installed with SMP/E.

The corresponding product variable is smpe.install.root.

The default is /usr/lpp/zWebSphere/V6R1.

configuration_root

Refers to the mount point for the configuration file system (formerly, the configuration HFS) in WebSphere Application Server for z/OS.

The configuration_root contains the various app_server_root directories and certain symbolic links associated with them. Each different configuration_root normally requires its own cataloged procedures under z/OS.

The default is /WebSphere/V6R1.

app_server_root

Refers to the top directory for a WebSphere Application Server node.

The node may be of any type—application server, deployment manager, or unmanaged for example. Each node has its own app_server_root. Before Version 6.0 of the product information, this was referred to as the ″WAS_HOME″ directory. Corresponding product variables are was.install.root and WAS_HOME.

The default varies based on node type. Common defaults are *configuration_root*/Appserver and *configuration_root*/DeploymentManager.

profile_root

Refers to the home directory for a particular instantiated WebSphere Application Server profile.

Corresponding product variables are server.root and user.install.root.

In general, this is the same as *app_server_root*/profiles/*profile_name*. On z/OS, this will be always be *app_server_root*/profiles/default because only the profile name ″default″ is used in WebSphere Application Server for z/OS.

Product version information

The WebSphere Application Server product contains structural differences from previous versions. The properties/version directory in the *app_server_root* contains important data about the product and its installed components, such as the build version and build date. This information is included in WAS.product and [component].component files.

Run the historyInfo command to create a report about installed maintenance packages. The historyInfo command creates a report on the console and also creates tracking files in the *app_server_root*/properties/version/history directory.

Time-stamped, detailed logs record each update process in the properties/version/log directory of the *app_server_root*.

This topic describes the XML data files that store product information for Version 6 WebSphere Application Server products. By default, the document type declarations (DTDs) for these files are in the properties/version/dtd folder of the *app_server_root*, or the server root directory. See the “Product version information” section for more information.

This topic includes the following sections:

- “Product information files”
- “Reports” on page 26
- “Logs and component backups” on page 26
- “Directory locations” on page 27
- “Operational description” on page 28
- “Data dictionary” on page 28

Product information files

XML files in the properties/version directory that store version information:

platform.websphere

One file whose existence indicates that a WebSphere Application Server product is installed. An example of the file follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE websphere PUBLIC "websphereId" "websphere.dtd">
<websphere name="IBM WebSphere Application Server" version="6.1"/>
```

The following XML files in the properties/version directory represent installed items and installation events such as product edition, version, component, and build information.

WAS.product

One file whose existence indicates the particular WebSphere Application Server product that is installed. The type of product installed is indicated by the <id> tag. Data in the file indicates the version, build date, and build level.

For example, <id>ND</id>.product indicates that the installed product is WebSphere Application Server Network Deployment. An example of the file follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE product PUBLIC "productId" "product.dtd">
<product name="IBM WebSphere Application Server - ND">
  <id>ND</id>
  <version>6.1.0</version>
  <build-info date="02/03/06" level="s0461.18"/>
</product>
```

*component-name***.component**

Any number of component files that each indicate the presence of an installed component, which is part of the product. Data in the file indicates the component build date, build version, component name, and product version. For example, the file might be the activity.component file, which indicates that the activity component is installed. The activity component is part of the Network Deployment product. An example of the file follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE component PUBLIC "componentId" "component.dtd">
<component build-date="05/31/06" build-version="a0522.07"
   name="activity" spec-version="6.1.0.0"/>
```

*extension.id***.extension**

Any number of extension files that each indicate the presence of an extension that you install as a user extension, as part of a service engagement, or as installed by a third party product. The *<extension.id>*.extension files are not created, logged, or removed by WebSphere Application Server products.

*ptf-id***.ptf**

Any number of maintenance package files that each indicate the presence of an installed refresh pack, fix pack, or interim fix.

XML files in the properties/version/history directory that store version history information files: The following XML files in the properties/version/history directory describe refresh packs, fix packs, and interim fixes that are currently installed. These XML files are related to installation items by the primary ID information, which is shown in the following examples as italicized text.

xxxmaintenance_package_identifierxxx.**ptfDriver**

A driver file has useful information about the entire contents of an interim fix or fix pack.

xxxmaintenance_package_identifierxxx.**ptfApplied**

The applied file has relevant information about the interim fixes or fix packs that are currently applied.

event.history

One file that lists update events that have occurred. An update event is an operation that installs or uninstalls an interim fix or a fix pack. The file is sorted by the date and time of the events that are listed.

Reports

WebSphere Application Server provides the ability to generate ***Version*** reports and History reports from the data in the files. The following report-generation scripts are available in the *app_server_root* bin directory.

Product version reports

The following report generation scripts extract data from XML data files in the properties/version folder:

- versionInfo script

  Lets you use parameters to create a version report on Linux and UNIX platforms, or on Windows platforms.
- genVersionReport script

  Generates the versionReport.html report file in the bin directory on Linux and UNIX platforms, or on Windows platforms. The report includes the list of components and installed and uninstalled maintenance packages.

Product history reports

The following report generation scripts extract data from XML data files in the properties/version/history folder:

- historyInfo script

  Lets you use parameters to create a history report on Linux and UNIX platforms, or on Windows platforms.
- genHistoryReport script

  Generates the historyReport.html report file in the bin directory on Linux and UNIX platforms, or on Windows platforms. The report includes the list of components and a history of installed and uninstalled maintenance packages.

Logs and component backups

WebSphere Application Server products use two other directories when performing update operations, for logging and backups:

app_server_root **/logs/update**

The logs directory for product updates.

The location of log files that describe events that occur during the use of the update installer program.

*app_server_root***properties/version/backup**

Product updates backup directory

WebSphere Application Server products back up components before applying interim fixes and fix packs. If you uninstall an interim fix or fix pack, WebSphere Application Server products restore the backed-up component JAR file.

File naming convention

Time stamp
YYYYMMDD_HHMMSS

For example: 20050324_211832 is 24-Mar-2004, 9:18:32 pm, GMT. All time stamps are in GMT.

ID
Interim fix ID or fix pack ID

For example: apar6789c is an interim fix ID; PTF_1 is a fix pack ID.

Operation
install | uninstall

Interim fix log file names
timeStamp_fixID_operation.log

For example, the Update installer program creates these logs: *app_server_root*logs/update/20050324_211832_apar6789c_install.log and *app_server_root*/logs/update/20050324_211912_apar6789c_uninstall.log

Interim fix component log file names
timeStamp_fixId_componentName_operation.log

For example, the update installer program creates these logs: *app_server_root*/logs/update/20050324_211832_apar6789c_ras_install.log and *app_server_root*/logs/update/20050324_211912_apar6789c_ras_uninstall.log

Fix pack log file names
timeStamp_ptfId_operation.log

For example, the update installer program creates these logs: *app_server_root*/logs/update/20050924_211832_was60_fp1_install.log and *app_server_root*/logs/update/20050924_211912_was60_fp1_uninstall.log

Fix pack component log file names
timeStamp_ptfId_componentName_operation.log

For example, prior to Fix Pack 2: properties/version/log/20050324_211832_was50_fp1_ras_install.log and properties/version/log/20030325_211912_was50_fp1_ras_uninstall.logThe update installer program creates these logs: *app_server_root*/logs/update/20050324_211832_was60_fp1_ras_install.log and *app_server_root*/logs/update/20030325_211912_was60_fp1_ras_uninstall.log

Backup JAR file names
*timeStamp_ptfId_componentName*_undo.jar or *timeStamp_fixId_componentName*_undo.jar

For example: 20020924_211832_apar6789c_ras_undo.jarDo not delete a backup JAR file. You cannot remove a component update if the corresponding backup JAR file is not present.

Update processing might also use a temporary directory if necessary. A Java property specifies this directory as described in the next section.

Directory locations

Product information files are located relative to the WebSphere Application Server product *app_server_root*, or the server root directory.

Default file paths are:

Version directory
app_server_root/properties/version

History directory
app_server_root/properties/version/history

Updates log directory
The update installer program stores log files in the *app_server_root*/logs/update directory.

Updates backup directory
app_server_root/properties/version/backup

DTD directory
app_server_root/properties/version/dtd

Temporary directory
Specified by the java.io.tmpdir Java system property

Operational description

WebSphere Application Server products update the product version history information while performing events that install or uninstall fixes or fix packs. Events that might occur include:

- A WebSphere Application Server product removes an interim fix file from the version directory when it uninstalls the corresponding fix.
- A WebSphere Application Server product adds a file with an extension of .ptf to the version directory to indicate than a refresh pack, a fix pack, or an interim fix is currently installed.
- A WebSphere Application Server product removes a file with an extension of .ptf from the version directory when it uninstalls the corresponding refresh pack, a fix pack, or an interim fix.
- A WebSphere Application Server product adds a driver file with an extension of .ptfDriver to the version/history directory when you run the historyInfo command. A fix pack driver file contains defining information for a fix pack.
- A WebSphere Application Server product adds a fix pack applied file with an extension of .ptfApplied to the version/history directory when when you run the historyInfo command. A fix pack application file contains information that identifies component updates that have been applied for a fix pack. The application file also provides links to component log and backup files.
- A WebSphere Application Server product makes entries in the history file, event.history, when it installs or uninstalls a maintenance package.
- A WebSphere Application Server product writes a line about a parent event for each refresh pack, a fix pack, or interim fix that it installs or uninstalls.
- A WebSphere Application Server product stores child component events for each component update that it installs or uninstalls, beneath the corresponding interim fix, fix pack, or refresh pack parent event.
- A WebSphere Application Server product stores one log file in the logs/update directory as it installs or uninstalls one interim fix, fix pack, or refresh pack.
- A WebSphere Application Server product stores one log file in the logs/update directory as it installs or uninstalls an interim fix, fix pack, or refresh pack in response to each component update that occurs.
- A WebSphere Application Server product stores a component backup file in the backup directory for each component update that it installs.
- A WebSphere Application Server product removes a component backup file from the backup directory for each component update that it uninstalls.

Data dictionary

Type Family: WebSphere product family

File Types:
websphere

File Type:
websphere

Elements:

```
name                 string     required
version              string     required
```

Persistence:
versionDir/platform.websphere

Type Detail:

The websphere file denotes the presence of WebSphere family products.

Element Detail:

```
websphere.name           The WebSphere product family name.
websphere.version        The WebSphere product family version.
```

```
Type Family:      product

File Types:       product
                  component
                  extension

File Type:        product

Persistence:      versionDir/WAS.product

Elements:         id                    string     required
                  name                  string     required
                  version               string     required
                  build-info            complex    required

Type Detail:

A product file is placed to denote the presence of a specific
WebSphere family product.
The product ID is embedded in the product file name.

Element Detail:

product.id               The id of the product.
product.name             The name of the product.
product.version          The version of the product.
product.build-info       An element containing build information for
                         the product.

Element Type:     build-info

Elements:         date                  date       required
                  level                 string     required

Type Detail:

A build-info instance details the build of a specific installed
WebSphere family product.

Element Detail:

build-info.date          The date on which the product was build.
build-info.level         The level code of the product's build.

File Type:        component

Persistence:      versionDir/name.component

Elements:         name                  string     required
                  spec-version          string     required
                  build-version         string     required
                  build-date            date       required

File Detail:

A component file denotes the presence of a specific component.
The component name is embedded in the component file name.

Element Detail:

component.name           The name of the component.
component.spec-version   The specification version of the component.
component.build-version  The build level of the component.
component.build-date     The build date of the component.
```

```
Type Family:      update

File Types:       ptf
                  ptf-applied

File Type:        ptf

Persistence:      versionDir/id.ptf

Elements:         id                   string     required
                  short-description    string     required
                  build-version        string     required
                  build-date           date       required
                  component-name       complex    min=1, max=unbounded

Type Detail:

A ptf file denotes the presence of some portion of a specific refresh pack,
fix pack, or interim fix.

The id of the refresh pack, fix pack, or interim fix is embedded in the fix
pack file name.

A ptf file contains a listing of component updates.

When installing a refresh pack, fix pack, or interim fix, you can omit certain
potential component updates, but only when the corresponding component is not
installed.

Examine a separate application file to determine the components that
a particular refresh pack, fix pack, or interim fix updates.

A refresh pack or fix pack can include updates for a number of interim fixes.


Element Detail:

ptf.id                   The ID of the fix pack.

ptf.short-description    A short description of the fix pack.

ptf.build-version        The build version of the fix pack.
                         This is distinct from the build version of
                         component updates contained within the fix pack.

ptf-build-date           The build date of the fix pack.  This is distinct
                         from the build version of the component updates
                         contained within the
                         fix pack.

ptf.component-name       A list of components.



File Type:        ptf-applied

Persistence:      versionDir/id.ptfApplied

Elements:         ptf-id               string     required
                  component-applied    complex    min=0, max=unbounded

Type Detail:

A ptf-applied collection specified what components have been updated for
the refresh pack, fix pack, or interim fix as specified by the ID.

Element Detail:
```

```
ptf-applied.ptf-id         The ID of the refresh pack, fix pack, or interim fix
                           for which applieds are recorded.

ptf-applied.component-applied  The list of recorded applications.

Element Type:     component-applied

Elements:         component-name        string     required
                  update-type           enum       required [enumUpdateType]
                  log-name              anyURL     required
                  backup-name           anyURL     required
                  time-stamp            date       required

Type Detail:

An applied instance is present to indicate the application of an
update for a particular interim fix, fix pack, or refresh pack to a
particular component.

(The particular interim fix, fix pack, or refresh pack is specified by
the applied parent.)  An applied provides sufficient information
to undo itself.

The elements of an applied are copies of values from update
events.

Element Detail:

component-applied.component-name   The name of the component which was updated.

component-applied.update-type      The type of the component update.

component-applied.log-name         The name of the log file that was generated by
                                   this application.

component-applied.backup-name      The name of the backup file that was generated by
                                   this application.

component-applied.time-stamp       The time of this application (the ending time of
                                   the corresponding update event).



Enum Type:        enumUpdateType

Values:           0 add
                  1 replace
                  2 remove
                  3 patch

Type Detail:

An update type instance specifies the type of an update.  An 'add' update adds
a component into an installation.  A 'replace' update replaces a particular
version of a component with a different version of that component.  A 'remove'
update removes a component.  A 'patch' update performs a limited update to a
component, in particular, without changing the version of the component.

When adding a component, that component may not already be present.
When replacing or removing a component, that component must be present.
When patching a component, that component must be present.

When replacing or removing a component, or when patching a component, usually,
at least one version prerequisite will be specified for the component update.

Value Detail:
```

```
enumUpdateType.add       Specifies that an update adds a component.

enumUpdateType.replace   Specifies that an update replaces a component.

enumUpdateType.remove    Specifies that an update removes a component.

enumUpdateType.patch     Specifies that an update modifies a component, but
                         does not change its version.


Type Family:      history

File Type:        event-history

Persistence:      historyDir/event.history

Elements:         update-event          complex    min=0, max=unbounded

Type Detail:

One event history is provided for a websphere product family installation.
This event history contains history of update events, corresponding with
the actual update events for that product family.

Element Detail:

event-history.update-event  The list of update events for the websphere
                            product family.  The top level events are refresh pack,
                            fix pack, and interim fix events, each containing one
                            or more component events.

Element Type:     update-event

Elements:         event-type            enum       required [enumEventType]
                  parent-id             string     required
                  id                    string     required
                  update-type           enum       required [enumUpdateType]
                  primary-content       anyURI     required
                  update-action         enum       required [enumEventAction]
                  log-name              anyURI     required
                  backup-name           anyURI     required
                  start-time-stamp      dateTime   required
                  result                string     required
                  update-event          complex    optional

Type Detail:

An update event denotes a single update action, applying to either a
fix, a fix pack, a refresh pack, or a component, according to the set
event type.

Element Detail:

update-event.event-type        The type of this event, either a refresh pack, fix
                               pack, or an interim fix type event, or a component
                               type event.

update-event.parent-id         This element is present only for component
                               events. The ID of the parent interim fix, fix
                               pack, or refresh pack of this event.

update-event.id                The ID of the interim fix, fix pack, refresh pack,
                               or component that was updated, interpreted according
                               to the type of the event.

update-event.update-type       The type of update for an update event.
```

```
update-event.update-action     The type of action for this event.

update-event.log-name          The name of the log file that was generated
                               for this event.

update-event.backup-name       The name of the backup file that was generated
                               for this event.

update-event.start-time-stamp  The XML timestamp of the starting time of the
                               event.  This timestamp follows the XML timestamp
                               format, meaning that time zone information is
                               included.

update-event.result            The result of the update.

update-event.update-event      A collection of child events.  This collection is
                               used for interim fix and fix pack type events.
                               This collection is empty for component type events.


Type Detail:

An event type instance specifies the type of an update event, which is either
a refresh pack, fix pack, or interim fix (ptf) event or a component event. The
interpretation of particular event elements depends on the set event type.

Value Detail:

EventType.ptf        Specifies that an event is for a refresh pack, fix pack,
                     or interim fix update.

EventType.component  Specifies that an event is for a component update.

Enum Type:        update-action

Values:           0 Install
                  1 Uninstall

Type Detail:

An event action instance specified the operation performed by an update, which
can be an install or uninstall operation.


Value Detail:

enumEventAction.install              Specifies that an event is an install
                                     operation.

enumEventAction.uninstall            Specifies that an event is an uninstall
                                     operation.

Enum Type:        enumUpdateType

Values:           0 Add
                  1 Replace
                  2 Remove
                  3 Patch

Type Detail:

An update type instance specifies the type of a component update.

An 'add' update adds a component into an installation.

A 'replace' update replaces a particular version of a component with
```

```
a different version of that component.

A 'remove' update removes a component.

A 'patch' update performs a limited update to a component,
in particular, without changing the version of the component.

To add a new component, the component must not exist.
To replace or remove a component, the component must exist.
To patch a component, the component must exist.

When replacing or removing a component, or when patching a component,
usually, at least one version prerequisite is specified for the
component update.

Value Detail:

enumUpdateType.add       Specifies that an update adds a component.

enumUpdateType.replace   Specifies that an update replaces a component.

enumUpdateType.remove    Specifies that an update removes a component.

enumUpdateType.patch     Specifies that an update modifies a component, but
                         does not change its version.

Enum Type:        enumEventResult

Values:           0 Succeeded
                  1 Failed
                  2 Cancelled

Type Detail:

An event result instance denotes a particular result for an update event.
The result indicates success, failure, or cancellation.

Value Detail:

enumEventResult.succeeded  Specifies that the operation was successful.

enumEventResult.failed     Specifies that the operation failed.

enumEventResult.cancelled  Specifies that the operation was cancelled.
```

Installation: Resources for learning

Use the following links to find relevant supplemental information about installation and customization. The information resides on IBM and non-IBM Internet sites, whose sponsors control the technical accuracy of the information. The information resides on IBM and non-IBM Internet sites, whose sponsors control the technical accuracy of the information.

These links are provided for convenience. Often, the information is not specific to the IBM WebSphere Application Server product, but is useful in all or in part for understanding the product. When possible, links are provided to technical papers and Redbooks that supplement the broad coverage of the release documentation with in-depth examinations of particular product areas.

One important link is:

How to buy WebSphere Application Server software

This IBM Web site describes pricing and technical details. If you have already purchased the software, view links to additional information about:

- Planning, business scenarios, and IT architecture

- Programming instructions and examples
- Programming specifications
- Administration
- Support

Planning, business scenarios, and IT architecture

- Supported hardware and software

  The official site for determining product prerequisites for hardware and software for all WebSphere Application Server products.
- IBM developerWorks WebSphere

  The home of technical information for developers working with WebSphere products. You can download WebSphere software, take a fast path to developerWorks zones, such as VisualAge Java or WebSphere Application Server, learn about WebSphere products through a newcomers page, tutorials, technology previews, training, and Redbooks, get answers to questions about WebSphere products, and join the WebSphere community, where you can keep up with the latest developments and technical papers.
- IBM WebSphere Application Server library and information centers Web site

  The IBM WebSphere Application Server Library Web site contains links to all WebSphere Application Server information centers, for all versions. It also lets you access each information center in your native language.
- IBM WebSphere Application Server home page

  The IBM WebSphere Application Server home page contains useful information, including support links and downloads for maintenance packages, APARs, tools, and trials.
- IBM WebSphere software platform home page

  The IBM WebSphere software platform home page introduces WebSphere products and describes how companies can easily transform to an e-business, with software that can grow as fast as the business it supports.
- WebSphere Application Server Edge components library and information centers Web site

  The information center for WebSphere Application Server Edge components contains complete documentation for the Caching Proxy and the Load Balancer.

Programming instructions and examples

- IBM developerWorks

  IBM developerWorks contains many excellent resources for developers, including tutorials on Web development-related topics. There is an excellent tutorial on the JDBC API.
- IBM Redbooks

  The IBM Redbooks site contains many documents that are related to WebSphere Application Server.

Programming specifications

- J2EE information

  For more information about J2EE specifications, visit the Sun site.

Administration

- WebSphere technical library on developerWorks

  The WebSphere library includes a wide range of content, including technical articles, downloads, product documentation, and tutorials
- The IBM Terminology Web site

  The IBM Terminology Web site consolidates the terminology from many IBM products in one convenient location. In addition to base computer terminology, terms and definitions from IBM brands and product families are included and explained.

Support

- Steps to getting support for WebSphere Application Server

Whether you are a new user looking for basic information, or an experienced user looking for a specific workaround, you can benefit immediately from IBM's extensive Web-based support. Download fixes, search on keywords, find how-to information, and possibly solve a problem -- all before contacting IBM Software Support directly.

- WebSphere Application Server for z/OS Support page

  Take advantage of the Web-based Support and Service resources from WebSphere Application Server to quickly find answers to your technical questions. Easily access the latest downloads, find workarounds to technical problems, or register to receive e-mail from IBM Support.
- IBM e-server Support: Fix Central

  A Web facility for downloading fixes for hardware and operating systems, including z/OS and i5/OS.
- Adobe Acrobat Web site

  This Adobe Web site offers a free download of the Adobe Acrobat Reader product.

Planning for installation

Use this task to prepare to install WebSphere Application Server for z/OS product code.

Print and review "Steps for creating your implementation plan" on page 22. Assemble a team of people to install and configure the product. Be sure that the team has the skills needed to plan, install, and configure WebSphere Application Server for z/OS. See "Skill requirements" on page 22 for more information.

You must completely install the product code for WebSphere Application Server for z/OS before you can set up an application serving environment. Obtain product code from IBM in one of the following formats:

- An **IBM ServerPac or SystemPac**, which consists of loadable product libraries and corresponding SMP/E data sets. A ServerPac or SystemPac contains program libraries with integrated maintenance for one or more products, which include a base operating system, such as z/OS, if desired. Program library data sets are loaded to disk as part of the ServerPac or SystemPac installation. After installation, perform maintenance with SMP/E.
- An **IBM Custom-Build Product Delivery Option (CBPDO)**, which consists of SMP/E relfiles. A CBPDO contains SMP/E relfiles and maintenance for one or more products. Install each product using SMP/E commands (APPLY / ACCEPT) or the corresponding panels. After installation, perform maintenance with SMP/E.

Perform the tasks in this section before you install the WebSphere Application Server for z/OS product code. By planning your product code placement and naming, you can ease future product maintenance and migration tasks.

1. Make sure that the z/OS system on which you will install WebSphere Application Server for z/OS meets the hardware and software requirements. See "Driving system requirements" on page 9 for more information.
2. Identify the software delivery option you will use. To review your product delivery options see "IBM SystemPac or ServerPac" on page 40 or "IBM Custom-Build Product Delivery Offering" on page 41 for more information.
3. Learn about WebSphere Application Server product data sets, and plan a naming convention. See "Product data sets" on page 37 for more information.
4. Learn about WebSphere Application Server product directories, and plan a mount point convention. See "Product file system" on page 39 for more information.
5. Review "Installation: Resources for learning" on page 34 for reference information you might need during installation.

When you have finished the planning process, you are ready to begin installing the product. See "Installing the product and additional software" on page 40.

Product data sets

This article describes the WebSphere Application Server for z/OS product data sets and recommends a product data set naming convention.

WebSphere Application Server for z/OS product code resides in MVS™ partitioned data sets (which contain the product data sets) and MVS hierarchical file system directories (which contain the product directory and its subdirectories). The default high-level qualifier for the product data sets is BBO.

Note: In this and subsequent articles, *was_hlq* is used to represent the high-level data set name qualifier for a particular set of WebSphere Application Server for z/OS product data sets.

Product data set contents

The WebSphere Application Server for z/OS product data sets are divided into target data sets (used during product customization and execution) and distribution libraries (used to ″back off″ maintenance if necessary).

The WebSphere Application Server for z/OS target libraries are as follows:

was_hlq.**SBBGLOAD**
Load module library for future use

was_hlq.**SBBOCLIB**
REXX scripts used by the Customization Dialog

was_hlq.**SBBODSD**
Side decks

was_hlq.**SBBOEXEC**
CLIST scripts

was_hlq.**SBBOJCL**
JCL for installation jobs

was_hlq.**SBBOLD2**
Load modules that must NOT be placed in LPA; can be in link list

was_hlq.**SBBOLOAD**
Load modules that can be placed in LPA or link list

was_hlq.**SBBOLPA**
Load modules that must be in LPA or loaded in ECSA by the location service daemon

was_hlq.**SBBOMAC**
Assembler macros

was_hlq.**SBBOMIG**
Load modules for use with IPCS dump formatting

was_hlq.**SBBOMJPN**
Customization Dialog message library (Japanese)

was_hlq.**SBBOMLIB**
Customization Dialog message library (English)

was_hlq.**SBBOMSG**
Message translation skeletons

was_hlq.**SBBOPJPN**
Customization Dialog panel library (Japanese)

was_hlq.**SBBOPLIB**
Customization Dialog panel library (English)

was_hlq.**SBBOSLB2**
Customization Dialog skeleton library of variable-length record files

was_hlq.**SBBOSLIB**
Customization Dialog skeleton library of fixed length, LRECL=80 files

The WebSphere Application Server for z/OS distribution libraries are as follows:

was_hlq.**ABBOANT**
HFS files

was_hlq.**ABBOCLIB**
REXX scripts used by the Customization Dialog

was_hlq.**ABBOEBCD**
HFS files (EBCDIC)

was_hlq.**ABBOEXEC**
CLISTs used by the Customization Dialog

was_hlq.**ABBOEXP**
Export files

was_hlq.**ABBOINC**
Include files

was_hlq.**ABBOJAR**
JAR files

was_hlq.**ABBOJCL**
JCL for installation jobs

was_hlq.**ABBOMAC**
Assembler macros

was_hlq.**ABBOMJPN**
Customization Dialog message library (Japanese)

was_hlq.**ABBOMLIB**
Customization Dialog message library (English)

was_hlq.**ABBOMSG**
Message translation skeletons

was_hlq.**ABBOOBJ**
Object modules for load module creation

was_hlq.**ABBOPJPN**
Customization Dialog panel library (Japanese)

was_hlq.**ABBOPLIB**
Customization Dialog panel library (English)

was_hlq.**ABBOSLB2**
Customization Dialog skeleton library of variable-length record files

was_hlq.**ABBOSLIB**
Customization Dialog skeleton library of fixed length, LRECL=80 files

See WebSphere Application Server for z/OS: Program Directory (GI11-2825) for allocation information about each target library and distribution library. Updates to this information are included in the Preventive Service Planning (PSP) bucket for each release of WebSphere Application Server for z/OS.

Product data set naming restrictions

The IBM ServerPac and SystemPac® and the IBM Custom-Build Product Delivery Option allow you to rename product data sets during installation. In addition, the IBM ServerPac and SystemPac allow you to merge members from similar target libraries into a single data set. However, keep in mind the following:

- You should not merge SBBOLD2 with any other load module library.
- The Customization Dialog script library (SBBOCLIB), message libraries (SBBOMSG and SBOJPN), panel libraries (SBBOPLIB or SBBOPJPN, and skeleton libraries (SBBOSLIB and SBBOSLIB) are allocated directly by the Customization Dialog startup exec. You should not merge these libraries with any other ISPF libraries, and they should all have the same high-level qualifier as the SBBOEXEC data set that contains the Customization Dialog startup exec.
- Ensure that the Customization Dialog data sets are cataloged.

Product data set naming convention

As noted above, certain WebSphere Application Server for z/OS data sets must have the same high-level data set name qualifier in order for the product to function correctly. Product maintenance and migration is easier if all product data sets have the same high-level qualifier.

On the other hand, in order to continue to run WebSphere Application Server for z/OS while applying maintenance, you must have at least two copies of the product data sets: one for the running application execution environment and one to which service is applied.

We recommend you choose a middle level qualifier for each separate release and maintenance level of WebSphere Application Server for z/OS. This middle level qualifier can reflect a very simple test/production distinction, such as with ″BBO.V6PROD.*″ or ″BBO.V6TEST.*″, or can include specific service level information, such as with ″WAS.W600102.*″ or ″WAS.W600103.*″.

There are many places where you must specify the product data set names, so, to avoid undue confusion, use the simplest data set naming scheme that accomplishes your maintenance goals.

Product file system

WebSphere Application Server for z/OS product code resides in MVS-partitioned product data sets and MVS file system (the product directory and its subdirectories). This topic describes the WebSphere Application Server for z/OS product directory.

Product directory

All WebSphere Application Server for z/OS product files reside in the product directory and its subdirectories. The default product directory is /usr/lpp/zWebSphere/V6R1. Throughout the product and documentation, *install_root* is used to represent the fully qualified path name of the WebSphere Application Server for z/OS product directory.

Locate the product directory and all of its subdirectories in the same hierarchical file system (HFS) or zSeries® file system (ZFS) data set. This data set can be the same as the z/OS root or version data set, which is not recommended, or a separate data set that is used just for WebSphere Application Server for z/OS. The installation jobs and program directory assume that such a separate data set is allocated. This data set is referred to as *was_hlq*.SBBOHFS, where *was_hlq* represents the product data set name high-level qualifiers. This directory gets created during the installation process. Refer to the Program Directory, http://www-306.ibm.com/software/webservers/appserv/was/library/ for more details.

Product directory and configuration directory

Each WebSphere Application Server for z/OS application serving environment (stand-alone application server node or Network Deployment cell) has configuration files in one or more WebSphere configuration directories. These configuration directories are created through the configuration process and contain symbolic links to files in the product directory.

Installing the product and additional software

Use this task to install WebSphere Application Server for z/OS product code.

Complete the steps in “Planning for installation” on page 36.

The product code for WebSphere Application Server for z/OS is installed using either an IBM ServerPac/SystemPac or an IBM Custom-Built Product Delivery Option (CBPDO). This section of the documentation provides guidance on using these two vehicles to install the WebSphere Application Server for z/OS product.

Perform the tasks in this section to install the WebSphere Application Server for z/OS product code.

1. Order an IBM ServerPac/SystemPac or IBM CBPDO that contains the appropriate WebSphere Application Server for z/OS product.
2. Follow the instructions for the delivery vehicle that you choose:
   - “IBM SystemPac or ServerPac”
   - “IBM Custom-Build Product Delivery Offering” on page 41
3. File the installation materials for later use during product maintenance.

When you have finished the installation process, you are ready to prepare your target systems for WebSphere Application Server for z/OS. See “Preparing the base operating system” on page 41 for more information.

IBM SystemPac or ServerPac

An IBM CustomPac (SystemPac, ServerPac or ProductPac) is a set of preloaded product data sets bundled with an IBM dialog that is used to load the data sets to disk and perform initial customization. In general, SMP/E work is not required during installation of a CustomPac offering. Instead, SMP/E data sets that correspond to the CustomPac service level are loaded onto the disk along with the product data sets. You can still use SMP/E to install preventive and corrective service after CustomPac installation.

If you use an IBM SystemPac or ServerPac, follow the instructions in the copy of *ServerPac: Installing your Order* that ships with your SystemPac or ServerPac.

See *ServerPac: Using the Installation Dialog* (SA22-7815) for information about the ISPF dialog used to install a SystemPac or ServerPac.

Notes:

- Be sure to choose a product data set naming convention that allows you to keep and maintain at least two copies of product libraries for maintenance purposes.
- If you are installing from a driving system, make sure that the maintenance level of the target system meets requirements for WebSphere Application Server for z/OS.
- When installation is complete, make sure that the product data sets are available to your z/OS target systems and the product code HFS is mounted at /usr/lpp/zWebSphere/V6R1 or a similar mount point of your choice on each target system.

For further information, see the following:

- eSupport Web site at http://www.ibm.com/software/webservers/appserv/zos_os390/support/
- PSP buckets
- IBM Software Support Center.

IBM Custom-Build Product Delivery Offering

An IBM Custom-Build Product Delivery Offering (CBPDO) is a set of product tapes for one or more IBM software products that is bundled with cumulative service. Install the products and service on your system using SMP/E.

If you use CBPDO, follow the instructions in the copy of *WebSphere Application Server for z/OS: Program Directory* that ships with your order.

Notes:

- Be sure to choose a product data set naming convention that allows you to keep and maintain at least two copies of product libraries for maintenance purposes.
- If you are installing from a driving system, make sure that the maintenance level of the target system meets requirements for WebSphere Application Server for z/OS.
- When installation is complete, make sure that the product data sets are available to your z/OS target systems and the product code HFS is mounted at /usr/lpp/zWebSphere/V6R1 or a similar mount point of your choice on each target system.

For further information, see the following:

- eSupport Web site at http://www.ibm.com/software/webservers/appserv/zos_os390/support/
- PSP buckets
- IBM Software Support Center.

Preparing the base operating system

Use this task to prepare your z/OS target systems for WebSphere Application Server for z/OS.

- Complete the steps in “Installing the product and additional software” on page 40.
- Identify the z/OS systems on which you plan to run WebSphere Application Server for z/OS.

The WebSphere Application Server for z/OS product makes extensive use of the underlying z/OS operating system services for security, reliability, and performance.

After you install the WebSphere Application Server for z/OS product code, perform the tasks in this section to prepare your z/OS target systems to run WebSphere Application Server for z/OS.

Note: Target systems are the systems on which WebSphere Application Server for z/OS will actually run. The driving system, on which the WebSphere Application Server for z/OS product code installation is performed, might or might not also be a target system.

1. Identify the first z/OS target system on which you plan to create a WebSphere Application Server for z/OS application serving environment.
2. Print off a copy of “Checklist: Preparing the base operating system” on page 50. Use this worksheet to identify which of the following steps have been completed for the target system and record information you will need during product configuration.
3. Prepare z/OS operating system settings. See “Preparing z/OS” on page 42 for detailed instructions.
4. Prepare z/OS sysplex settings. See “Preparing the sysplex” on page 45 for detailed instructions.
5. Prepare the z/OS job entry subsystem (JES2 or JES3). See “Preparing JES2 or JES3” on page 46 for detailed instructions.
6. Identify TCP/IP resources you want to use and prepare your network. See “Preparing TCP/IP” on page 48 for more information.
7. Set up Resource Recovery Services (RRS). See “Preparing Resource Recovery Services” on page 46 for more information.

8. Set up your SAF-compliant security package. See "Preparing the security server (RACF)" on page 48 if you will use the z/OS Security Server. If you will use another SAF-compliant security product, consult the product's manufacturer for assistance.
9. Set up component trace (CTRACE). See Setting up component trace (CTRACE).
10. Repeat these steps for each z/OS target system on which you plan to run WebSphere Application Server for z/OS.
11. Keep the worksheets you filled out as you will need some of the information you recorded on them during product configuration.

When you have completed this task for each z/OS target system, you are ready to plan your WebSphere Application Server for z/OS application serving environments on these target systems. See "Planning for product configuration" on page 51 for more information.

Preparing z/OS

Follow these steps to prepare your z/OS operating system to run WebSphere Application Server for z/OS.

1. Make sure that all software prerequisites listed in "Target system requirements" on page 10 are met.
2. Make sure that the UNIX System Services environment is active and that the BPXPRMxx settings in effect meet or exceed the following minimum values:

   MAXTHREADS: 10000
   MAXTHREADTASKS: 5000
   MAXFILEPROC: 10000
   MAXSOCKETS (AF_INET domain): 12000
3. If you use localization and alternate code pages with UNIX System Services, make sure that all WebSphere Application Server for z/OS server, administrator and client user IDs (any user IDs that run WebSphere Application Server for z/OS scripts) are run with environment variables LANG and LC_ALL both set to the same locale based on code page IBM-1047. Settings based on any other code page can cause the scripts to fail. See ″Changing the Locale in the Shell″ in *UNIX System Services User's Guide* for more information.
4. Make sure that the /tmp directory has at least 20 megabytes of free space.

   WebSphere Application Server for z/OS makes extensive use of the /tmp directory.

   You can use the `df -kP /tmp` shell command to show the number of available 1K blocks in the /tmp directory HFS. Divide the number of available 1K blocks by 1024 to determine the number of megabytes of free space.

   If your /tmp directory resides in a permanent read-write HFS, use the confighfs command in /usr/lpp/dfsms/bin to extend it as necessary. For example, the following command will add an additional 10 MB of space to the HFS in which /tmp resides:

   ```
   /usr/lpp/dfsms/bin/confighfs -x 10m /tmp
   ```

   If your /tmp directory resides in a temporary file system (TFS), modify the MOUNT statement in BPXPRMxx that defines it to add additional space. To define a 20 MB TFS and mount it at /tmp, for example, use the following MOUNT command:

   ```
   MOUNT FILESYSTEM('/TMP') TYPE(TFS) MOUNTPOINT('/tmp') PARM('-s 20')
   ```

   Note: If you do not specify a space ('-s') value, then the undesirably small default of 1 megabyte will be used.
5. Determine the full data set names of the following system data sets used by WebSphere Application Server for z/OS:

   SCEERUN
   Language Environment runtime library

   SCEERUN2
   Language Environment runtime library

System SSL
System SSL runtime library (SGSKLOAD for z/OS Version 1 Release 5 and below, SIEALNKE for z/OS Version 1 Release 6 and above)

Also, determine whether these data sets are in the system link pack area (LPA) or link list. Record this information on the worksheet.

Note: If you are configuring WebSphere Application Server for z/OS on a collection of z/OS systems, with some systems at z/OS Release 5 or earlier and others at z/OS Release 6 or later, you might need to manually edit some of the cataloged procedures created by the customization process. See “Link pack area, link list, and STEPLIB” for more information.

6. Make sure that all the following data sets are APF authorized:
   - cee_hlq.SCEERUN
   - cee_hlq.SCEERUN2
   - gsk_hlq.SGSKLOAD (on all systems at z/OS Version 1 Release 5 and below)
   - sys_hlq.SIEALNKE (on all systems at z/OS Version 1 Release 6 and above)
   - was_hlq.SBBOLPA
   - was_hlq.SBBOLOAD
   - was_hlq.SBBGLOAD
   - was_hlq.SBBOLD2
7. Decide whether to run WebSphere Application Server for z/OS from the system link pack area and link list or from STEPLIB. See “Link pack area, link list, and STEPLIB” for more information.
8. Make sure that any IEFUSI or JES2/JES3 exits on your system do not restrict WebSphere Application Server for z/OS address spaces to an address space size of less than 512 MB. Each WebSphere Application Server for z/OS address space should have a region size of at least 512 MB. All WebSphere Application Server for z/OS cataloged procedures are shipped with a default of REGION=0M.
9. Make sure that the TSO segment default region size for WebSphere Application Server for z/OS installer and administrator TSO user IDs is at least 128 MB.

Note: If you configured WebSphere Application Server for z/OS using the English ISPF Customization Dialog, and wish to allow the display of Japanese characters correctly in your environment, you need to modify some script files.

1. Edit setupCmdLine.sh file

   from: CONSOLE_ENCODING=″-Dws.input.encoding=cp1407 -Dws.output.encoding=cp1407″

   to: CONSOLE_ENCODING=″-Dws.input.encoding=cp1399 -Dws.output.encoding=cp1399″
2. Edit wsadmin.sh file

   from: EXTRA_D_ARGS=″-Dfile.encoding=ISO8859-1

   to: EXTRA_D_ARGS=″-Dfile.encoding=IBM-932

Making these two changes will enable you to see the Japanese messages correctly.

Link pack area, link list, and STEPLIB

By placing load module data sets in the link pack area (LPA) and system link list, you can obtain improved MVS system performance and some virtual storage savings.

Product data sets

You can place only one set of WebSphere Application Server for z/OS load modules (any release) in the link pack area and link list because the same load module names are used across releases. For each application serving environment, the Customization Dialog or Profile Management tool is used to specify whether the environment’s product data sets are already in the link pack area and link list or require loading from STEPLIB.

If load modules for a previous version of WebSphere Application Server for z/OS are in the system link pack area and link list, you must run the new version of WebSphere Application Server for z/OS from STEPLIB. If no previous version is in the system link pack area and link list, then you can choose a set of WebSphere Application Server for z/OS product libraries and place them in the link pack area and link list.

Recommendations:

- Choose the WebSphere Application Server for z/OS product data sets that your production application serving environment will use.
- Plan to load SBBOLOAD into the system link pack area. This will require approximately 8K of common storage below the 16M line and between 20M and 40M of extended common storage. (In version 6 release 1, only the CTRACE support module remains in SBBOLPA.)
- Plan to load SBBOLD2 into the system link list. You cannot place this data set in the link pack area.

System data sets

The WebSphere Application Server for z/OS customization process assumes that the following data sets are in the system link list:

- SCEERUN
- SCEERUN2
- System SSL (SGSKLOAD or SIEALNKE)

If these data sets are not in the link list, be sure to follow the optional customization steps that direct you to add these data sets to STEPLIB concatenations in WebSphere Application Server JCL, and in the setupCmdLine.sh script. Certain Language Environment® load modules can be placed in the link pack area, providing performance improvements. For more information see the Language Environment Customization manual for your z/OS release.

STEPLIB statement in setupCmdLine.sh

Each WebSphere Application Server for z/OS node home directory uses a setupCmdLine.sh shell script to set environment variables during the execution of administrative scripts. One of the functions of this shell script is to set the STEPLIB variable, which provides STEPLIB capability when running a file system executable file.

During initial node configuration, the Customization Dialog sets the value of the STEPLIB variable in setupCmdLine.sh to the same list of data sets as the STEPLIB concatenation in the node's cataloged procedures. You might need to update this shell script (which resides in *profile_root* /bin/ for each node) for a variety of reasons. Update the shell script if you do any of the following:

- Change which WebSphere Application Server for z/OS product code level is in the system link pack area and link list.
- Apply maintenance and need to change which WebSphere Application Server for z/OS product code level a cell is using.

  Note: All nodes in a cell on the same system must use the same product code level.
- Need to add additional STEPLIB data sets. (For example, for DB2 or CICS interfaces.)

To update STEPLIB in setupCmdLine.sh, edit the copy of the script in each affected *profile_root* /bin/ directory and find lines like the following:

```
STEPLIB='WAS.V60.TEST.SBBOLD2':$STEPLIB
STEPLIB='WAS.V60.TEST.SBBOLOAD':$STEPLIB
export STEPLIB
```

Add or remove lines from this concatenation as needed. Be sure to update each node's setupCmdLine.sh script, including the one for the deployment manager.

Switching between LPA/LNKLST and STEPLIB

If you decide to move a particular level of WebSphere Application Server for z/OS product code (previously run from STEPLIB) into the system link pack area and link list, perform the following steps:

1. Identify the application serving environments making use of these product libraries. (That is, the ones that have cataloged procedure STEPLIB statements for these particular SBBOLPA, SBBOLOAD, and SBBOLD2 data sets.)
2. Make sure that no other version of WebSphere Application Server for z/OS is in either the link pack area or link list.
3. Place SBBOLOAD in either the link pack area or link list, and place SBBOLD2 in the link list.
4. Edit all WebSphere Application Server for z/OS cataloged procedures that use these data sets, removing SBBOLPA, SBBOLOAD, and SBBOLD2 from any STEPLIB in which they appear. In the cataloged procedures generated by the Customization Dialog, the proclib members that will require editing are the INCLUDE JCL members that have names ending in ″Z″.
5. For Version 6 Release 1, load a copy of BBORTS61 load module from SBBOLPA into the system link pack area. This load module is required for component trace support; all levels of Version 6 Release 1 can share the same copy of BBORTS61.

If you no longer want to run a particular level of WebSphere Application Server for z/OS product code (formerly in LPA / link list) from STEPLIB instead, perform the following steps:

1. Identify the application serving environments that make use of these product libraries. (That is, the ones that do not have cataloged procedure STEPLIB statements for SBBOLPA, SBBOLOAD, and SBBOLD2.)
2. Edit all WebSphere Application Server for z/OS cataloged procedures for these application serving environments. In the cataloged procedures generated by the Customization Dialog, the proclib members that will require editing are the INCLUDE JCL members that have names ending in ″Z″.
   - Add SBBOLOAD, and SBBOLD2 to the STEPLIB concatenation in all location service daemon cataloged procedures and in all controller and servant cataloged procedures.
3. For each of the affected application serving environments, add SBBOLOAD and SBBOLD2 to the STEPLIB statement in the *profile_root* /bin/ setupCmdLine script in each WebSphere Application Server home directory.
4. Modify IPL procedures so that the WebSphere Application Server for z/OS data sets are no longer in the system link pack area and link list.
5. Ensure that you continue to load a copy of the BBORTS61 load module (release independent) into the system link pack area. You can load this module from any Version 5.x or later SBBOLPA data set.

Preparing the sysplex

WebSphere Application Server for z/OS uses a number of z/OS sysplex services. Therefore, each target system used to run WebSphere Application Server for z/OS must be either a monoplex (single system sysplex) or a member of a sysplex. For more information, see *z/OS MVS Setting Up a Sysplex* (SA22-7625).

Connect systems in a sysplex with channel-to-channel (CTC) communications or through a coupling facility, which is a special logical partition used to share data between sysplex members. Couple data sets on DASD are also used for sysplex coordination.

WebSphere Application Server for z/OS uses the System Logger, an MVS component that allows applications to log data in a sysplex, to log error and trace information and provide XA transaction logging. The System logger creates and manages log streams, which are written first to a coupling facility or local in-memory buffer, then transferred to log data sets on DASD for longer term access. Log streams that are written to local buffers rather than to a coupling facility are called DASD-only log streams.

Follow these steps to prepare your system for a sysplex.

1. Determine if your z/OS target system is already configured as a sysplex.
   a. If so, continue on to the next step.
   b. If not, follow the instructions in *z/OS MVS Setting Up a Sysplex* (SA22-7625) to configure it as a monoplex. Record the sysplex name for later use.
2. Determine if System Logger is already in use on your system.
   a. If so, continue on to the next step.
   b. If not, follow the instructions in the section ″Preparing to Use System Logger Applications″ in *z/OS MVS Setting Up a Sysplex* (SA22-7625).
3. Decide whether WebSphere Application Server for z/OS log streams should reside in a coupling facility or local in-memory buffers. Record the SMS data class, SMS storage class and data set name prefix to be used for log data sets. If WebSphere Application Server for z/OS log streams will reside in a coupling facility, choose the structure name to be used.

Preparing JES2 or JES3

WebSphere Application Server for z/OS uses job entry subsystem (JES2/JES3) services like any other MVS application. Follow these steps to prepare your job entry subsystem to run WebSphere Application Server for z/OS.

1. Identify the cataloged procedure library or libraries (proclibs) that you will use to hold cataloged procedures for WebSphere Application Server for z/OS. You might need to use separate proclibs for each system in a sysplex.
2. If your system uses JES2 EXIT06 or JES3 IATUX03 to control specification of the REGION= value on JOB or EXEC statements, make sure that this control is relaxed for WebSphere Application Server for z/OS address spaces.
3. If you plan to send WebSphere Application Server for z/OS trace output to the JES spool, make sure you have adequate spool space available. WebSphere Application Server for z/OS address spaces can produce a large number of trace records when tracing is activated.

Preparing Resource Recovery Services

WebSphere Application Server for z/OS uses Resource Recovery Services (RRS) to support two-phase transaction commit. This article describes the process of preparing RRS.

Note: RRS must be up and running before WebSphere Application Server for z/OS servers are started. See *z/OS MVS Programming: Resource Recovery* (SA22-7616) for more information.

Normally, all systems in a sysplex share a common set of RRS logs for syncpoint processing. If you want to associate specific systems in a sysplex for syncpoint processing, you can specify a log group name when you start RRS. The default log group name is the sysplex name. If you specify a different log group name when you start RRS, it will coordinate syncpoint processing with all systems in the sysplex that use the same RRS log group name.

Resource recovery services log streams

RRS uses five log streams that are shared by the systems in the log group. Every MVS image that runs RRS needs access to the coupling facility and the DASD on which are defined the system logger log streams for its log group.

Note: You can define RRS log streams as coupling facility log streams or as DASD-only log streams.

If using coupling facility log streams, the RRS images on different systems in a sysplex run independently but share log streams to keep track of the work. If a system fails, an instance of RRS on a different system in the sysplex can use the shared logs to take over the failed system’s work.

Use DASD-only log streams only in either single system sysplexes with one RRS image or a sysplex in which information should not be shared among RRS images.

The following list summarizes the RRS logs. In the list, *lgname* is the log group name. The default log group name is the sysplex name.

ATR.*lgname***.ARCHIVE**
: Information about completed units of recovery (URs). This log stream is recommended but optional.

ATR.*lgname***.RM.DATA**
: Information about the resource managers using RRS services.

ATR.*lgname***.MAIN.UR**
: The state of active URs. RRS periodically moves this information into the RRS delayed UR state log when UR completion is delayed.

ATR.*lgname***.DELAYED.UR**
: The state of active URs when UR completion is delayed.

ATR.*lgname***.RESTART**
: Information about incomplete URs needed during restart. This information enables a functioning RRS instance to take over incomplete work left over from an RRS instance that failed.

In a multiple-system sysplex, RRS log streams should normally reside in a coupling facility.

All RRS transaction logging for WebSphere Application Server for z/OS will occur solely in the DELAYED.UR log stream. You can still configure your MAIN.UR log stream so that it can handle a production workload in case you deploy a new container or the WebSphere Application Server for z/OS infrastructure changes. WebSphere Application Server for z/OS has no significant impact on the RM.DATA or RESTART logs.

Use the following steps to configure RRS.

1. Copy the RRS cataloged procedure, ATRRRS, from SYS1.SAMPLIB to SYS1.PROCLIB (or another proclib in the MSTJCLxx concatenation), and rename it RRS.

   If you want, you can set the log group name (GNAME) in the RRS cataloged procedure to a specific value. If you will share the ATRRRS proc among several systems, however, you might prefer to set the log group name at RRS startup or use a system variable in IEASYMxx to set each system's RRS log group name.
2. Establish the dispatching priority of the RRS address space.

   The best way to control RRS's dispatching priority is through the workload manager (WLM). IBM recommends that you put RRS in the SYSSTC service class. The service class you choose must give RRS a dispatching priority greater than or equal to the dispatching priority of applications and resource managers that use RRS. SYSSTC will usually accomplish this. For information about system-provided service classes, see *z/OS MVS Planning: Workload Management* (SA22-7602).
3. Define RRS as a subsystem.

   Place the following statement in an active IEFSSNxx parmlib member:
   ```
   SUBSYS SUBNAME(RRS)
   ```

   Place this statement after the statement that defines the primary subsystem. The subsystem name (RRS) must match the name of the RRS cataloged procedure. For more information about IEFSSNxx, see *z/OS MVS Initialization and Tuning Reference* (SA22-7592).

   Note: RRS does not support dynamic subsystem definition, so you cannot use the SETSSI ADD,SUBNAME=RRS command to define RRS as a subsystem. Even though this command will appear to succeed, subsequent attempts to start RRS will fail.
4. Set up the RRS log streams.
   - For your convenience, the WebSphere Application Server for z/OS product library SBBOJCL contains the following sample jobs to allocate the RRS log streams:

BBORRSLC
Allocates RRS log streams in a coupling facility

BBORRSLD
Allocates RRS log streams as DASD-only log streams

5. Start RRS.
   - To start RRS with a specific log group name "*lgname*", enter the following MVS console command:

     ```
     START RRS,GNAME=lgname
     ```

   - To stop RRS, enter the following MVS console command:

     ```
     SETRRS CANCEL
     ```

   Note: Do not stop RRS while WebSphere Application Server for z/OS servers are running.

For more information on setting up and running RRS, see *z/OS MVS Programming: Resource Recovery* (SA22-7616).

Related reference
Resource Recovery Service (RRS) tuning tips for z/OS
Use these tips to tune your z/OS operating system to optimize WebSphere Application Server performance.

Preparing the security server (RACF)

WebSphere Application Server for z/OS uses a SAF-compliant security product for its operating system security interfaces. The WebSphere Application Server for z/OS documentation assumes the use of z/OS Security Server (RACF). If you use another security product, consult the vendor for more information.

All z/OS systems in a sysplex must have access to consistent security information--shared RACF database or equivalent. If a shared security database is not used, you are responsible for ensuring that all WebSphere Application Server for z/OS security definitions are in effect on all systems in the sysplex.

1. Determine which RACF databases provide security information on your z/OS systems. If any WebSphere Application Server for z/OS cell will run on z/OS systems that have no shared RACF database, make plans to guarantee security database consistency for WebSphere Application Server for z/OS user IDs and privileges.
2. WebSphere Application Server for z/OS requires list-of-groups (GRPLIST) checking. See *z/OS Security Server RACF Security Administrators Guide* for information about turning on GRPLIST support.

Preparing TCP/IP

WebSphere Application Server for z/OS follows the CORBA standard, Internet Inter-ORB Protocol (IIOP), for communications. Accordingly, you must consider changes to your TCP/IP network and modify the TCP/IP configuration.

This section provides background information about changes you will need to make to your Domain Name Server (DNS) and TCP/IP. The actual steps to perform are in the customized instructions of the Customization Dialog and the Profile Management tool.

Consider the following tips for your TCP/IP network on z/OS.

- You can get started with a simple Domain Name Server (DNS) name server and a single z/OS image, but you should design your initial configuration with growth in mind.

  You might, for instance, intend to expand your business applications beyond the monoplex to a full sysplex configuration for performance reasons or to prevent a single point of failure. Several considerations come to bear here.

  Several DNS implementations and network router implementations allow the use of a generic location service daemon IP name while dynamically routing network traffic to like-configured servers. If you intend to expand your system beyond a monoplex, it might be worthwhile to use one of these implementations from the start. Non-round-robin DNS name servers limit your ability to expand without retrofitting a name server that allows dynamic network traffic routing.

Recommendation: If you are running in a sysplex, set up your TCP/IP network with Sysplex Distributor. This makes use of dynamic virtual IP addresses (DVIPAs), which increase availability and aid in workload balancing.

Beyond Sysplex Distributor, you have your choice of the following DNS and router implementations on or off z/OS:

– Non-round-robin DNS name servers.
– Round-robin DNS name servers.
– Network routers, such as the IBM Load Balancer. (In previous releases, IBM Load Balancer was known as Network Dispatcher.)

- Select the location service daemon IP name.

  For your stand-alone application server, choose the host name of the server under which you are running. For your deployment manager, choose a generic IP name that can resolve to any or all of the systems where location service daemons run.

  You must define the location service daemon host IP name during installation and customization. Use the location service daemon IP name you chose.

  Note: The administrative console has a location service daemon configuration page on which you set location service daemon variables.

- Select the port for the location service daemon server.

  If you change the location service daemon port number, you can access existing objects after you recycle all your servers. You cannot, however, access the following:

  – Any object handles your application stored to disk
  – Any object references your application stored in the persistent contexts of the name space.

- Set location service daemon port numbers and IP addresses.

  These are initially set in the Customization Dialog, but you can subsequently change them in the administrative console. Access the location service daemon configuration page through the administrative console navigation bar (on the left side of screen) under System Administration. If you need to use the IIOP through a firewall, ensure that your firewall supports IIOP.

  When recovering a server somewhere other than its configured system, ensure that the same port is not already in use on the system on which it is recovering. If it is, configure the server with a unique port to avoid a conflict.

  If comparing WebSphere Application Server for z/OS and WebSphere Application Server for other platforms, realize that only WebSphere Application Server for z/OS has an ORB SSL port.

  HTTP and HTTPS ports are found in individual servers under the Web container transports, which are in the administrative console as additional properties on the Web container configuration page (which is off the server configuration page).

  Watch for HTTP transport port conflicts if you previously installed WebSphere Application Server for z/OS.

  Ensure that you set up the following port assignments (along with those in the z/OS port assignments chart) on servers that require them in the administrative console:

  – ORB port
  – ORB SSL port
  – Web container transport port
  – Web container transport SSL port

  See the administrative console and the information center for more information on the WebSphere variables and how to set their values.

  You define ports differently depending on whether they are for the first server or subsequent servers. The first server you create is defined, along with its ports, through the Customization Dialog or the Profile Management tool. You have the ability to explicitly specify the ports as you define the server. Subsequent servers and their ports are defined through the administrative console. This means that you define the server first and the ports are automatically assigned. Then, once defined, you can inspect and change the port definitions through the administrative console.

- Some ports, such as the ORB SSL port and the server startup status port, are obtained dynamically.
- Other TCP/IP-related activities include setting up NFS, Web server, and Kerberos (which are all optional).
- If you use the DNS on z/OS, you might want to change the refresh timer interval (-t value) associated with the named location service daemon.

  The -t value specifies the time (nn, in seconds) between refreshes of cell names and addresses and of the weights associated with those names and addresses. The default is sixty seconds. Reducing the -t value will shorten the lapse time required to register the location service daemon IP name with the DNS, but will also increase DNS processing overhead. In our testing, we used an interval of 10 seconds.

  If you use the z/OS DNS, you have to set a location service daemon variable. Do this by setting WebSphere Variable at cell level:

  ```
  daemon_wlmable=1
  ```

  Note: You can perform this for only one cell in a sysplex at a time.

  For details, see *z/OS Communications Server: IP Configuration Reference*.

Checklist: Preparing the base operating system

Print out this worksheet and use it when collecting information about the z/OS system on which you plan to implement WebSphere Application Server for z/OS. Check off each item as you complete the task.

Date: ______________________

System name: _________________________ Sysplex name: _________________________

Preparing z/OS

- _____ Target system hardware and software requirements, including required maintenance in Preventive Service Planning (PSP) bucket, are met.
- _____ UNIX System Services is active with minimum required BPXPRMxx values or better.
- _____ The /tmp directory has at least 20 MB of free space.
- _____ The full data set names of required system data sets are specified:

Item	Data set name	In LPA or link list?
SCEERUN		
SCEERUN2		
System SSL		

- _____ System exits (IEFUSI) do not restrict WebSphere Application Server for z/OS address spaces to less than 512 MB.
- _____ The TSO segment default region size for WebSphere Application Server for z/OS installer and administrator TSO user IDs is at least 128 MB.

Preparing the sysplex

- _____ The target system is configured as a monoplex or into a multisystem sysplex. (Record sysplex name above.)
- _____ System Logger is configured:

Should the application servers reside in a coupling facility **or** DASD-only log streams?	

If a coupling facility, specify the structure name	
SMS data class (for log data sets)	
SMS storage class (for log data sets)	

Preparing JES or JES3

- _____ The system proclib for application server cataloged procedures is specified:

System proclib for application server cataloged procedures	

- _____ JES2 exit EXIT06 or JES3 exit IATUX03 do not prevent use of REGION= value on JOB or EXEC statements for WebSphere Application Server for z/OS address spaces.
- _____ Spool space is added if necessary.

Preparing Resource Recovery Services

- _____ The RRS cataloged procedure is present in system proclib:

Procedure name:	

- _____ The RRS dispatching priority is set using SYSSTC or other means.
- _____ The RRS cataloged procedure name is defined as a subsystem name in IEFSSN00. The subsystem name must match the cataloged procedure name.
- _____ The RRS log streams are set up.
- _____ RRS starts successfully.

Preparing Security Server (RACF)

- _____ If multiple security databases are in use, a plan is in place to provide database consistency.
- _____ List-of-groups (GRPLIST) support is active.

Preparing Component Trace (CTRACE)

- _____ Component trace is set up and working.

Planning for product configuration

This tasks helps you plan WebSphere Application Server for z/OS application serving environments for your z/OS target systems.

- Complete the steps in “Installing the product and additional software” on page 40 and “Preparing the base operating system” on page 41.
- Read the section “WebSphere Application Server for z/OS terminology” on page 52.
- If you already have any release of WebSphere Application Server for z/OS running on your z/OS target system, review Coexisting.

WebSphere Application Server for z/OS uses ″application serving environments″ to provide its functions. Configuring these application serving environments after product installation requires a fair amount of planning and coordination. If you have not previously configured WebSphere Application Server for z/OS, you should configure a ″practice″ stand-alone application server using the default options then proceed to configure the actual product configuration that you want.

Perform the tasks in this section to choose an application serving environment configuration and plan the necessary details for configuration.

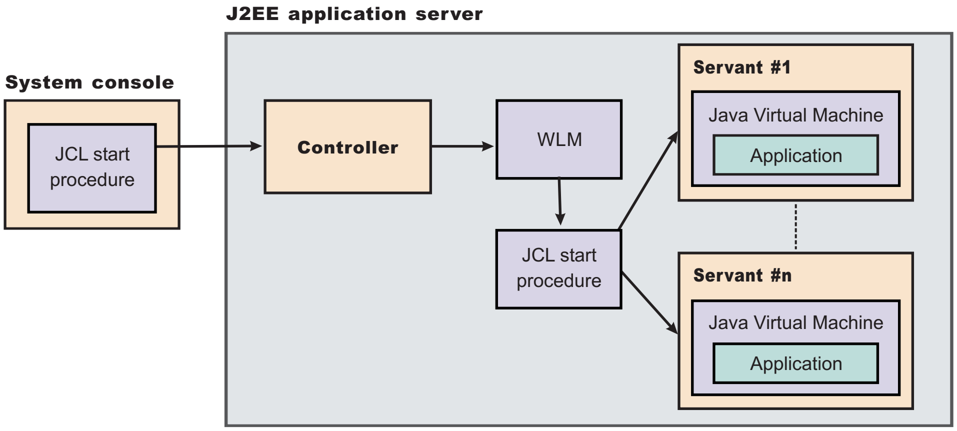
1. Decide whether to set up a stand-alone application server or a Network Deployment cell. See "Choosing between a stand-alone or Network Deployment configuration" on page 68 for more information.
2. Familiarize yourself with "z/OS considerations for WebSphere Application Server for z/OS" on page 55.
3. (Optional) If you have never set up a WebSphere Application Server for z/OS application serving environment before, follow the steps in "Building a practice WebSphere Application Server for z/OS cell" on page 84 to gain experience in configuring and working with an application serving environment.
4. Follow the directions for the type of application serving environment you want to configure:
   - "Planning for a stand-alone application server cell" on page 86
   - "Planning for a Network Deployment cell" on page 100.
   - "Planning for a new managed node in a Network Deployment cell" on page 112
   - "Planning to federate a stand-alone server into a Network Deployment cell" on page 121

   Use the worksheet included with each option to record your planning decisions and additional configuration information.

When you have completed the planning worksheet for the configuration you have selected, you are ready to configure the application serving environment. See "Configuring the product after installation" on page 131 for information.

WebSphere Application Server for z/OS terminology

In WebSphere Application Server for z/OS, the functional component on which applications run is called a *server*. Servers comprise address spaces that actually run code.



Within each server are two kinds of address spaces: controllers and servants. A *controller* runs system authorized programs and manages tasks, such as communication, for the server. Each server has one controller that you start with a JCL start procedure when you enter the appropriate start command on the MVS console.

A *servant* is the address space in which the JVM resides. It runs unauthorized programs such as business applications. Depending on the workload, a server has one or more servants running at a time. When work builds up, WLM dynamically starts additional servants to meet the demand.

Note: The location service daemon and node agent are specialized servers and have no servants. The control region adjunct (not shown in the diagrams) is a specialized servant that interfaces with the new service integration busses to provide messaging services.

Here is a quick breakdown of the different server types on your system:

Unmanaged (stand-alone) application server
The application server that was set up during stand-alone configuration that hosts your J2EE applications.

Managed (Network Deployment) application server
The application server set up during Network Deployment configuration that hosts your J2EE applications.

Location service daemon
A server that is the initial point of contact for client requests in either configuration.

JMS server
Hosts the JMS function in the WebSphere Application Server for z/OS, which controls the MQ broker and queue manager in either configuration. The JMS server no longer exists as in previous versions of WebSphere Application Server for z/OS. Its function has been replaced with new service integration busses.

Deployment manager
A specialized application server that hosts the administrative console application (it hosts only administrative applications) and provides cell-level administrative function in a Network Deployment configuration. The administrative console application administers servers (grouped into nodes) on many different systems. The deployment manager is the sole occupant of its own node structure that does not need a node agent because there are no application servers in the node, and a cell can have only one deployment manager.

Note: The version of the administrative console application that runs in the deployment manager is designed to manage multinode environments, whereas the version in the stand-alone application server is for single node environments only.

Node agent
Provides node-level administrative function in a Network Deployment configuration.

Note: Every element of the configuration (servers, clusters, nodes and cells) has both a long and short name:

- The ″Server name″ is the server long name used in the HFS path and the principal name by which the server is known to WebSphere Application Server for z/OS. It is used to identify the server through the administrative console and scripting. It is a mixed case name and greater than 8 characters in length.
- The ″Server short name″ is the platform-specific native alias and the principal name by which the server is known to z/OS. It is used to identify the server to underlying z/OS facilities, such as the Security Server, JES, WLM and ARM. For example, the server short name is used as the MVS JOBNAME.
- The ″Cluster short name″ is used as the WLM application environment name.

A *cluster* is a *logical grouping* of like-configured servers. Clusters exist to promote scalability and availability; workload balancing occurs across the servers in a cluster. Clusters allow you to partition workloads into separate servers while still referring to them as a single unit. Clustering is typically applied to a multinode cell, where each node is configured on a separate system and the cluster has a member (server) on each node. Client requests are distributed among the cluster members based on workload manager decisions.

Note: If you intend for your cluster to span multiple systems in a sysplex, you might need to set up a shared HFS. See Considerations for clustered servers and stateful session beans for a list of conditions that would make a shared HFS necessary.

A node contains servers that can be part of a cluster. The cluster can span nodes if all involved nodes are in the same cell.

Here is a quick breakdown of cells, nodes, and clusters:

cell A logical collection of WebSphere Application Server for z/OS nodes that are administered together. The cell is the largest unit of organization.

- Nodes that comprise a cell can reside on systems in the same sysplex, differing sysplexes, on the same z/OS monoplex, or on differing systems entirely. A cell that consists of nodes on differing systems or sysplexes is called a *heteregeneous cell*.
- A z/OS sysplex or monoplex can contain multiple WebSphere Application Server for z/OS cells.
- Different cells can have nodes on the same systems, although a given node can be a member of only one cell.
- There are two kinds of WebSphere Application Server for z/OS cells: stand-alone application server cells, and Network Deployment cells.

node A logical collection of servers on a particular z/OS system in the cell.

- The cell to which a node belongs can span several systems, but the node must remain within a single z/OS system.
- A z/OS system can contain multiple WebSphere Application Server for z/OS nodes, belonging to the same or different cells.
- A stand-alone WebSphere Application Server for z/OS cell consists of a single node. Due to administrative constraints, this node should have only a single application server in it.
- A Network Deployment cell consists of a deployment manager node, which is responsible for cell-wide administrative tasks, and any number of managed nodes. Each managed node contains a node agent, which handles communication with the cell's deployment manager, and any number of application servers.

cluster

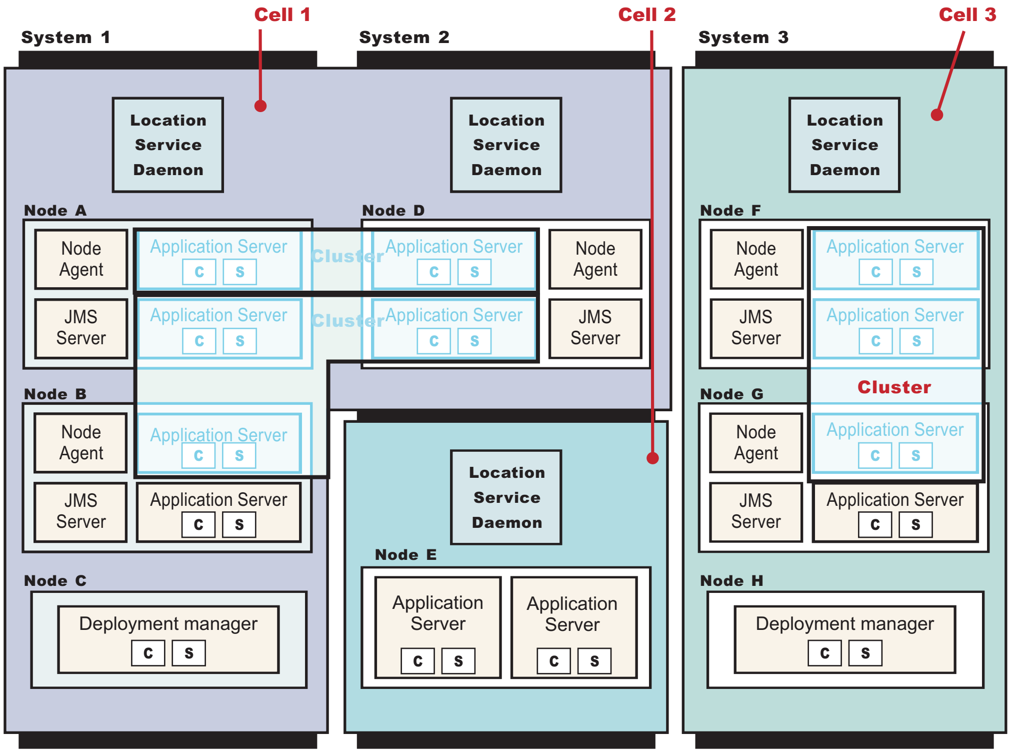
A logical collection of like-configured application servers that provides performance, reliability and administration advantages.

- A cluster can span nodes and systems within the same cell.
- Clusters are not a layer in the cell/node/server hierarchy. Instead, they are a way of grouping servers that host the same applications within a cell.

A cluster can span nodes and systems within the same cell.

To help you understand the interaction between servers, clusters, nodes and cells, here is a diagram depicting various configurations you can set up in your Network Deployment sysplex:

WebSphere for z/OS: Possible configurations in a sysplex



Cells 1 and 3 in the illustration depict Network Deployment configuration cells. Cell 2 is a stand-alone configuration cell.

Note: This precise node assignment does not need to apply. The deployment manager node can exist on one system, other managed nodes (that have been federated into the deployment manager) can exist on differing systems. Such a configured cell of differing machines or operating systems is called a *heterogeneous cell* and expands the possible topologies you can consider for your network deployment.

Using a heterogeneous cell to support mixed platforms within a cell

Cells can span z/OS sysplex environments and spanning other operating systems. For example, z/OS nodes, Linux nodes, UNIX nodes, and Windows nodes can exist in the same Application Server cell. This kind of configuration is referred to as a *heterogeneous* cell.

A heterogeneous cell does require significant planning. The Heterogeneous Cells – cells with nodes on mixed operating system platforms white paper outlines the planning and system considerations required to build a heterogeneous cell.

Related concepts

Introduction: Environment
Your WebSphere Application Server product environment includes Web server plug-ins, WebSphere Application Server variables, and other data objects. Configure values for settings in these categories using the Environment section of the administrative console.

z/OS considerations for WebSphere Application Server for z/OS

Use this section to familiarize yourself with the z/OS facilities used by the WebSphere Application Server for z/OS application serving environment.

Cataloged procedures

This concept is an explanation of how WebSphere Application Server for z/OS server uses the JCL cataloged procedures.

Each WebSphere Application Server for z/OS server uses a JCL cataloged procedure. These procedures are all fairly similar and consist of a main cataloged procedure and an INCLUDE member that contains DD statements. Here are sample cataloged procedure library members for a controller as generated by the Customization Dialog:

Procedure library member BBO6ASR:

```
//BBO6ASR  PROC ENV=,Z=BBO6ASRZ
//  SET ROOT='/WebSphere/V6R1'
//BBOSR   EXEC PGM=BBOSR,REGION=0M,TIME=NOLIMIT,
// PARM='TRAP(ON,NOSPIE),ENVAR("_EDC_UMASK_DFLT=007") /'
//BBOENV DD PATH='&ROOT/&ENV/was.env'
//  INCLUDE MEMBER=&Z
```

Procedure library member BBO6ASRZ:

```
//*
//* Output DDs
//*
//CEEDUMP   DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
//SYSOUT    DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
//SYSPRINT  DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
//*
//*Steplib Setup
//*
//STEPLIB  DD DISP=SHR,DSN=WAS.V61.SBBOLD2
//         DD DISP=SHR,DSN=WAS.V61.SBBOLOAD
```

The `PGM=` parameter on the `EXEC` statement in the cataloged procedure specifies the type of WebSphere Application Server for z/OS server. The program names are as follows:

- BBOCTL (controller)
- BBOSR (servant)
- BBOCRA (adjunct process)
- BBODAEMN (location service daemon)

The `PARM=` parameter on the `EXEC PGM` statement contains the parameters that are passed to the program identified by the `PGM=` parameter.

Customization Note: The appropriate interface for making changes to the language environment (LE) parameters is through the `PARM=` parameter; however, do not modify any LE parameters without first consulting with the IBM support team. The LE parameters are set internally to ensure the best possible performance of the WebSphere Application Server, which is the main LE application running in the address space. If you need to add or change LE parameters, make sure that you work with the IBM WebSphere support team to ensure that the internally set parameters are not compromised.

The `BBOENV DD` statement points to the was.env (startup parameter) file for the server. The path to this file consists of the configuration HFS directory name (hardcoded using the ROOT JCL variable) and the symbolic link for this particular server, which is specified at startup using the `ENV=` parameter.

The `INCLUDE` statement in the main cataloged procedure causes the incorporation of all JCL statements from the INCLUDE member (in this case, BBO6ASRZ). By convention, the INCLUDE member's name consists of the name of the main cataloged procedure with the letter ″Z″ appended to the end. The INCLUDE member provides common `DD` statements for the started task.

The `STEPLIB DD` statement for the location service daemon determines the WebSphere Application Server for z/OS code base intended for use. If the location service daemon finds the SBBOLPA modules in its STEPLIB concatenation, it loads them into common storage. If the modules are not found in STEPLIB, then the location service daemon checks for their presence in the system link pack area.

Note: If the modules are found in neither location, the location service daemon reports an error and terminates.

Once the modules are located (and loaded if necessary), the location service daemon holds a vector table of load module addresses that the rest of the cell's servers will use.

All other servers for the location service daemon's cell on the same z/OS system must use the same product code base as the location service daemon:

- Each server `STEPLIB DD` statement (in the INCLUDE JCL member) should have the same WebSphere Application Server for z/OS data sets as listed in the location service daemon's `STEPLIB DD` statement.

  Note: The server STEPLIB DD concatenation can also include additional DD statements for CICS or DB2 libraries and so on.
- Each server's home directory must point to a product HFS that matches the service level of the location service daemon's WebSphere Application Server for z/OS load modules.

The controller cataloged procedure includes some additional statements before the `EXEC` statement that invokes BBOCTL. These are used to invoke the post installer program that applies any needed maintenance to the configuration HFS and its home directories when service is applied to the product HFS and load modules.

The following sections describe the cataloged procedures required for each configuration, provide a recommended naming convention, and explain how the SAF user ID for each server is determined.

Cataloged procedures for stand-alone application servers

A stand-alone application server uses the following cataloged procedures with corresponding INCLUDE JCL members for each:

- Controller cataloged procedure
- Servant cataloged procedure
- Adjunct process cataloged procedure
- Location service daemon cataloged procedure

You can use the same cataloged procedures for different stand-alone servers IF the configuration HFS and product code level (including STEPLIB) are the same for both servers. The Customization Dialog will overwrite identically named proclib members if identical names are chosen or if customization jobs and files are regenerated. If you have a local STEPLIB or make other changes to the procedures, therefore, make sure that they are reapplied after you use the Customization Dialog.

Cataloged procedures for Network Deployment cells

A Network Deployment cell uses the following cataloged procedures, with corresponding INCLUDE JCL members for each:

For the deployment manager:

- Deployment manager controller cataloged procedure
- Deployment manager servant cataloged procedure

For each application server node:

- Application server controller cataloged procedure (also used for the node agent)

- Application server servant cataloged procedure

For the location service daemon (one per z/OS system):

- Location service daemon cataloged procedure

The application server servant cataloged procedure is the only one likely to require modification, in order to place libraries (CICS, DB2, and so on) in the STEPLIB concatenation.

You can use the same cataloged procedures for several nodes in a Network Deployment cell, or even for several cells, IF the configuration HFS and product code level (including STEPLIB) are the same for both servers. The Customization Dialog will overwrite identically-named proclib members if identical names are chosen or if customization jobs and files are regenerated. If you have a local STEPLIB or make other changes to the procedures, therefore, make sure that they are reapplied after you use the Customization Dialog.

A recommended cataloged procedure naming convention

Use a consistent naming convention for your WebSphere Application Server for z/OS cataloged procedures. The procedure name should distinguish between WebSphere Application Server for z/OS version, configuration HFS, and which data sets are listed in STEPLIB. Limit cataloged procedure names to seven characters to allow the Customization Dialog to create INCLUDE JCL member names by appending the letter ″Z.″

For example, the following convention works for either a stand-alone application server or Network Deployment cell, where *cc* is a two-character cell identifier:

Deployment manager controller	*cc*6DCR
Deployment manager servant	*cc*6DSR
Location service daemon controller	*cc*6DMN
Application server controller	*cc*6ACR
Application server servant	*cc*6ASR
Control region adjunct	*cc*6CRA

If you require separate cataloged procedures for nodes on different systems in a sysplex (if they need independently settable STEPLIB statements to allow for a nondisruptive restart for example), either place the location service daemon and application server procedures in system-specific proclibs, or append a one-character system identifier to the cataloged procedure names for the location service daemon and application servers.

Assigning user IDs to WebSphere Application Server for z/OS address spaces

If you use z/OS Security Server (RACF) as your SAF-compliant security system on z/OS, then STARTED class profiles are used to assign started task user IDs to each WebSphere Application Server for z/OS server. These STARTED profiles are set up by the Customization Dialog batch jobs. Update these STARTED profiles as needed to place servers you create yourself under the appropriate user IDs.

Controllers (deployment manager, location service daemon, node agent or applications server controller) are started using a console `START` command that you issue either from the MVS console or internally. For these servers, the STARTED profile name that is checked is of the form `procname.jobname`.

Whenever it creates a controller or daemon cataloged procedure, the Customization Dialog also creates a STARTED profile that associates all controllers using that cataloged procedure with the appropriate controller user ID and configuration group. Therefore, if you set up a stand-alone application server with default names, the Customization Dialog would create the following STARTED profiles for controllers:

- `RDEFINE STARTED BBO6ACR.* STDATA(USER(ASCR1) GROUP(WSCFG1) TRACE(YES))`
- `RDEFINE STARTED BBO6DMN.* STDATA(USER(WSDMNCR1) GROUP(WSCFG1) TRACE(YES))`

Note: `TRACE(YES)` writes message IRR812I to the MVS console whenever the profile is used.

Servant regions (application server servants and adjunct processes) are started using Workload Manager (WLM). For these servers, the STARTED profile name that is checked is of the form `jobname.jobname`.

Unfortunately, there is no way to assign all servers using a particular servant cataloged procedure to a servant user ID. Therefore, the Customization Dialog creates both a generic STARTED profile for servants, based on the jobname prefix BBO, and a specific STARTED profile for any servant whose jobname does not begin with BBO, that assign their servants to the servant user ID, and a specific STARTED profile for the adjunct process that assigns it to the CRA user ID. If default names are chosen, the following servant STARTED profiles are created for a stand-alone application server:

- `RDEFINE STARTED BBO*.* STDATA(USER(ASSR1) GROUP(WSCFG1) TRACE(YES))`

  Note: There is no specific application servant profile because the server name BBOS001S is ″caught″ by this profile.
- `RDEFINE STARTED BBOS001A.* STDATA(USER(ASCRA1) GROUP(WSCFG1) TRACE(YES))`

When you choose cataloged procedure names, make sure that the appropriate STARTED profile is in place to map the server to its appropriate SAF user ID. Use the RACF ISPF panels or the `RLIST STARTED` command to display the STARTED profiles.

If you use another SAF-compliant security system, contact the security server vendor for WebSphere Application Server for z/OS setup information.

Cataloged procedure for the administrative asynchronous task

The asynchronous administrative task also requires a cataloged procedure. This very simply cataloged procedure does not include a STEPLIB or configuration HFS pointer. However, since it must run under a specific user ID and group associated with the security domain of the cell for which it runs, you must choose a different cataloged procedure name for each security domain or cell.

Recommendation:

Name the administrative asynchronous task cataloged procedure ″*cc*6SH,″ where *cc* is a two-character cell identifier.

Component trace

This concept describes how WebSphere Application Server for z/OS uses the z/OS component trace facility.

WebSphere Application Server for z/OS uses z/OS component trace (CTRACE) facilities to manage the collection and storage of trace data. CTRACE data is written to address space buffers in private (pageable) storage, which can be formatted using IPCS if a dump of the address space is taken. CTRACE data can also be written to trace data sets on disk or tape using an external writer. Although CTRACE data is primarily output for use by IBM service personnel, using CTRACE capabilities at your installation allows you to have additional trace data available when a problem first occurs. Because CTRACE efficiently uses system resources, you can collect valuable trace data with minimal impact on performance.

Configuration file system

This article describes the planning decisions you need to make when setting up a WebSphere Application Server for z/OS configuration file system.

Cell, node, and server settings as well as deployed applications are stored in the WebSphere Application Server for z/OS configuration file system.

Note: You can use a zSeries file system (zFS) or hierarchical file system (HFS) for the configuration file system.

The following sections present decisions that you must make when setting up your WebSphere Application Server for z/OS configuration file system and give you information on how to make those decisions based on the needs of your planned configuration.

Each node needs a home directory

Every WebSphere Application Server for z/OS node--whether a stand-alone application server, deployment manager, managed application server node, or location service daemon--requires a read/write home directory, sometimes referred to as its WAS_HOME.

This is the structure of a WebSphere Application Server for z/OS configuration file system, mounted at /WebSphere/V6R1. It contains a WebSphere Application Server home directory for a single application server named BBOS001, with a cell and a node both named SYSA.

```
    /WebSphere/V6R1
        /AppServer
            /bin
            /classes
            /java
            /lib
            /logs
            /profiles
            /default    -> this is the profile_root directory
            /temp
            ...
       /Daemon
            /config
                /SYSA
       SYSA.SYSA.BBODMNB       ->  /WebSphere/V6R1/Daemon/config/SYSA/SYSA/BBODMNB
       SYSA.SYSA.BBOS001       ->
/WebSphere/V6R1/AppServer/profiles/default/config/cells/SYSA/nodes/SYSA/servers/server1
       SYSA.SYSA.BBOS001.HOME  ->  /WebSphere/V6R1/AppServer
```

The WebSphere Application Server home directory for BBOS001 is named AppServer. It contains directories with complete configuration information for the SYSA node and the BBOS001 server.

The /Daemon directory contains configuration information for location service daemons defined to nodes in this configuration file system.

Note: The /Daemon/config subdirectory is subdivided by cell name. If the cells have different short names, the location service daemon information for each is kept separate.

The daemon home directory has the fixed WebSphere Application Server home name *Daemon*.

Symbolic links are used to access startup parameters

In addition to the WebSphere Application Server home directories themselves, the configuration file system contains a multipart symbolic link for each server that points to the startup parameters for the server. The symbolic link is named *cell_short_name.node_short_name.server_short_name*.

The sample configuration file system above contains a symbolic link SYSA.SYSA.BBODMNB to start the location service daemon and a symbolic link SYSA.SYSA.BBOS001 to start the BBOS001 application server. The second symbolic link is specified in the ENV parameter on the `START` command when the server or location service daemon is started from the MVS console:

```
START procname,JOBNAME=BBOS001,ENV=SYSA.SYSA.BBOS001
```

Each symbolic link points to the subdirectory where the server's was.env file resides. This file contains the information required to start the server.

Note: During post-install processing, described below, the server JCL needs to specify the WebSphere Application Server home directory itself, rather than the location of the was.env file. This is the purpose of the SYSA.SYSA.BBOS001.HOME symbolic link shown above.

Sharing the configuration file system between cells

Two or more WebSphere Application Server for z/OS cells (stand-alone application server, Network Deployment, or both) can share a WebSphere Application Server for z/OS configuration file system, provided the following conditions are met:

- All cells using the configuration file system must be set up using the same common groups and users. In particular, each must have the same administrator user ID and configuration group.
- The cells must have distinct cell short names.
- Each node must have its own WAS_HOME directory that is not shared with any other node or cell.

As noted above, you can share the daemon home directory (/Daemon) between cells, as it has subdirectories farther down for each cell in the configuration file system.

Note: Be aware that sharing a configuration file system between cells increases the likelihood that problems with one cell might cause problems with other cells in the same configurations file system.

Sharing the configuration file system between systems

Two or more z/OS systems can share a configuration file system, provided the z/OS systems have a shared file system and the configuration file system is mounted R/W. All updates are made by the z/OS system that ″owns″ the mount point. For a Network Deployment cell, this is generally the z/OS system on which the cell deployment manager is configured.

Choosing a WebSphere Application Server for z/OS configuration file system mount point

The choice of WebSphere Application Server for z/OS configuration file system mount points depends on your z/OS system layout, the nature of the application serving environments involved, and the relative importance of several factors: ease of setup, ease of maintenance, performance, recoverability, and the need for continuous availability.

In a single z/OS system:

If you run WebSphere Application Server for z/OS on a single z/OS system, you have a wide range of choices for a z/OS configuration file system mount point. You might want to put several stand-alone application servers in a single configuration file system with a separate configuration file system for a production server or for a Network Deployment cell. Using separate configuration file system data sets improves performance and reliability, while using a shared configuration file system reduces the number of application server cataloged procedures you need.

You might have one configuration file system with your development, test and quality assurance servers, all in the same common groups and uses as in the following example:

```
/WebSphere/V6_test
              /DevServer    - home to stand-alone server DVCELL, with server DVSR01A
              /TestServer1  - home to stand-alone server cell T1CELL, with server T1SR01A
              /TestServer2  - home to stand-alone server cell T2CELL, with server T2SR01A
              /QAServer     - home to Network Deployment cell QACELL, with deployment manager QADMGR and server QVSR
```

and a separate configuration HFS for your production cell:

```
/WebSphere/V6_prod
              /CorpServer1  - home to Network Deployment cell CSCELL, with deployment manager CSDMGR and server CSSR
```

In a multisystem z/OS sysplex with no shared HFS:

In a multisystem sysplex with no shared HFS, each z/OS system must have its own configuration file system data sets. For stand-alone application servers and for Network Deployment cells that do not span systems, the options are the same as for a single z/OS system.

For Network Deployment cells that span systems:

Here you have two options:

- You can use a different mount point for the cell's configuration file system data sets on each system. This allows you to move nodes easily between systems (if a system becomes inoperative or is being upgraded for example), since each mount point is unused on the other systems in the sysplex, allowing you to mount the failed system's configuration file system data sets on an alternate system in the sysplex.

  On system LPAR1, for example, you might have a configuration file system for one part of a cell:

  ```
  /var/WebSphere/V6config1
                   /DeploymentManager  - home to deployment manager F1DMGR in cell F1CELL
                   /AppServer1         - home to node F1NODEA and servers F1SR01A and F1SR02A
  ```

  with a second configuration file system on LPAR2:

  ```
  /var/WebSphere/V6config2
                   /AppServer2         - home to node F1NODEB and servers F1SR02B (clustered) and F1SR03B
  ```

  This setup has the advantage that you can move the deployment manager and node F1NODEA to LPAR2 or move node F1NODEB to LPAR1. The disadvantage of this configuration is that F1NODEA and F1NODEB will require separate sets of cataloged procedures.
- Or you can use the same mount point for all configuration file system data sets in a particular cell. This allows you to use common cataloged procedures and make the systems look very similar.

  Using the same cell setup as above, node LPAR1 would have one configuration file system:

  ```
  /var/WebSphere/V6F1
                   /DeploymentManager  - home to deployment manager F1DMGR in cell F1CELL
                   /AppServer1         - home to node F1NODEA and servers F1SR01A and F1SR02A
  ```

  and LPAR2 would have a separate file system at the same mount point:

  ```
  /var/WebSphere/V6F1
                   /AppServer2         - home to node F1NODEB and servers F1SR02B (clustered) and F1SR03B
  ```

  However, relocation of either LPAR's node(s) to the other system would require merging a copy of one configuration file system into the other.

In a multisystem z/OS sysplex with a shared HFS:

If your sysplex has a shared hierarchical file system, you can simply mount a large configuration file system for the entire cell. When using the Customization Dialog, specify the common configuration file system mount point on each system. As noted above, you should update the configuration file system from the z/OS system hosting the deployment manager. Performance will depend on the frequency of configuration changes, and ensure you devote extra effort to tuning if this option is chosen.

Alternatively, you can mount a separate configuration file system on each system, perhaps using the system-specific file system mounted at /&SYSNAME on each system:

```
/LPAR1/WebSphere/V6F1
                   /DeploymentManager  - home to deployment manager F1DMGR in cell F1CELL
                   /AppServer1         - home to node F1NODEA and servers F1SR01A and F1SR02A

/LPAR2/WebSphere/V6F1
                   /AppServer2         - home to node F1NODEB and servers F1SR02B (clustered) and F1SR03B
```

Each system (LPAR1 and LPAR2) mounts its own configuration file system on its system-specific mount point. When using the Customization Dialog, specify the following:

- /LPAR1/WebSphere/V6F1 on LPAR1
- /LPAR2/WebSphere/V6F1 on LPAR2

Performance is better with this option than with a shared sysplex, and, depending on choice of mount point, it might be possible to mount a configuration file system temporarily on the other LPAR if the original owner is down. You can make cataloged procedures system-specific or use &SYSNAME to select the configuration file system mount point.

If you really want to use the same apparent mount point for all configuration file system data sets, you can use symbolic links to redirect a common mount point to a different file system on each system:

- ln -s $SYSNAME/WebSphere WebSphere
- Mount LPAR1's configuration file system at /LPAR1/WebSphere/V6F1.
- Mount LPAR2's configuration file system at /LPAR2/WebSphere/V6F1.

If this is done correctly, you can specify a configuration mount point of /WebSphere/V6F1 for each system in the Customization Dialog and still enjoy the benefits of system-specific customization file system data sets. However, when this setup is used, it is **not** possible to easily move configuration file system data sets from one system to another. All nodes expect to find their data in /WebSphere/V6F1, and you can mount only one configuration file system at this mount point on each system.

Recommendations:

- On a single z/OS system:
  - Create a configuration file system at /WebSphere/V6R0 and use it to create a ″practice″ stand-alone server. Place home directories for additional non-production stand-alone application servers in the same configuration file system.
  - Create a separate configuration file system at /WebSphere/V6R0_*cell_short_name* for each production stand-alone server or Network Deployment cell.
- On a multisystem sysplex with no shared file system, follow the recommendations above for a single z/OS system. This will allow you to use common cataloged procedures for each cell. Establish separate mount points on each system for any cell that you might need to recover on an alternate system in the sysplex.
- On a multisystem sysplex with a shared file system, use a shared configuration file system when performance is not an issue or when a shared file system is required to support specific WebSphere Application Server for z/OS functions. Use nonshared configuration file system data sets when performance is an issue, or when you must avoid a single point of failure.

Choosing WebSphere Application Server home directory names

The WebSphere Application Server home directory is always relative to the configuration file system in which it resides. In the Customization Dialog, therefore, you choose the configuration file system mount point on one panel and fill in just the single directory name for the home directory on another. But when instructions direct you to go to the WAS_HOME directory for a server, they are referring to the entire path name, configuration file system and home directory name combined (/WebSphere/V6R1/AppServer for example).

You can choose any name you want for a home directory if it is unique in the configuration file system. If you are creating a stand-alone application server or new managed server node to federate into a Network Deployment cell, be sure to choose one that is not in use in the Network Deployment cell's configuration file system.

If you have one node per system, you might want to use some form of the node name or system name. Alternatively, you can use ″DeploymentManager″ for the deployment manager and ″AppServer*n*″ for each application server node.

Relationship between the configuration file system and the product HFS

The configuration file system contains a large number of symbolic links to files in the product HFS (/usr/lpp/zWebSphere/V6R1 by default). This allows the server processes, administrator, and clients to access a consistent WebSphere Application Server for z/OS code base.

Note that these symbolic links are set up when the WebSphere Application Server home directory is created and are very difficult to change. Therefore, systems that require high availability should keep a separate copy of the WebSphere Application Server for z/OS product HFS and product data sets for each maintenance or service level in use (test, assurance, production, and so forth) to allow system maintenance, and use intermediate symbolic links to connect each configuration HFS with its product HFS.

Tip: If you configure your Network Deployment environment using the default value for the product HFS path in the Customization Dialog, it will result in all the nodes pointing directly at the mount point of the product HFS. This makes rolling maintenance in a nondisruptive manner almost impossible. If a cell is configured in this way, applying service to the product HFS affects all the nodes at the same time; and if multiple cells are configured in this way, applying service to the product HFS affects all the cells at the same time. You might want to specify what is referred to as an ″intermediate symbolic link″ between each node's configuration HFS and the actual mount point of the product HFS. This strategy is described in the WebSphere Application Server for z/OS V5 - Planning for Test, Production and Maintenance white paper. See the Washington Systems Center Sample WebSphere for z/OS ND Configuration white paper for more information about this issue and its relationship to applying maintenance. See the WebSphere for z/OS: Updating an Existing Configuration HFS to Use Intermediate Symbolic Links instructions for information on obtaining and using a utility that would allow you to update an existing configuration HFS to use intermediate symbolic links.

When a WebSphere Application Server for z/OS node is started, the service level of the configuration is compared against the service level of the product file system. If the configuration file system service level is higher than that of the product file system (probably meaning that an old product file system is mounted), the node's servers will terminate with an error message. If the configuration file system service level is lower than that of the product file system (meaning that service has been applied to the product code base since the node was last started), a task called the post-installer checks for any actions that need to be performed on the configuration file system to keep it up to date. For more information about the post-installer, see "Applying product maintenance" on page 244.

Digital certificates

This article describes digital certificates and keyrings for SSL/JSSE, what certificates are created during customization and why.

Digital certificates and keyrings or keystores are required for Secure Socket Layer communication. These certificates may be stored in the System Authorization Facility security database, or in files in the configuration file system.

See Secure Sockets Layer security for WebSphere Application Server for z/OS for information about what certificates are created.

Logstreams

This concept is an explanation of error log, client error log, transaction XA partner log, with pointers to sample creation/deletion jobs (new in WebSphere Application Server release 6 version 1).

WebSphere Application Server for z/OS uses the System Logger, an MVS component that allows applications to log data in a sysplex, to log error and trace information and provide XA transaction logging.

The System logger creates and manages log streams, which are written first to a coupling facility or local in-memory buffer, then transferred to log data sets on DASD for longer term access. Log streams that are written to local buffers rather than to a coupling facility are called DASD-only log streams.

Output destinations

This concept is an explanation of the various server DD statements, console output, trace output, and dump output.

Since WebSphere Application Server controllers and servants are z/OS started task address spaces, they can produce a variety of output:

- Server output and error messages
- Trace records
- System dumps.

This output can be written to a variety of destinations:

- JES2 print and punch files (referred to as ″sysout″ or ″job output″)
- Files written to the configuration file system or other file systems
- z/OS log streams
- Component trace data sets.

See Managing operator message routing for an explanation of server output and error messages.

For information about dump routing, see Choosing diagnostic information sources.

SAF user IDs

This concept is an explanation of the types of SAF user IDs required by WebSphere Application Server; these user IDs are created during customization.

Each WebSphere Application Server for z/OS address space must be associated with a System Authorization Facility user ID. (A SAF-compliant security package, such as RACF, is required by the WebSphere Application Server runtime.)

See the customization worksheets for required user IDs and default names for each type of configuration.

SAF groups

This concept is an explanation of the SAF groups required by WebSphere Application Server; these groups are created during customization.

WebSphere Application Server for z/OS uses System Authorization Facility groups to associate user IDs with common sets of permissions.

A common set of SAF groups is used across a WebSphere Application Server for z/OS cell. See “Customization Dialog variables: Common MVS Groups and Users” on page 80 for required groups and default group names.

SAF profiles

This concept is an explanation of the SAF profiles used WAS; these SAF profiles are created during customization. SAF profiles are used if z/OS managed security is selected.

System Authorization Facility profiles are created during customization to grant necessary authorities to WebSphere Application Server for z/OS address spaces.

See System Authorization Facility profile names generated by the Customization Dialog for more information.

Scheduler database

This concept describes the scheduler service in WebSphere Application Server and the timing intervals.

The scheduler service in WebSphere Application Server is responsible for starting actions at particular times or intervals. The performance of the scheduler database is critical to efficient scheduler operation.

WebSphere Application Server for z/OS provides the following sample jobs in the SBBOJCL product data set to create a local scheduler database using DB2:

BBOCRTTS	Create DB2 tablespaces for a scheduler database
BBOCRTSC	Create DB2 schemas for a scheduler database

The following sample jobs in SBBOJCL can be used to delete a scheduler database in DB2 when it is no longer needed:

BBODRPSC	Drop DB2 schemas for a scheduler database
BBODRTSC	Drop DB2 tablespaces for a scheduler database

Make copies of these jobs, customize them according to the instructions in the job, and run as needed to create or delete scheduler databases.

TCP/IP port convention

This article lists the default server values for WebSphere Application Server for z/OS.

z/OS port assignments

The following table lists the default port assignments for z/OS.

Port	Stand-alone location service daemon	Stand-alone application server	ND location service daemon	ND application server	Node agent	Deployment manager
HTTP		9080		9080		
HTTP/S		9443		9443		
Admin Console Port	9060			9060		9060
Admin Console Secure Port	9043			9043		9043
Bootstrap		2809		9810	2809	9809
ORB	5655	2809	5755	9810	2809	9809
ORB SSL	5656	0	5756	0	0	0
SOAP/JMX		8880		8880	9360	8879
Node Discovery					7272	
Node Multicast Discovery					5000	
Cell Discovery						7277

Port	Stand-alone location service daemon	Stand-alone application server	ND location service daemon	ND application server	Node agent	Deployment manager
Service Integration		7276				
Service Integration Secure		7286				
Service Integration MQ Interoperability		5558				
Service Integration MQ Interoperability Secure		5578				
Session Initiation Protocol Port		5060				
Session Initiation Protocol Secure Port		5061				
High Availability Manager Communications		9353			9354	9352

Location service daemon ports

Stand-alone application server node location service daemons are considered temporary. The ports assigned to a stand-alone application server node's location service daemon are used only until that node is federated. It is advisable to set aside a couple of ports to serve as ″interim ports″ for the stand-alone application server node location service daemon. The ″permanent″ location service daemon ports are the ones assigned to the deployment manager. Those same ports are copied to location service daemons created when a stand-alone application server node on another MVS image is federated into the deployment manager cell.

Node agent ports

There is a node agent per MVS image on which the cell spans. One design option calls for all node agents to have the exact same ports so the Sysplex Distributor is able to balance the traffic between the two. The node agent is created when the BBOWADDN customized job is run.

Server clusters

A server cluster is a grouping of two or more servers into a one logical server. A cluster is created through the administrative console. Servers within a cluster are called ″cluster members.″ Servers (″members″) within a cluster start out being clones of one another. When it comes to the TCP ports for the members in a cluster, the administrative console allows you during the creation of the cluster to specify if you want the HTTP ports to be unique or the same. The other ports -- bootstrap, DRS, ORB, ORB SSL and SOAP -- will be made unique by the application server.

For complex configurations with multiple members in a cluster it is advisable to make the members be as nearly identical to one another as possible, including the TCP ports. Therefore, when planning it is recommended a range of ports be allocated for a cluster with the intention to make certain all members of that cluster were given the same set of ports. Because WebSphere will automatically generate unique DRS, ORB, ORB SSL and SOAP ports for the second cluster member, it is necessary to go back in and remap the ports back to the ports set aside for the server cluster

Note: When a ″vertical cluster″, two members on the same MVS image, is the potential configuration, you will need to consider port sharing by two members of the same cluster on the same MVS image.

Workload management (WLM)

This concept is an explanation of how WebSphere Application Server for z/OS uses the workload management (WLM) function of z/OS to start and manage servers in response to workload activity.

Each J2EE application server in a WebSphere Application Server for z/OS cell uses WLM to start servants as WLM application environments. Thus, each application server must be associated with a WLM application environment name. The ″Cluster transition name″ in the WebSphere Application Server for z/OS configuration is used as the WLM application environment name.

Choosing between a stand-alone or Network Deployment configuration

A table is presented that contains specifics on the differences between a WebSphere Application Server for z/OS stand-alone cell and Network Deployment cell.

	Stand-alone cell	**Network Deployment cell**
Configuration:	Set up each stand-alone server node through the ISPF Customization Dialog. Set up additional servers within the node through the administrative console or scripting.	Set up each deployment manager node through the ISPF Customization Dialog. Add application server nodes to the Network Deployment cell through the ISPF Customization Dialog.
Address spaces:	Minimum: four (location service daemon, controller, servant, control region adjunct)	Minimum: seven (location service daemon, application server controller, application server servant, application server control region adjunct, deployment manager controller, deployment manager servant, node agent)
	Maximum: Limited only by resources.	Maximum: Limited only by resources.
Administrative isolation:	Each stand-alone server node is a separate administrative domain.	All nodes in the cell are in the same administrative domain.
Operational isolation:	You can start and stop servers independently. Each server has an independent, unshared JNDI namespace.	You can start and stop servers independently. The JNDI namespace is shared among all servers in the cell.
Application servers allowed to have multiple servants?	Yes	Yes
Clustering available?	No	Yes

Application server naming convention plan strategy

This article identifies the names you must define during WebSphere Application Server for z/OS configuration and provides a recommended naming convention.

Each WebSphere Application Server for z/OS cell, node, server, and cluster must have both a long name and a short name.

- Long names are the principal names by which cells, nodes, servers and clusters are known to WebSphere Application Server for z/OS. These are the names used in scripting and the administrative console. Long names can be up to 50 characters long and include mixed case alphabetic characters, numeric characters, and the following special characters: ! ^ ( ) _ - . { } [ ]
- Short names are specific to the z/OS implementation of WebSphere Application Server and are the principal names by which cells, nodes, servers and clusters are known to z/OS. (Note that z/OS has an eight-character limit on many operating system interface values.) Short names must be one to eight characters long, use only uppercase alphabetic, numeric and national characters, and cannot begin with a numeric character.

While it is possible to assign names to WebSphere Application Server for z/OS objects on an ad-hoc basis, it is safer and more efficient to assign names in an orderly fashion.

Note: The WebSphere Application Server for z/OS customization process provides default values for all these names. We recommend that you use these default names only when building a ″practice″ application server configuration.

The recommended naming convention given here is based on the one given in the IBM Washington Systems Center white paper, ″Sample WebSphere for z/OS ND Configuration.″ Find this document on the Web at http://www.ibm.com/support/techdocs--search for document number WP100367 under the category ″White Papers.″

Assumptions

A stand-alone application server is very simple. It has a basic cell and node structure with a single application server on a single z/OS system. (Actually, it is possible to add additional application servers to a stand-alone node, but you cannot manage them from the administrative console. This makes such a configuration unwieldy.)

A Network Deployment cell includes a deployment manager (a node of its own), together with any number of application server nodes. Each application server node contains a node agent and any number of application servers. The deployment manager can reside on the same z/OS system as one of the application server nodes or on a z/OS system of its own.

- **The recommended naming convention assumes that there is only one node per cell on a given z/OS system.** This is not really a functional limitation because having multiple nodes from the same cell on the same z/OS system does not provide any better administrative separation, and only provides slightly more reliability, than having a single node, while incurring additional overhead.
- **The recommended naming convention allows you to cluster any application server.** Creation of a cluster begins with either an existing application server that is then cloned or a newly-defined application server that is then cloned. The cloning process adds new, identical application servers on other application server nodes in the cell. To use the recommended naming convention, follow these steps:
  1. Select a two-character cell identifier *cc* for the WebSphere Application Server for z/OS cell you plan to create.
  2. Select a one-character system identifier *x* for each z/OS system that will support some part of the cell.
  3. Select a two-digit server identifer *nn* for each application server you create. Use a different value of *nn* for each application server in the cell. (This will allow clustering of any application server later on.) Use ″01″ for a stand-alone application server.

Cell long name and short name

Each WebSphere Application Server for z/OS cell, whether a stand-alone application server or a Network Deployment cell, must have a long name and a short name. The cell long name is arbitrary, but the cell short name must be unique in each configuration HFS.

Recommendations:

- Set the cell short name to ″*cc*CELL″ in uppercase (″AZCELL″ for example).
- Set the cell long name to be the same as the cell short name but lowercase (″azcell″ for example).
- A cell name must be unique in any circumstance in which the product is running on the same physical machine or cluster of machines, 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.NameNotFoundException exception, in which case, you need to create uniquely named cells.

Node long name and short name

Each WebSphere Application Server for z/OS node must have a long name and a short name. A stand-alone application server has just one node, though it keeps its node name if it is federated into a Network Deployment cell. A Network Deployment cell has a deployment manager node and a number of application server nodes.

Note: As noted above, we assume that a cell has only one application server node on each z/OS system.

The node long name and short name must be unique within the cell. The node short name of a stand-alone application server intended for federation into a Network Deployment cell should also be unique within the Network Deployment cell.

Recommendations:

For a deployment manager node:

- Set the node short name to ″*cc*DMGR″ in uppercase (″AZDMGR″ for example).
- Set the node long name to be the same as the node short name but lowercase (″azdmgr″ for example).

For an application server node:

- Set the node short name to ″*cc*NODE*x*″, where *x* is the system identifier (″AZNODEA″ for example).
- Set the node long name to be the same as the node short name but lowercase (″aznodea″ for example).

Server long name and short name

Each WebSphere Application Server for z/OS server must have a long name and a short name.

The server long name and short name must each be unique within the node and should not exist in any other node to which you might clone this server to form a cluster. In the Customization Dialog, server short names are limited to seven characters.

Recommendations:

For a deployment manager server:

- Set the server short name to ″*cc*DMGR″ in uppercase (″AZDMGR″ for example).
- The server long name is ″dmgr″ (unchangeable).

For a node agent:

- Set the server short name to ″*cc*AGNT*x*″ (″AZAGNTA″ for example).
- The server long name is ″nodeagent″ (unchangeable).

For an application server node:

- Set the server short name to ″*cc*SR*nnx*″, where *nn* is the server identifier and *x* is the system identifier (″AZSR01A″ for example).
- Set the node long name to be the same as the node short name but lowercase (″azsr01a″ for example).

Jobnames

Each server controller, servant and control region adjunct (CRA) has an MVS jobname, set by the Customization Dialog:

- The controller jobname is the same as the server short name (″AZSR01A″ for example).
- The servant jobname is the server short name with an ″S″ appended (″AZSR01AS″ for example).
- The control region adjunct jobname is the server short name with an ″A″ appended (″AZSR01AA″ for example).

These values cannot be changed in the Customization Dialog. This is also why the dialog limits server short names to seven characters--to allow room for the final character that differentiates a servant or CRA from the controller.

It is possible to change any server jobname using the administrative console. Because of this, for example, it is possible to change server short names to be eight characters long. However, this would require you to make corresponding changes to RACF profiles and so on.

The location service daemon

The location service daemon is a special server with no servant or CRA. Each cell must have one location service daemon running on each z/OS system on which the cell is configured. If a server is started and finds that its location service daemon is not running, it starts the location service daemon before continuing its processing.

A location service daemon has a jobname, which is also used as its server short name.

Note: A location service daemon has no server long name.
All location service daemons in a cell use the same jobname.

A stand-alone application server, like any cell, has a location service daemon of its own. However, when the stand-alone application server is federated into a Network Deployment cell, the newly-created application server node in the Network Deployment cell uses its own location service daemon--one with the same jobname and TCP/IP ports as the deployment manager's location service daemon--and the location service daemon originally used by the stand-alone application server is abandoned.

Recommendations:

- For a Network Deployment cell, set the location service daemon jobname to ″*cc*DEMN″ (″AZDEMN″ for example).
- For a stand-alone application server, set the location service daemon jobname to ″*cc*DEMN*mm*″, where *mm* is chosen to make the location service daemon jobname unique on the z/OS system.

Cluster transition name

Each controller that has servants uses Workload Management (WLM) to start the servants. To do this, it needs a WLM application environment name.

- For an unclustered application server (or deployment manager), the cluster transition name is used as the WLM application environment name.
- When a cluster is created, it is assigned a cluster short name. If the cluster is based on an existing initial server, then the cluster short name is set to the server's cluster transition name. If the cluster is based on a new initial server, then the cluster short name is set to BBOC00*n*, where *n* is a number incremented as needed to maintain uniqueness in the cell. The cluster short name is used as the WLM application environment name for all servers in the cluster.

Therefore, the cluster transition name is used as the WLM application environment name for a stand-alone application server, for the same server after it is federated (or created from scratch, unclustered) in a Network Deployment cell, or for an entire server cluster based on the original server.

When you assign a cluster transition name to a server, it must be unique in the cell.

Recommendations:

- For a deployment manager, set the cluster transition name to "*cc*DMGR". This is the same as the server short name.
- For an application server, set the cluster transition name to "*cc*SR*nn*" ("AZSR01" for example). This is the same as the server short name without the system identifier.

  Note: This also illustrates why a different two-digit server identifier is used for each server. Suppose you had two servers AZSR01A and AZSR01B. If you create a cluster based on AZSR01A and want to extend it to the B system, the obvious server name choice is AZSR01B - which is already taken. But if the servers are named AZSR01A and AZSR02B, then you can cluster either and extend the cluster to the other system without conflict. The respective cluster short names would be AZSR01 and AZSR02.

Cataloged procedures

Each WebSphere Application Server for z/OS server uses a JCL cataloged procedure. These procedures are all fairly similar and consist of a main cataloged procedure and an INCLUDE member that contains DD statements. Here are sample cataloged procedure library members for a controller as generated by the Customization Dialog:

Procedure library member BBO6ASR:

```
//BBO6ASR  PROC ENV=,Z=BBO6ASRZ
//  SET ROOT='/WebSphere/V6R1'
//BBOSR   EXEC PGM=BBOSR,REGION=0M,TIME=NOLIMIT,
// PARM='TRAP(ON,NOSPIE),ENVAR("_EDC_UMASK_DFLT=007") /'
//BBOENV DD PATH='&ROOT/&ENV/was.env'
//  INCLUDE MEMBER=&Z
```

Procedure library member BBO6ASRZ:

```
//*
//* Output DDs
//*
//CEEDUMP   DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
//SYSOUT    DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
//SYSPRINT  DD SYSOUT=*,SPIN=UNALLOC,FREE=CLOSE
//*
//*Steplib Setup
//*
//STEPLIB  DD DISP=SHR,DSN=WAS.V61.SBBOLD2
//         DD DISP=SHR,DSN=WAS.V61.SBBOLOAD
```

The `PGM=` parameter on the `EXEC` statement in the cataloged procedure specifies the type of WebSphere Application Server for z/OS server. The program names are as follows:

- BBOCTL (controller)

- BBOSR (servant)
- BBOCRA (adjunct process)
- BBODAEMN (location service daemon)

The PARM= parameter on the EXEC PGM statement contains the parameters that are passed to the program identified by the PGM= parameter.

Customization Note: The appropriate interface for making changes to the language environment (LE) parameters is through the PARM= parameter; however, do not modify any LE parameters without first consulting with the IBM support team. The LE parameters are set internally to ensure the best possible performance of the WebSphere Application Server, which is the main LE application running in the address space. If you need to add or change LE parameters, make sure that you work with the IBM WebSphere support team to ensure that the internally set parameters are not compromised.

The BBOENV DD statement points to the was.env (startup parameter) file for the server. The path to this file consists of the configuration HFS directory name (hardcoded using the ROOT JCL variable) and the symbolic link for this particular server, which is specified at startup using the ENV= parameter.

The INCLUDE statement in the main cataloged procedure causes the incorporation of all JCL statements from the INCLUDE member (in this case, BBO6ASRZ). By convention, the INCLUDE member's name consists of the name of the main cataloged procedure with the letter ″Z″ appended to the end. The INCLUDE member provides common DD statements for the started task.

The STEPLIB DD statement for the location service daemon determines the WebSphere Application Server for z/OS code base intended for use. If the location service daemon finds the SBBOLPA modules in its STEPLIB concatenation, it loads them into common storage. If the modules are not found in STEPLIB, then the location service daemon checks for their presence in the system link pack area.

Note: If the modules are found in neither location, the location service daemon reports an error and terminates.

Once the modules are located (and loaded if necessary), the location service daemon holds a vector table of load module addresses that the rest of the cell's servers will use.

All other servers for the location service daemon's cell on the same z/OS system must use the same product code base as the location service daemon:

- Each server STEPLIB DD statement (in the INCLUDE JCL member) should have the same WebSphere Application Server for z/OS data sets as listed in the location service daemon's STEPLIB DD statement.

  Note: The server STEPLIB DD concatenation can also include additional DD statements for CICS or DB2 libraries and so on.
- Each server's home directory must point to a product HFS that matches the service level of the location service daemon's WebSphere Application Server for z/OS load modules.

The controller cataloged procedure includes some additional statements before the EXEC statement that invokes BBOCTL. These are used to invoke the post installer program that applies any needed maintenance to the configuration HFS and its home directories when service is applied to the product HFS and load modules.

The following sections describe the cataloged procedures required for each configuration, provide a recommended naming convention, and explain how the SAF user ID for each server is determined.

Cataloged procedures for stand-alone application servers

A stand-alone application server uses the following cataloged procedures with corresponding INCLUDE JCL members for each:

- Controller cataloged procedure
- Servant cataloged procedure
- Adjunct process cataloged procedure
- Location service daemon cataloged procedure

You can use the same cataloged procedures for different stand-alone servers IF the configuration HFS and product code level (including STEPLIB) are the same for both servers. The Customization Dialog will overwrite identically named proclib members if identical names are chosen or if customization jobs and files are regenerated. If you have a local STEPLIB or make other changes to the procedures, therefore, make sure that they are reapplied after you use the Customization Dialog.

Cataloged procedures for Network Deployment cells

A Network Deployment cell uses the following cataloged procedures, with corresponding INCLUDE JCL members for each:

For the deployment manager:

- Deployment manager controller cataloged procedure
- Deployment manager servant cataloged procedure

For each application server node:

- Application server controller cataloged procedure (also used for the node agent)
- Application server servant cataloged procedure

For the location service daemon (one per z/OS system):

- Location service daemon cataloged procedure

The application server servant cataloged procedure is the only one likely to require modification, in order to place libraries (CICS, DB2, and so on) in the STEPLIB concatenation.

You can use the same cataloged procedures for several nodes in a Network Deployment cell, or even for several cells, IF the configuration HFS and product code level (including STEPLIB) are the same for both servers. The Customization Dialog will overwrite identically-named proclib members if identical names are chosen or if customization jobs and files are regenerated. If you have a local STEPLIB or make other changes to the procedures, therefore, make sure that they are reapplied after you use the Customization Dialog.

A recommended cataloged procedure naming convention

Use a consistent naming convention for your WebSphere Application Server for z/OS cataloged procedures. The procedure name should distinguish between WebSphere Application Server for z/OS version, configuration HFS, and which data sets are listed in STEPLIB. Limit cataloged procedure names to seven characters to allow the Customization Dialog to create INCLUDE JCL member names by appending the letter ″Z.″

For example, the following convention works for either a stand-alone application server or Network Deployment cell, where *cc* is a two-character cell identifier:

Deployment manager controller	*cc*6DCR
Deployment manager servant	*cc*6DSR

Location service daemon controller	*cc*6DMN
Application server controller	*cc*6ACR
Application server servant	*cc*6ASR
Control region adjunct	*cc*6CRA

If you require separate cataloged procedures for nodes on different systems in a sysplex (if they need independently settable STEPLIB statements to allow for a nondisruptive restart for example), either place the location service daemon and application server procedures in system-specific proclibs, or append a one-character system identifier to the cataloged procedure names for the location service daemon and application servers.

Assigning user IDs to WebSphere Application Server for z/OS address spaces

If you use z/OS Security Server (RACF) as your SAF-compliant security system on z/OS, then STARTED class profiles are used to assign started task user IDs to each WebSphere Application Server for z/OS server. These STARTED profiles are set up by the Customization Dialog batch jobs. Update these STARTED profiles as needed to place servers you create yourself under the appropriate user IDs.

Controllers (deployment manager, location service daemon, node agent or applications server controller) are started using a console `START` command that you issue either from the MVS console or internally. For these servers, the STARTED profile name that is checked is of the form `procname.jobname`.

Whenever it creates a controller or daemon cataloged procedure, the Customization Dialog also creates a STARTED profile that associates all controllers using that cataloged procedure with the appropriate controller user ID and configuration group. Therefore, if you set up a stand-alone application server with default names, the Customization Dialog would create the following STARTED profiles for controllers:

- `RDEFINE STARTED BBO6ACR.* STDATA(USER(ASCR1) GROUP(WSCFG1) TRACE(YES))`
- `RDEFINE STARTED BBO6DMN.* STDATA(USER(WSDMNCR1) GROUP(WSCFG1) TRACE(YES))`

Note: `TRACE(YES)` writes message IRR812I to the MVS console whenever the profile is used.

Servant regions (application server servants and adjunct processes) are started using Workload Manager (WLM). For these servers, the STARTED profile name that is checked is of the form `jobname.jobname`.

Unfortunately, there is no way to assign all servers using a particular servant cataloged procedure to a servant user ID. Therefore, the Customization Dialog creates both a generic STARTED profile for servants, based on the jobname prefix BBO, and a specific STARTED profile for any servant whose jobname does not begin with BBO, that assign their servants to the servant user ID, and a specific STARTED profile for the adjunct process that assigns it to the CRA user ID. If default names are chosen, the following servant STARTED profiles are created for a stand-alone application server:

- `RDEFINE STARTED BBO*.* STDATA(USER(ASSR1) GROUP(WSCFG1) TRACE(YES))`

  Note: There is no specific application servant profile because the server name BBOS001S is "caught" by this profile.
- `RDEFINE STARTED BBOS001A.* STDATA(USER(ASCRA1) GROUP(WSCFG1) TRACE(YES))`

When you choose cataloged procedure names, make sure that the appropriate STARTED profile is in place to map the server to its appropriate SAF user ID. Use the RACF ISPF panels or the `RLIST STARTED` command to display the STARTED profiles.

If you use another SAF-compliant security system, contact the security server vendor for WebSphere Application Server for z/OS setup information.

Cataloged procedure for the administrative asynchronous task

The asynchronous administrative task also requires a cataloged procedure. This very simply cataloged procedure does not include a STEPLIB or configuration HFS pointer. However, since it must run under a specific user ID and group associated with the security domain of the cell for which it runs, you must choose a different cataloged procedure name for each security domain or cell.

Recommendation:

Name the administrative asynchronous task cataloged procedure ″*cc*6SH,″ where *cc* is a two-character cell identifier.

Planning for TCP/IP port convention

This article lists the default server values for WebSphere Application Server for z/OS.

z/OS port assignments

The following table lists the default port assignments for z/OS.

Port	Stand-alone location service daemon	Stand-alone application server	ND location service daemon	ND application server	Node agent	Deployment manager
HTTP		9080		9080		
HTTP/S		9443		9443		
Admin Console Port	9060			9060		9060
Admin Console Secure Port	9043			9043		9043
Bootstrap		2809		9810	2809	9809
ORB	5655	2809	5755	9810	2809	9809
ORB SSL	5656	0	5756	0	0	0
SOAP/JMX		8880		8880	9360	8879
Node Discovery					7272	
Node Multicast Discovery					5000	
Cell Discovery						7277
Service Integration		7276				
Service Integration Secure		7286				
Service Integration MQ Interoperability		5558				

Port	Stand-alone location service daemon	Stand-alone application server	ND location service daemon	ND application server	Node agent	Deployment manager
Service Integration MQ Interoperability Secure		5578				
Session Initiation Protocol Port		5060				
Session Initiation Protocol Secure Port		5061				
High Availability Manager Communications		9353			9354	9352

Location service daemon ports

Stand-alone application server node location service daemons are considered temporary. The ports assigned to a stand-alone application server node's location service daemon are used only until that node is federated. It is advisable to set aside a couple of ports to serve as ″interim ports″ for the stand-alone application server node location service daemon. The ″permanent″ location service daemon ports are the ones assigned to the deployment manager. Those same ports are copied to location service daemons created when a stand-alone application server node on another MVS image is federated into the deployment manager cell.

Node agent ports

There is a node agent per MVS image on which the cell spans. One design option calls for all node agents to have the exact same ports so the Sysplex Distributor is able to balance the traffic between the two. The node agent is created when the BBOWADDN customized job is run.

Server clusters

A server cluster is a grouping of two or more servers into a one logical server. A cluster is created through the administrative console. Servers within a cluster are called ″cluster members.″ Servers (″members″) within a cluster start out being clones of one another. When it comes to the TCP ports for the members in a cluster, the administrative console allows you during the creation of the cluster to specify if you want the HTTP ports to be unique or the same. The other ports -- bootstrap, DRS, ORB, ORB SSL and SOAP -- will be made unique by the application server.

For complex configurations with multiple members in a cluster it is advisable to make the members be as nearly identical to one another as possible, including the TCP ports. Therefore, when planning it is recommended a range of ports be allocated for a cluster with the intention to make certain all members of that cluster were given the same set of ports. Because WebSphere will automatically generate unique DRS, ORB, ORB SSL and SOAP ports for the second cluster member, it is necessary to go back in and remap the ports back to the ports set aside for the server cluster

Note: When a ″vertical cluster″, two members on the same MVS image, is the potential configuration, you will need to consider port sharing by two members of the same cluster on the same MVS image.

Planning for security

A Common MVS Group and User definition in z/OS provides WebSphere Application Server for z/OS with a set of cell-wide z/OS Security Server (RACF) security definitions.

Note: The cells involved might be stand-alone application server cells, Network Deployment cells, or both. A Common MVS Group and User definition includes the following:

- Common definitions for administrative users and groups
- SAF groups to which servers belong

You can use multiple Common MVS Groups and Users to isolate WebSphere Application Server for z/OS cells from one another within a sysplex. For example, you should create separate common MVS Groups and Users for cells that require different groups of users to have administrative authority.

During installation you now have the option of enabling administrative security during initial cell customization, this procedure is referred to as *security out of the box*. This protects the cell from unauthorized modification, which can occur if security is not enabled. To enable security during installation, refer to “Initial security configuration” for more information.

The common MVS Groups and Users definition in z/OS provides WebSphere Application Server for z/OS with a set of cell-wide z/OS Security Server (RACF) definitions, these definitions will be required during installation.

1. Print a copy of “Customization Dialog worksheet: Common MVS Groups and Users” on page 81.
2. Fill out the worksheet as described in “Customization Dialog variables: Common MVS Groups and Users” on page 80.
3. Save the worksheet for use during the customization.

Initial security configuration

During installation you now have the option of enabling administrative security during initial cell customization, this procedure is referred to as ″security out of the box″. This protects the cell from unauthorized modification, which can occur if security is not enabled.

When a new stand-alone application server or Network Deployment cell is created, there are three initial security choices in WebSphere Application Server for z/OS Version 6.1:

- Use a z/OS security product to manage user identities and authorization policy
- Use WebSphere Application Server to manage user identities and the authorization policy
- Do not enable security

This article describes the three initial security options and the configuration effects of each.

Remember that WebSphere Application Server for z/OS always requires the presence of a SAF-compliant security system to provide operating system security. Regardless of which security option is chosen:

- SAF user IDs for WebSphere Application Server started tasks are always created during customization.
- SAF groups are created for the configuration, servant and local user groups are created during customization, and granted necessary permissions
- SAF SERVER profiles are used to control servant access to controller regions.
- If daemon SSL is selected during customization, a keyring and digital certificate for the daemon are created in SAF.

Note: Each of the initial security configurations is basic, requiring few choices during customization; after configuration is complete, additional work is usually required to match cell security policies to the needs of the enterprise. See the Security section of the InfoCenter for more information.

Option 1: Use a z/OS security product to manage user identities and authorization policy

If this option is chosen during customization:

1. Each WebSphere Application Server user and group identity corresponds to a user ID or group in the z/OS system's SAF-compliant security system (IBM'S RACF, or an equivalent product).
2. Access to WebSphere Application Server roles is controlled using the SAF EJBROLE profile.
3. Digital certificates for SSL communication are stored in the z/OS security product.

The z/OS system's security product is always used to control WebSphere Application Server for z/OS started task identities, and the location service daemon's digital certificate (if daemon SSL is selected). However, when this security option is selected, all WebSphere Application Server administrators and administrative groups must be defined to SAF as well. Later, if application security is enabled, the SAF security database holds those user identities as well.

This option is appropriate when servers or cells will reside entirely on z/OS systems, with SAF as the user registry. Customers who plan to implement an LDAP or custom user registry, but who will map WebSphere Application Server identities to SAF identities and use EJBROLE profiles for authorization, should also choose this option so that initial SAF EJBROLE setup is performed.

When this option is chosen during customization, the following SAF user IDs are created:

- An administrator user ID
- An ″unauthorized user″ ID, to represent WebSphere Application Server identities which have not been authenticated

SAF EJBROLE profiles for administrative roles (administrator, configuration, deployer, monitor and operator) are created, and the administrator user ID is granted the administrator role.

SAF CBIND profiles are created, and granted to the configuration group.

Digital certificates are created in the SAF security system for each server controller (deployment manager or application server controller).

Digital keyrings are created in the SAF security system for the administrator, controller, controller region adjunct, and server user IDs, and the appropriate certificates are attached to these keyrings.

A security domain name may be specified when this option is chosen; the security domain name becomes part of the APPL, CBIND and EJBROLE profile names used for authorization checking.

Option 2: Use WebSphere Application Server to manage user identities and authorization policy

If this option is chosen during customization:

1. Each WebSphere Application Server user and group identity corresponds to an entry in a WebSphere Application Server user registry. The initial user registry is a simply file-based user registry, created during customization, and residing in the configuration file system.
2. Access to WebSphere Application Server roles is controlled using WebSphere Application Server role bindings. In particular, administrative roles are controlled using the ″Console users and groups″ settings in the administrative console.
3. Digital certificates for SSL communication are stored in the configuration file system.

The z/OS system's security product is always used to control WebSphere Application Server for z/OS started task identities, and the location service daemon's digital certificate (if daemon SSL is selected). However, when this security option is selected, all WebSphere Application Server users and groups for administrative access are defined in the WebSphere user registry, rather than in SAF. Later, if application security is enabled, the WebSphere Application Server user registry holds those user identities as well.

This option is appropriate when servers or cells will reside on a mix of z/OS and non-z/OS systems, as well as for customers who plan to implement an LDAP or custom user registry to replace the initial registry. (Customers who plan to implement an LDAP or custom user registry with identity mapping to SAF should select z/OS-managed security during customization; see above.)

When this option is chosen during customization, a file-based user registry is created in the configuration file system.

An administrator user ID (and an optional samples user ID and group) are added to the file-based user registry.

The administrator user ID is added to the list of authorized console users.

Self-signed digital certificates for servers are created in the configuration file system automatically by WebSphere Application Server.

Option 3: Do not enable security

If this option is chosen, no administrative security is configured. Anyone with access to the administrative console port can make changes to the server or cell configuration.

A post-customization security setup is recommended.

The initial security setup options in WebSphere Application Server are very basic, and are intended only to provide initial administrative security. After your server or cell is up and running, you may wish to:

- Switch to another user registry. You can use LDAP or a custom user registry instead of the SAF security database or file-based registry.
- Define additional administrators, or distribute administrative roles
- Implement application security

Customization Dialog variables: Common MVS Groups and Users

This topic lists definitions for the terms that you will see in the WebSphere Application Server for z/OS Customization Dialog.

Note: If you are setting up a ″practice″ stand-alone application server, choose the default values wherever possible.

Cell-wide user IDs and groups

The first part of setting up a security domain is to choose the cell-wide user IDs and group names. Each name should contain one to eight alphanumeric characters with an alphabetic first character.

Note: You can also use national characters (#, $, and @), but these are better avoided as they can lead to compatibility problems later.

Each user ID will also require a UNIX System Services UID number, and each group will require a UNIX System Services GID number:

- UID values must be unique numeric values between 1 and 2,147,483,647. Do not use a UID value of 0.
- GID values must be unique numeric values between 1 and 2,147,483,647.

Although you can set up several cells using a single security domain definition, you should not share user IDs and groups between separate security domains.

WebSphere application Server Owner user ID

This user ID owns many of the cell's files in the configuration file system. It must have the WebSphere Application Server configuration group (below) as its default UNIX System Services group.

Choose names and GID values for the following SAF groups, and enter them on the worksheet:

WebSphere Application Server Configuration Group

This is the default group name for the WebSphere Application Server administrator user ID and all server user IDs. This is the group owner for most files in the configuration HFS, so access to this group should be limited.

WebSphere Application Server Servant Group

Connect all servant user IDs to this group. You can use it to assign subsystem permissions, such as DB2 authorizations, to all servants in the security domain.

WebSphere Application Server Local User Group

Connects all local WAS client user IDs to this group. In unauthenticated user (guest) ID's are required for WebSphere Application Server, they should have this as their default group.

WebSphere Application Server user ID home directory

Note: This field was added to the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.1.

WebSphere Application Server user ID home directory

Specify a new or existing z/OS HFS directory in which home directories for WebSphere Application Server for z/OS user IDs will be created by the customization process. This directory does not need to be shared among z/OS systems in a WebSphere Application Server cell. If you use an existing world-writeable directory, you may need to manually set the permissions after WebSphere Application Server customization, which sets the permissions to 755.

Customization Dialog worksheet: Common MVS Groups and Users

Date: ______________________

____

Sysplex(es) on which you will use these common MVS groups and users:

____

Print out this worksheet and use it when collecting information about the WebSphere Application Server for z/OS Customization Dialog variables. The variables and defaults are provided along with spaces for you to fill in your own value for each.

Common MVS groups and users

Item	Default	Your value
Owner user ID	WSOWNER	

Cell-wide user IDs and groups

WebSphere Application Server Configuration Group Information		
Group	WSCFG1	
GID	2500	
WebSphere Application Server Owner Information		

User ID	WSOWNER	
UID	2405	
WebSphere Application Server Servant Group Information		
Group	WSSR1	
GID	2501	
WebSphere Application Server Local User Group Information		
Group	WSCLGP	
GID	2502	
WebSphere Application Server user ID Home Directory		
	/var/WebSphere/home	

Name of security domain configuration variable data set:

____

Planning for common groups and users

A security domain definition in z/OS provides WebSphere Application Server for z/OS with a set of cell-wide z/OS Security Server (RACF) security definitions.

Note: The cells involved might be stand-alone application server cells, Network Deployment cells, or both.
A security domain definition includes the following:

- Common definitions for administrative users and groups
- SAF groups to which servers belong

You can use multiple security domains to isolate WebSphere Application Server for z/OS cells from one another within a sysplex. For example, you should create separate security domains for cells that require different groups of users to have administrative authority.

If you plan to use SAF to control access to J2EE roles by way of the EJBROLE class, you might want to define a security domain identifier as part of the security domain definition. This causes role names to be prefixed with the security domain identifier.

1. Print a copy of “Customization Dialog worksheet: Common MVS Groups and Users” on page 81.
2. Fill out the worksheet as described in “Customization Dialog variables: Common MVS Groups and Users” on page 80.
3. Save the worksheet for use during security domain customization.

Customization Dialog variables: Common MVS Groups and Users

This topic lists definitions for the terms that you will see in the WebSphere Application Server for z/OS Customization Dialog.

Note: If you are setting up a ″practice″ stand-alone application server, choose the default values wherever possible.

Cell-wide user IDs and groups

The first part of setting up a security domain is to choose the cell-wide user IDs and group names. Each name should contain one to eight alphanumeric characters with an alphabetic first character.

Note: You can also use national characters (#, $, and @), but these are better avoided as they can lead to compatibility problems later.

Each user ID will also require a UNIX System Services UID number, and each group will require a UNIX System Services GID number:

- UID values must be unique numeric values between 1 and 2,147,483,647. Do not use a UID value of 0.
- GID values must be unique numeric values between 1 and 2,147,483,647.

Although you can set up several cells using a single security domain definition, you should not share user IDs and groups between separate security domains.

WebSphere application Server Owner user ID
This user ID owns many of the cell's files in the configuration file system. It must have the WebSphere Application Server configuration group (below) as its default UNIX System Services group.

Choose names and GID values for the following SAF groups, and enter them on the worksheet:

WebSphere Application Server Configuration Group
This is the default group name for the WebSphere Application Server administrator user ID and all server user IDs. This is the group owner for most files in the configuration HFS, so access to this group should be limited.

WebSphere Application Server Servant Group
Connect all servant user IDs to this group. You can use it to assign subsystem permissions, such as DB2 authorizations, to all servants in the security domain.

WebSphere Application Server Local User Group
Connects all local WAS client user IDs to this group. In unauthenticated user (guest) ID's are required for WebSphere Application Server, they should have this as their default group.

WebSphere Application Server user ID home directory

Note: This field was added to the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.1.

WebSphere Application Server user ID home directory
Specify a new or existing z/OS HFS directory in which home directories for WebSphere Application Server for z/OS user IDs will be created by the customization process. This directory does not need to be shared among z/OS systems in a WebSphere Application Server cell. If you use an existing world-writeable directory, you may need to manually set the permissions after WebSphere Application Server customization, which sets the permissions to 755.

Customization Dialog worksheet: Common MVS Groups and Users

Date: ______________________

____

Sysplex(es) on which you will use these common MVS groups and users:

____

Print out this worksheet and use it when collecting information about the WebSphere Application Server for z/OS Customization Dialog variables. The variables and defaults are provided along with spaces for you to fill in your own value for each.

Common MVS groups and users

Item	Default	Your value
Owner user ID	WSOWNER	

Cell-wide user IDs and groups

WebSphere Application Server Configuration Group Information		
Group	WSCFG1	
GID	2500	
WebSphere Application Server Owner Information		
User ID	WSOWNER	
UID	2405	
WebSphere Application Server Servant Group Information		
Group	WSSR1	
GID	2501	
WebSphere Application Server Local User Group Information		
Group	WSCLGP	
GID	2502	
WebSphere Application Server user ID Home Directory		
	/var/WebSphere/home	

Name of security domain configuration variable data set:

Building a practice WebSphere Application Server for z/OS cell

Use this task to practice configuring WebSphere Application Server for z/OS. If you are installing the product for the first time without migration from an earlier version, it is helpful to install a practice application serving environment in order to learn the customization process.

Note the following when you install your practice runtime:

- Be careful when typing and following the instructions in the customization.
- Note the user ID requirements in the generated instructions. If your user IDs are not configured correctly, the jobs might not run successfully.
- Keep track of each step so that you do not skip or repeat any steps.
- Examine each job’s output (not only the return code) to confirm that everything is configured correctly. Sometimes, the return code indicates no problems but the job output contains errors. For a proper configuration, you should have no errors in your job output unless the instructions specifically describe errors in the job output.

You should install a practice runtime when you install WebSphere Application Server on z/OS for the first time and want to learn the steps for installing and customizing it.

Install using either the ISPF Customization Dialogs or the Profile Management tool.

1. Print a copy of the “Customization Dialog worksheet: Common MVS Groups and Users” on page 81 and fill it out using “Customization Dialog variables: Common MVS Groups and Users” on page 80 as a guide. When filling out the worksheet, use default values wherever possible.

   Note: Make sure that the user ID names, group names, and UID/GID values that you specify are not already being used on your z/OS system.

2. Print a copy of the “Customization Dialog worksheet: Stand-alone application server cell” on page 96 and fill it out using “Customization Dialog variables: Stand-alone application server cell” on page 87 as a guide.

   Note: Make sure that the user ID names, group names, UID/GID values, and TCP/IP port numbers that you specify are not already being used on your z/OS system.
3. Read the topic in “Using the Customization Dialog” on page 133.
4. Read the topic in “Using the Profile Management tool” on page 225.
5. Follow the steps in“Creating common MVS groups and users” on page 138.
6. Follow the steps in “Creating a stand-alone application server cell” on page 227. View and follow the generated instructions, which tell you how to:
   - Perform the manual configuration updates in the generated stand-alone application server instructions. These steps affect parts of your system that are usually controlled. These are changes that the systems programmer responsible for your z/OS system should review.
   - Create your log streams and data sets. The first four jobs (BBOMSGC - BBOWCTR) are straightforward system-programming tasks.
   - Update your server-specific security definitions. The next two jobs (BBOCBRAJ and BBOCBRAK) generate and then issue the RACF commands necessary for defining the users, groups, profiles, and permissions for the WebSphere Application Server for z/OS runtime servers.
   - Submit the BBOCBRAK job, or take it and the BBOCBRAJ job to your security administrator for approval. Your security administrator should issue those commands or submit the supplied jobstreams. If your installation has a different profile structure, you might have to modify the RACF commands generated by these execs to suit your particular needs.

     Note: Your installation must have ″list of groups″ on for these commands to work because the servers must be connected to the WebSphere Application Server for z/OS administrator group.
   - Create the configuration file system and WebSphere Application Server for z/OS home directory for your server. Jobs BBOWCHFS and BBOWHFSA (run at this point) and job BBOWWPFA (see below) run BPXBATCH shell scripts to define, customize, and load data into the configuration HFS and manipulate the ownership and permission attributes. For this reason, you must run these jobs under a user ID with UID=0.
   - Create cataloged procedures for the server and, optionally, for RRS and component trace.
   - Set up the runtime (configuration) file system for the new application server. The BBOWWPFA job might run for some time. The BBOWHFSB job cleans up the configuration file system and makes sure that all file ownerships are correct.
   - Start the new stand-alone application server, and run the Install Verification Test (IVT).

   Note:

      - If the BBOWWPFA (profile creation) job fails with the following error:

        ```
        Cannot use the directory: The /WebSphere/V6R1M0/AppServer/profiles/default directory exists and is not empt
        INSTCONFFAILED: Cannot create profile: The profile does not exist.
        ```

        delete the WAS_HOME/profiles/default directory and all its contents before rerunning BBOWWPFA.
      - After you get the application server up and successfully run the IVT, you might want to change the built-in administrators defined for WebSphere Application Server for z/OS or at least change their passwords. If you do not want the user IDs assigned to started tasks used to sign on or submit jobs, make the user IDs protected by adding the keywords ″NOPASSWORD″ and ″NOOIDCARD″ to each user’s RACF profile.
      - If you have some other security product such as Top Secret or ACF2 instead of RACF, contact your security system vendor for the appropriate security system commands needed

to configure WebSphere Application Server for z/OS. You might need to contact the vendor for the latest maintenance and guidance on WebSphere Application Server for z/OS customization.

7. Troubleshoot any problems you encounter while customizing your application server.

   If you encounter problems while customizing your application server, review the steps that you have performed--especially regarding such things as specific user IDs under which jobs must be run. Check all job output for any error messages that you might have missed. See “Troubleshooting installation and configuration” on page 248 for additional advice.

   Watch out for these common mistakes:

   - Navigating the configuration file system with a UID of ″0″ can alter files or their ownership and permission attributes, making them inaccessible to the WebSphere Application Server for z/OS runtime servers and administrators. To avoid this problem, use the WebSphere Application Server for z/OS administrator user ID.
   - If you decide to change any of the customized variables after you submit any of these jobs, do not make manual modifications to the generated jobstreams or data. Cancel the installation, and start over by regenerating all the jobstreams and start over from the BBOWHFSA job.

After you have successfully followed the instructions, you will have set up a WebSphere Application Server for z/OS stand-alone application server. See “Working with your new server” on page 174 for more information.

Read the concept articles under “Planning for product configuration” on page 51 and plan one or more application serving environments that fit your system environment and business needs.

You might want to delete the practice application server that you just set up in order to save space on your system, to clean up your data sets, or for other reasons. Follow these steps to delete it from your system:

1. Stop the server.
2. Unmount and delete the configuration file system.
3. Delete the cataloged procedures.
4. Remove any TCP/IP port reservations for the practice application server.
5. Delete the RACF user IDs, groups, and profiles that you have created unless you use the same security domain for a different WebSphere Application Server for z/OS cell.

Planning for a stand-alone application server cell

Decide on the security standards that you plan to use for this cell and complete the “Customization Dialog worksheet: Common MVS Groups and Users” on page 81.

A stand-alone application server cell is the simplest WebSphere Application Server for z/OS configuration on which you can deploy and run applications. A stand-alone application server cell includes the following:

- A basic cell and node configuration
- A location service daemon
- An application server that runs the administrative console application. You can deploy and run additional applications on this server.

You cannot add additional servants to an application server running the stand-alone version of the administrative console application. You can define additional application servers in the stand-alone cell, but you cannot control them using the administrative console. For more complicated or robust WebSphere Application Server for z/OS application-serving environments, the Network Deployment cell configuration is recommended.

If you have never configured a WebSphere Application Server for z/OS cell, try “Building a practice WebSphere Application Server for z/OS cell” on page 84 first.

1. Print a copy of “Customization Dialog worksheet: Stand-alone application server cell” on page 96.
2. Fill out the worksheet as described in “Customization Dialog variables: Stand-alone application server cell.”
3. Save the worksheet for use during stand-alone application server customization.

Customization Dialog variables: Stand-alone application server cell

Definitions for z/OS Customization Dialog variables are covered in this article.

This article lists definitions for the terms you see in the WebSphere Application Server for z/OS Customization Dialog.

The panels corresponding to the following tables set up the names, network configuration, start procedures, and user IDs for a stand-alone server.

Recommendation: Use the IBM default names the first time you install WebSphere Application Server for z/OS to make the installation instructions easier to follow.

System Locations

This section identifies the target z/OS system on which you will configure the new stand-alone application server, along with system data set names.

System name

The system name for the target z/OS system on which you will configure WebSphere Application Server for z/OS.

Sysplex name

The sysplex name for the target z/OS system on which you will configure WebSphere Application Server for z/OS.

Tip: If you are not sure what the system name (&SYSNAME) and sysplex name (&SYSPLEX) are, use the console command `D SYMBOLS` on the target z/OS system to display them.

For the following, specify the fully qualified data set names without quotes.

Rule: You can specify up to 44 characters for the data set names.

PROCLIB

An existing procedure library where the WebSphere Application Server for z/OS cataloged procedures are added.

SYSEXEC

A variable-block (RECFM=VB, LRECL=255) data set into which the customization process places REXX EXECs to be called from TSO, such as the WebSphere Application Server for z/OS error log browser, BBORBLOG. You must allocate this data set and concatenate it as part of the SYSEXEC DD allocation in your installation-wide TSO logon PROC or allocation exec.

If your existing SYSEXEC DD data set concatenation consists of fixed-blocked (RECFM=FB) data sets, you must make a copy of the hlq.DATA data set (produced by the Customization Dialog) after the customization process is complete, and place the copy in the SYSEXEC concatenation.

If you do not specify a data set name, the customization process does not place any REXX EXECs in any data set.

WebSphere Application Server for z/OS product data sets

Specify the following WebSphere Application Server for z/OS libraries so they can be accessed by the customized jobstreams the dialog produces. These data sets must be cataloged. See “Product data sets” on page 37 for more information.

Run WebSphere Application Server from STEPLIB (Y/N)?

Specifies whether to load WebSphere Application Server for z/OS load modules from STEPLIB (″Y″) or from the link pack area and link list (″N″). See “Link pack area, link list, and STEPLIB” on page 43 for more information.

Note: Specify ″Y″ if you have another instance of WebSphere Application Server for z/OS (Version 4 or later) in the system link pack area or link list.

SBBOLPA

WebSphere Application Server for z/OS load module library. It has modules that should go into LPA or the location service daemon STEPLIB.

SBBOLOAD

WebSphere Application Server for z/OS 31-bit load module library. It has members that should go into the link list or LPA, or into STEPLIB.

SBBGLOAD

WebSphere Application Server for z/OS 64-bit load module library. It has members that should go into the link list or LPA, or into STEPLIB.

SBBOLD2

WebSphere Application Server for z/OS load module library that you installed through SMP/E. It has members that should go into the link list, or into STEPLIB. **DO NOT** place them in LPA.

SBBOEXEC

WebSphere Application Server for z/OS CLIST library.

SBBOMSG

SBBOMSG WebSphere Application Server for z/OS message skeletons for language translation.

Location of file system resident components

WebSphere Application Server product directory

The name of the directory where WebSphere Application Server for z/OS files reside after installation. See “Product file system” on page 39 for more information.

WebSphere configuration file system

This section defines the WebSphere configuration file system that you will use for the stand-alone application server. See “Configuration file system” on page 59 for more information.

Mount point

Read/write file system directory mount point where application data and environment files are written. The customization process creates this mount point if it does not already exist.

Name File System data set you will create and mount at the above mount point.

Rule: You can specify up to 44 characters for the data set name.

Volume, or ’*’ for SMS

Specify either the DASD volume serial number to contain the above data set or ″*″ to let SMS select a volume. Using ″*″ requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle data set allocation automatically, list the volume explicitly.

Primary allocation in cylinders

Initial size allocation in cylinders for the above data set.

Recommendation: The minimum suggested size is 420 cylinders (3390).

Secondary allocation in cylinders

Size of each secondary extent in cylinders.

Recommendation: The minimum suggested size is 100 cylinders.

File system type (HFS or ZFS)

This is the type of file system that will be used when creating the WebSphere for z/OS configuration file system. The default is HFS.

WebSphere error log stream

This section defines the System Logger log stream that you will use for WebSphere Application Server for z/OS error logging if TRACEBUFFLOC=BUFFER. Having the error log in the log stream is optional. By default, the error log is written to JES SYSOUT.

Name Name of the WebSphere error log stream you will create.

Rules:

- The name must be 26 or fewer characters.
- Do NOT put quotes around the name.

RRS setup (if not done previously)

Note: RRS setup was removed from the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.

If you do not have the RRS log streams set up, the Customization Dialog will create the jobs you can use to set up the log streams. See “Preparing Resource Recovery Services” on page 46 for more information.

Group name
Specify the XCF group name.

Recommendation: Use your sysplex name.

Data class
Specify an existing DFSMS Data Class for the log stream data set allocation. Applicable only if you use SMS.

Storage class
An existing DFSMS storage class for allocation of the DASD staging data set for this log stream. Applicable only if you use SMS.

HLQ for data sets
The high-level qualifier for your log stream data set name and staging data set name.

Is log stream CF resident (Y|N)
If the log stream is to be created on a coupling facility, specify ″Y″. If on DASD, specify ″N″.

Create RRS PROC (Y|N)
If you specify ″Y″, the dialog copies the ATRRRS cataloged procedure into SYS1.PROCLIB so that RRS can be started.

If you already have RRS set up, specify ″N″.

Component Trace

Note: Except for specifying the Trace Parmlib member suffix, component-trace setup was removed from the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.

WebSphere Application Server for z/OS uses component trace (CTRACE) to capture and to display trace data in trace data sets. WebSphere Application Server for z/OS identifies itself to CTRACE with the short cell name.

Procedure name
This is the CTRACE external writer start procedure to be created. It is identified in the WebSphere Application Server for z/OS CTRACE member (CTIBBOxx) in PARMLIB.

Rule: The name must be one to seven characters.

User ID
RACF user ID to be created and associated with the CTRACE external writer start procedure.

Group The name of an existing RACF group you will associate with this user.

Recommendation: Use the ″default started task″ RACF group for your target z/OS system.

Name Specify a fully qualified data set name, such as WAS390.CTRACE1, for the data set to be created. The default includes the system name of the system on which the Customization Dialog is running.

Rules:

- You can specify up to 44 characters for the data set name.
- Do not use quotes.

Volume, or ″*″ for SMS
Specify either the DASD volume serial number containing the above data set or ″*″ to let DFSMShsm™ select a volume. Using ″*″ requires SMS. Using ″*″ requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle data set allocation automatically, list the volume explicitly.

Primary space in cylinders
The primary space for the trace data set.

Secondary space in cylinders
The secondary space for the trace data set.

Trace Parmlib member suffix
Value that is appended to CTIBBO to form the member name for the Trace Parmlib member.

Ctrace Writer Definitions

WebSphere application Server for z/OS uses component trace (CTRACE) to capture and to display trace data in trace data sets. WebSphere Application Server for z/OS identifies itself to CTRACE with the *cell short name.*

Trace Parmlib member suffix
Value that is appended to CTIBBO to form the member name for the Trace parmlib member.

Server customization

WebSphere Application Server home directory
Directory in which the application server configuration resides. See “Configuration file system” on page 59 for more information.

Note: You can alter only the portion after the configuration file system mount point.

Cell name (short)
Name that identifies the cell to z/OS facilities such as SAF.

Rules:

- Name must be eight or fewer characters and all uppercase.
- Name must be unique among all other cells in the sysplex.

Cell name (long)
Primary external identification of this WebSphere Application Server for z/OS cell. This name identifies the cell as displayed through the administrative console.

Note: If you intend to ever add this stand-alone server node to a Network Deployment cell, ensure that the stand-alone server cell name is different from the Network Deployment cell name.

Rules:

- Name must be 50 or fewer characters and can be of mixed case.
- Name must be unique among all other cells in the sysplex.

Node name (short)
Name that identifies the node to z/OS facilities such as SAF.

Rules:

- Name must be eight or fewer characters and all uppercase.
- Name must be unique within the cell.

Node name (long)

Primary external identification of this WebSphere Application Server for z/OS node. This name identifies the node as displayed through the administrative console.

Note: If you intend to ever add this stand-alone server node to a deployment manager cell, ensure that the stand-alone server node name is not the same as that of any existing node in the Network Deployment cell.

Rules:

- Name must be 50 or fewer characters and can be of mixed case.
- Name must be unique within the cell.
- The application server must be defined on its own node; no other server can exist on the same node as the application server.

Server name (short)

This value identifies the server to z/OS facilities such as SAF.

Note: The server short name is also used as the server JOBNAME.

Rule: Name must usually contain seven or fewer all-uppercase characters. If you want to instead specify eight characters, you must follow the steps outlined in Converting a 7-character server short name to 8 characters.

Server name (long)

Name of the application server and the primary external identification of this WebSphere Application Server for z/OS server. This name identifies the server as displayed through the administrative console.

Rule: Name must be 50 or fewer characters and can be of mixed case.

Cluster transition name

WLM APPLENV (WLM application environment) name for this server.

Note: If this server is converted into a clustered server, this name becomes the cluster short name. The cluster short name is the WLM APPLENV name for all servers that are part of the same cluster. See “Cataloged procedures” on page 72 for more information.

Rule: Name must be eight or fewer characters and all uppercase.

Admin asynch operations procedure name

Specifies the JCL procedure name of a started task that is launched by way of the START command by node agents or application servers to perform certain asynchronous administrative operations (such as node synchronization) and add and remove a node. See “Cataloged procedures” on page 72 for more information.

Asychronous administration userid

This user ID is used to run asynchronous administration operations procedure. It must be a member of the WebSphere Application Server configuration group.

Install samples? (Y/N)

Specify whether you do (″Y″) or do not (″N″) want to install the WebSphere Application Server for z/OS sample applications (the Samples Gallery).

Note: These applications are not supported in a Network Deployment cell.

Server address space information

This section defines the cataloged procedures and associated user IDs for the server regions. See “Cataloged procedures” on page 72 for more information.

Rule: In the following, unless specified otherwise, names must be eight or fewer characters.

Note: If you want to specify jobnames with eight instead of seven characters, you must follow the steps outlined in Converting a 7-character server short name to 8 characters.

Controller Information

Jobname

The jobname, specified in the MVS START command JOBNAME parameter, associated with the application server controller. This is the same as the server short name and it cannot be changed through the Customization Dialog.

Procedure name

Name of member in your procedure library to start the application server controller.

Rule: Name must be seven or fewer characters.

User ID

The user ID associated with the application server controller.

Note: If you are using a non-IBM security system, the user ID might have to match the procedure name. Please refer to your security system's documentation.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Servant Information

Jobname

The jobname used by WLM to start the application server servant. This is set to the server short name, followed by the letter ″S″, and it cannot be changed through the Customization Dialog.

Procedure name

Name of member in your procedure library to start the application server servant.

Rule: Name must be seven or fewer characters.

User ID

The user ID associated with the application server servant.

Note: If you are using a non-IBM security system, the user ID might have to match the procedure name. Please refer to your security system's documentation.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers between 1 and 2,147,483,647 within the system.

Control Region Adjunct

Jobname

The jobname used by WLM to start the application server control region adjunct. This is set to the server short name followed by the letter ″A″, and it cannot be changed through the Customization Dialog.

Procedure name

Name of the member in your procedure library that starts the control region adjunct.

Rule: Name must be seven or fewer characters.

User ID

The user ID associated with the control region adjunct.

Note: If you use a non-IBM security system, the user ID might have to match the procedure name. Please refer to your security system's documentation for more information.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Server TCP/IP information

Note: Do not choose port values already in use.

Node host name

IP name or address of the system on which the server is configured. This value is used by other WebSphere Application Server for z/OS functions to connect to this server.

Note: The node host name must always resolve to an IP stack on the system where the application server runs. The node host name cannot be a DVIPA or a DNS name that, in any other way, causes the direction of requests to more than one system.

SOAP JMX Connector port
Port number for the JMX HTTP connection to this server based on the SOAP protocol. JMX is used for remote administrative functions, such as invoking scripts through wsadmin.sh.

Rule: Value cannot be 0.

ORB Listener IP name
IP address on which the server's ORB listens for incoming IIOP requests. The default is ″*″, which instructs the ORB to listen on all available IP addresses.

ORB port
Port for IIOP requests that acts as the bootstrap port for this server and also as the port through which the ORB accepts IIOP requests.

Rule: Value cannot be 0.

ORB SSL port
Port for secure IIOP requests. The default is ″0″, which allows the system to choose this port.

HTTP transport IP name
IP address on which the server's Web container should listen for incoming HTTP requests. The default is ″*″, which instructs the Web container to listen on all available IP addresses.

Note: The ″transport host name″ becomes the ″hostname″ in the virtualhosts.xml file, which makes setting a specific IP address here less than ideal because, if you do so, you are restricting yourself to that IP address until you go into the administrative console and add another virtual host.

Administrative console port
Port for HTTP requests to the administrative console.

Administrative console secure port
Port for secure HTTP requests to the administrative console.

HTTP transport port
Port for HTTP requests.

Rule: Value cannot be 0.

HTTPS Transport port
Port for secure HTTP requests.

Rule: Value cannot be 0.

High Availability Manager Communication Port
Port on which the High Availability Manager listens.

Rule: Value cannot be 0.

Service Integration port
Port for service-integration requests.

Rule: Value cannot be 0.

Service Integration Secure port
Port for secure service-integration requests.

Rule: Value cannot be 0.

Service Integration MQ Interoperability port
Port for service-integration MQ interoperability requests.

Rule: Value cannot be 0.

Service Integration MQ Interoperability Secure port
Port for secure service-integration MQ interoperability requests.

Rule: Value cannot be 0.

Session Initiation Protocol (SIP) port

Session Initiation Protocol Secure port

Location service daemon

The location service daemon is the initial point of client contact in WebSphere Application Server for z/OS. The server contains the CORBA-based location service agent, which places sessions in a cell. All RMI/IIOP IORs (for example, for enterprise beans) establish connections to the location service daemon first, then forward them to the target application server.

Daemon home directory
Directory in which the location service daemon resides. This is set to the configuration file system mount point/Daemon and cannot be changed.

Daemon jobname
Specifies the jobname of the location service daemon, specified in the JOBNAME parameter of the MVS start command used to start the location service daemon.

Caution: When configuring a new cell, be sure to choose a new daemon jobname value.

Note: A server automatically starts the location service daemon if it is not already running.

Procedure name
Name of the member in your procedure library to start the location service daemon.

Rule: Name must be seven or fewer characters.

User ID
The user ID associated with the location service daemon.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

IP Name
The fully qualified IP name, registered with the Domain Name Server (DNS), that the location service daemon uses. The default is your node host name.

Note:

- In a sysplex, you should consider using a virtual IP address (VIPA) for the location service daemon IP name.
- Select the IP name for the location service daemon carefully. Once you have chosen a name, it is difficult to change, even in the middle of customization. This name must not be a numeric, such as, 3.7.2543.

Daemon Listen IP
The default value is *.

Rule: The default is * or a numeric IP address.

Port The port number on which the location service daemon listens.

Note: Select the port number for the location service daemon carefully. You can choose any value you want, but, once chosen, it is difficult to change, even in the middle of customization.

SSL Port
The port number on which the location service daemon listens for SSL connections.

Register daemon with WLM DNS
If you use the WLM DNS (connection optimization), you must select ″Y″ to register your location service daemon with it. Otherwise, select ″N.″

Note: Only one location service daemon per LPAR can register its domain name with WLM DNS. If you have multiple cells in the same LPAR and register one location service daemon and then a second, the second will fail to start.

SSL customization

If you plan to enable Administrative Security at some point, as is recommended, fill in the following SSL values:

Certificate authority keylabel
The name of the key label that identifies the certificate authority (CA) to be used in generating server certificates.

Generate Certificate Authority (CA) certificate
Select ″Y″ to generate a new CA certificate. Select ″N″ to have an existing CA certificate generate server certificates.

Expiration date for CA authority
The expiration date used for any X509 Certificate Authority certificates, as well as the expiration date for the personal certificates generated for WebSphere Application Server for z/OS servers. You must specify this even if you selected ″N″ for ″Generate Certificate Authority (CA) certificate.″

Default RACF keyring name
The default name given to the RACF keyring used by WebSphere Application Server for z/OS. The keyring names created for repertoires are all the same within a cell.

Enable SSL on location service daemon
Select ″Y″ if you want to support secure communications using Inter-ORB Request Protocol (IIOP) to the location service daemon using SSL. If you specify ″Y″, a RACF keyring will be generated for the location service daemon to use.

Security Customization

During the initial setup, you can choose one of the following three options for administrative security.

Option 1 - z/OS-managed security
Use the z/OS system's SAF-compliant security database to define WebSphere Application Server users. The EJBROLE profile will be used to control role-based access to applications. An administrator user ID and an unauthenticated user ID will be created and defined in the security database. Select this option if the WebSphere Application Server environment will run entirely on z/OS with a shared SAF-compliant (Local OS) user registry, or if you plan to implement a non-Local OS user registry (such as LDAP) with mapping to SAF user IDs.

Option 2 - WebSphere-managed security
Use a simple file-based registry to define WebSphere Application Server users. An administrator user ID will be created and defined in the file-based registry.

Option 3 - No security
Do not enable administrative security. This option is not recommended.

Your WebSphere Application Server environment will not be secured until you configure and enable security manually.

Depending on the security option you choose, there may be additional customization dialog values you need to set.

Security customization - z/OS-managed security

For this security option, you must decide whether to set a security domain name, and choose an administrator user ID and an unauthenticated (guest) user ID.

Use security domain identifier in RACF profiles
Set this to Y if you wish to include a security domain name in certain SAF security checks (APPL, CBIND, EJBROLE). Enter a 1-8 security domain name.

Administrator user ID
For Administrator user ID enter a valid SAF user ID which will become the initial cell administrator. If this user ID already exists, it must have the WebSphere Application Server configuration group for this cell as its default Unix System Services group. Also specify a valid UID for this user ID.

Unauthenticated User ID
Enter a valid SAF user ID which will be associated with unauthenticated client requests. Also specify a valid UID for this user ID.

Security customization - WebSphere-managed security

For this security option, you must choose an administrator user ID and password.

Administrator user ID
Enter an alphanumeric user ID which you will use to log on to the administrative console and perform administrative tasks. This user ID and its password will initially be the only entry in the file-based user registry.

Administrator password
This password must not be blank.

Security Customization - no security

For this security option, there are no other choices to make. Your WebSphere Application Server environment will not be secured until you configure and enable security manually.

Web Server Configuration

If you want to create a web server definition at this point, fill in the following values.

Note:

Only one Web server can be defined on a stand alone application server.

Web Server Type
Valid values: IHS, HTTPSERVER_ZOS, APACHE, IPLANET, DOMINO, IIS.

Web Server Operating System
Windows®, Linux™, Solaris, AIX®, hpux, OS390, OS400

Web Server Name
The name used in defining the Web server in the admin console.

Web Server Host or IP Address
IP name or address of the z/OS on which the Web server is located.

Web Server Port
HTTP Port on which the Web server is listening.

Web Server Install Directory Path
(*varies by user configuration*)

Web Server Plugin Install Directory Path
(*varies by user configuration*)

Customization Dialog worksheet: Stand-alone application server cell

Date: ______________________

Purpose of this stand-alone application server:
____

System name: _________________________ Sysplex name: _________________________

Print out this worksheet and use it when collecting information about the WebSphere Application Server for z/OS Customization Dialog variables. The variables and defaults are provided, along with spaces for you to fill in your own value for each.

System Locations

Item	Default	Your value
PROCLIB	SYS1.PROCLIB	

WebSphere Application Server for z/OS product data sets

Run WebSphere Application Server from STEPLIB (Y/N)?	Y	
SBBOLPA	*was_hlq*.SBBOLPA	
SBBOLOAD	*was_hlq*.SBBOLOAD	
SBBGLOAD	*was_hlq*.SBBGLOAD	
SBBOLD2	*was_hlq*.SBBOLD2	
SBBOEXEC	*was_hlq*.SBBOEXEC	
SBBOMSG	*was_hlq*.SBBOMSG	

Locations of file system resident components

WebSphere Application Server product directory	/usr/lpp/zWebSphere/V6R1	

WebSphere configuration file system

Mount point	/WebSphere/V6R1	
Name	OMVS.WAS.CONFIG.HFS	
Volume, or '*' for SMS	*	
Primary allocation in cylinders	420	
Secondary allocation in cylinders	100	
File system type (HFS or ZFS)	HFS	

WebSphere error log stream

Name	WAS.ERROR.LOG	

Component Trace

Trace Parmlib member suffix	60	

Server customization

WebSphere Application Server home directory	*configmountpoint*/AppServer	
Cell name (short)	*systemname*	

Cell name (long)	*systemname*	
Node name (short)	*systemname*	
Node name (long)	*systemname*	
Server name (short)	BBOS001	
Server name (long)	server1	
Cluster transition name	BBOC001	
Admin asynch operations procedure name	BBOW6SH	
Asynchronous administration user ID	WSADMSH	
Asynchronous user ID value	2504	
Install samples? (Y/N)	Y	

Server address space information

Controller Information		
Jobname	*servershortname*	(cannot change)
Procedure name	BBO6ACR	
User ID	ASCR1	
UID	2431	
Servant Information		
Jobname	*servershortname*S	(cannot change)
Procedure name	BBO6ASR	
User ID	ASSR1	
UID	2432	
Control Region Adjunct		
Jobname	*servershortname*A	(cannot change)
Procedure name	BBO6CRA	
User ID	ASCRA1	
UID	2433	

Server TCP/IP information

Node host name	(blank)	
SOAP JMX Connector port	8880	
ORB Listener IP name	*	
ORB port	2809	
ORB SSL port	0	
HTTP transport IP name	*	
Administrative console port	9060	
Administrative secure console port	9043	
HTTP port	9080	
HTTP SSL port	9443	

High Availability Manager Communication Port	9353	
Service Integration port	7276	
Service Integration Secure port	7286	
Service Integration MQ Interoperability port	5558	
Service Integration MQ Interoperability Secure port	5578	
Session Initiation Protocol (SIP) Port	5060	
Session Initiation Secure Protocol (SIP) Port	5061	

Location service daemon

Daemon home directory	*configmountpoint*/Daemon	(cannot change)
Daemon jobname	BBODMNB	
Procedure name	BBO6DMN	
User ID	WSDMNCR1	
UID	2411	
IP name	*nodehostname*	
Port	5655	
SSL Port	5656	
Register daemon with WLM DNS	N	

Note: IP name must not be a numeric, such as, 3.7.25.43.

SSL Customization

Certificate Authority Keylabel	WebSphereCA	
Generate Certificate Authority (CA) certificate	Y	
Expiration date for CA authority	2010/12/31	
Default RACF keyring name	WASkeyring.<cellname>	
Enable SSL on location service daemon	Y	

Security configuration

Initial security: (1 = z/OS-managed security, 2 = WebSphere-managed security, 3 = No security)	1	

z/OS-managed security

Use security domain identifier in RACF profiles:	N	
Security domain identifier.		

WebSphere Application Server Administrator Information		
User ID	WSADMIN	
UID	2403	
WebSphere Application Server Unauthenticated User Information		
User ID	WSGUEST	
UID	2402	

WebSphere-managed security

Administrator user ID	WSADMIN	
Password:		

Web Server Configuration

Web Server Type	HTTPSERVER_ZOS	
Web Server Operating System	os390	
Web Server Name	webserver1	
Web Server Host or IP Address	(*nodehostname*)	
Web Server Port	80	
Web Server Install Directory Path	(*varies by type*)	
Web Server Plugin Install Directory Path	(*varies by type*)	

Name of stand-alone application server configuration variable data set:

____

Planning for a Network Deployment cell

This article covers the requirements for a Network Deployment cell.

Decide on the security standards that you plan to use for this cell and complete the “Customization Dialog worksheet: Common MVS Groups and Users” on page 81.

A Network Deployment cell is a full-function WebSphere Application Server for z/OS configuration on which you can deploy and run applications. A Network Deployment cell includes the following:

- A cell configuration.
- A deployment manager that runs the administrative console application.
- One or more application server nodes (one is recommended) on each z/OS target system hosting portions of the cell. Each node consists of a node agent and some number of application servers.
- A single location service daemon on each z/OS system.

This part of the configuration process creates the initial cell configuration, the deployment manager, and a location service daemon for the z/OS system on which the deployment manager runs. Once the Network Deployment cell is created, add application server nodes by creating and federating new managed nodes, or by federating stand-alone application server nodes into the Network Deployment cell.

When configuring your deployment manager node, keep the following in mind:

- When allocating target data sets for this option, it is possible, though not recommended, to use the same target data sets that you used for the stand-alone application server node. The jobnames for each configuration are very close to one another; and if you use the same target data sets, you might find it difficult to keep the two sets of jobs separate. Therefore, it is better to create a new set of target data sets and keep the two sets of jobs separate from one another.
- If possible, set up your file system such that the root file system is shared among all processors and the deployment manager's configuration is in a configuration HFS on a system-generic mount point.

  Note: This configuration scenario is the best for certain tasks, such as starting the deployment manager on another system, that you might want to perform in the future.

1. Print a copy of "Customization Dialog worksheet: Network Deployment cell" on page 108.
2. Fill out the worksheet as described in "Customization Dialog variables: Network Deployment cell."
3. Save the worksheet for use during Network Deployment cell customization.

Customization Dialog variables: Network Deployment cell

This article lists definitions for the terms you will come across in the WebSphere Application Server for z/OS Customization Dialog.

The WebSphere Application Server for z/OS runtime requires four servers in a Network Deployment cell: application server, deployment manager, node agent, and location service daemon. The panels corresponding to the following tables set up the names, network configuration, start procedures, and user IDs for a Network Deployment cell.

System Locations

This section identifies the target z/OS system on which you will configure the deployment manager for the Network Deployment cell, along with system data set names.

System name

The system name for the target z/OS system on which you will configure WebSphere Application Server for z/OS.

Sysplex name

The sysplex name for the target z/OS system on which you will configure WebSphere Application Server for z/OS.

Tip: If you are not sure what the system name (&SYSNAME) and sysplex name (&SYSPLEX) are, use the console command `D SYMBOLS` on the target z/OS system to display them.

For the following, specify the fully qualified data set names without quotes.

Rule: You can specify up to 44 characters for the data set names.

PROCLIB

An existing procedure library where the WebSphere Application Server for z/OS cataloged procedures are added.

WebSphere Application Server for z/OS product data sets

Specify the following WebSphere Application Server for z/OS libraries so they can be accessed by the customized jobstreams the dialog produces. These data sets must be cataloged. See "Product data sets" on page 37 for more information.

Run WebSphere Application Server from STEPLIB (Y/N)?

Specifies whether to load WebSphere Application Server for z/OS load modules from STEPLIB (″Y″) or from the link pack area and link list (″N″). See "Link pack area, link list, and STEPLIB" on page 43 for more information.

Note: Specify ″Y″ if you have another instance of WebSphere Application Server for z/OS (Version 4 or later) in the system link pack area or link list.

SBBOLPA
WebSphere Application Server for z/OS load module library. It has modules that should go into LPA or the location service daemon STEPLIB.

SBBOLOAD
WebSphere Application Server for z/OS 32-bit load module library. It has members that should go into the link list or LPA, or into STEPLIB.

SBBGLOAD
WebSphere Application Server for z/OS 64-bit load module library. It has members that should go into the link list or LPA, or into STEPLIB.

SBBOLD2
WebSphere Application Server for z/OS load module library that you installed through SMP/E. It has members that should go into the link list, or into STEPLIB. **DO NOT** place them in LPA.

SBBOEXEC
WebSphere Application Server for z/OS CLIST library.

SBBOMSG
SBBOMSG WebSphere Application Server for z/OS message skeletons for language translation.

Locations of file system resident components

WebSphere Application Server product directory
The name of the directory where WebSphere Application Server for z/OS files reside after installation. See “Product file system” on page 39 for more information.

WebSphere configuration file system

This section defines the WebSphere configuration file system that you will use for the deployment manager. See “Configuration file system” on page 59 for more information.

Mount point
Read/write file system directory mount point where application data and environment files are written. The customization process creates this mount point if it does not already exist.

Name File system data set you will create and mount at the above mount point.

Rule: You can specify up to 44 characters for the data set name.

Volume, or ’*’ for SMS
Specify either the DASD volume serial number to contain the above data set or ″*″ to let SMS select a volume. Using ″*″ requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle data set allocation automatically, list the volume explicitly.

Primary allocation in cylinders
Initial size allocation in cylinders for the above data set.

Recommendation: The minimum suggested size is 250 cylinders (3390).

Secondary allocation in cylinders
Size of each secondary extent in cylinders.

Recommendation: The minimum suggested size is 100 cylinders.

File system type (HFS or ZFS)
This is the type of file system that will be used when creating the WebSphere for z/OS configuration file system. The default is HFS.

WebSphere error log stream

This section defines the System Logger log stream that you will use for WebSphere Application Server for z/OS error logging if TRACEBUFFLOC=BUFFER. Having the error log in the log stream is optional. By default, the error log is written to JES SYSOUT.

Name Name of the WebSphere error log stream you will create.

Rules:

- The name must be 26 or fewer characters.
- Do NOT put quotes around the name.

Ctrace Writer Definitions

WebSphere application Server for z/OS uses component trace (CTRACE) to capture and to display trace data in trace data sets. WebSphere Application Server for z/OS identifies itself to CTRACE with the *cell short name.*

Trace Parmlib member suffix
Value that is appended to CTIBBO to form the member name for the Trace parmlib member.

Server customization

During this customization task, you will create a cell configuration, a deployment manager node and deployment manager, and a location service daemon. The panels corresponding to the following tables set up the names, network configuration, start procedures, and user IDs for a deployment manager server.

Rule: In the following customization, names must be eight or fewer characters unless otherwise specified.

WebSphere Application Server home directory
Directory in which the deployment manager resides. See "Configuration file system" on page 59 for more information.

Note: You can alter only the portion after the configuration file system mount point.

Cell name (short)
Name that identifies the cell to z/OS facilities such as SAF.

Rules:

- Name must be eight or fewer characters and all uppercase.
- Name must be unique among all other cells in the sysplex.

Cell name (long)
Primary external identification of this WebSphere Application Server for z/OS cell. This name identifies the cell as displayed through the administrative console.

Rules:

- Name must be 50 or fewer characters and can be of mixed case.
- Name must be unique among all other cells in the sysplex.

Node name (short)
Name that identifies the deployment manager node to z/OS facilities such as SAF.

Rules:

- Name must be eight or fewer characters and all uppercase.
- Name must be unique within the cell.

Node name (long)
Primary external identification of this WebSphere Application Server for z/OS deployment manager node. This name identifies the node as displayed through the administrative console.

Rules:

- Name must be 50 or fewer characters and can be of mixed case.
- Name must be unique within the cell.

Server name (short)
This value identifies the deployment manager server to z/OS facilities such as SAF.

Note: The server short name is used as the server JOBNAME.

Rule: Name must usually contain seven or fewer all-uppercase characters. To change the jobname later to an eight-character value, you must follow the steps outlined in Converting a 7-character server short name to 8 characters.

Server name (long)

Name of the deployment manager application server and the primary external identification of this WebSphere Application Server for z/OS. This name identifies the server as displayed through the administrative console. The server name has a fixed name (long) of ″dmgr″.

Cluster transition name

WLM APPLENV name for this server.

Note: The deployment manager is not clusterable, so this value never actually becomes the cluster short name of this server's cluster. However, like an application server, the deployment manager still needs an APPLENV, so the cluster transition name is used for this purpose.

Rule: Name must be eight or fewer characters and all uppercase.

Deployment manager definitions

Rule: In the following definitions, names must be eight or fewer characters unless otherwise specified.

Controller Information

Jobname

The jobname, specified in the MVS START command JOBNAME parameter, associated with the deployment manager controller. This is the same as the server short name and it cannot be changed through the Customization Dialog.

Procedure name

Name of member in your procedure library to start the deployment manager controller.

Rule: Name must be seven or fewer characters.

User ID

The user ID associated with the deployment manager controller.

Note: If you are using a non-IBM security system, the user ID might have to match the procedure name. Please refer to your security system's documentation.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Servant Information

Jobname

The jobname used by WLM to start the deployment manager servant. This is set to the server short name followed by the letter ″S″, and it cannot be changed through the Customization Dialog.

Procedure name

Name of member in your procedure library to start the deployment manager servant.

Rule: Name must be seven or fewer characters.

User ID

The user ID associated with the deployment manager servant.

Note: If you are using a non-IBM security system, the user ID might have to match the procedure name. Please refer to your security system's documentation.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Deployment manager TCP/IP information

Note: Do not choose port values already in use.

Node host name

IP name or address of the system on which the server is configured. This value is used by other WebSphere Application Server for z/OS functions to connect to this server.

Note: The node host name must always resolve to an IP stack on the system where the application server runs. (You can either have multiple IP stacks on a given MVS image and have the deployment manager and stand-alone application server tied to separate host names, or you can associate them with the same node host name.) The node host name cannot be a DVIPA or a DNS name that, in any other way, causes the direction of requests to more than one system.

SOAP JMX Connector port

Port number for the JMX HTTP connection to this server based on the SOAP protocol. JMX is used for remote administrative functions, such as invoking scripts through wsadmin.sh.

Rule: Value cannot be 0.

Cell Discovery Address port

Port number used by node agents to connect to this deployment manager server.

ORB Listener IP name

IP address on which the server's ORB listens for incoming IIOP requests. The default is ″*″, which instructs the ORB to listen on all available IP addresses.

ORB port

Port for IIOP requests that acts as the bootstrap port for this server and also as the port through which the ORB accepts IIOP requests.

Rule: Value cannot be 0.

ORB SSL port

Port for secure IIOP requests. The default is ″0″, which allows the system to choose this port.

HTTP transport IP name

IP address on which the server's Web container should listen for incoming HTTP requests. The default is ″*″, which instructs the Web container to listen on all available IP addresses.

Note: The ″transport host name″ becomes the ″hostname″ in the virtualhosts.xml file, which makes setting a specific IP address here less than ideal because, if you do so, you are restricting yourself to that IP address until you go into the administrative console and add another virtual host.

Administrative console port

Port for HTTP requests to the administrative console.

Administrative console secure port

Port for secure HTTP requests to the administrative console.

High Availability Manager communication port

Port on which the High Availability Manager listens.

Location service daemon

The location service daemon is the initial point of client contact in WebSphere Application Server for z/OS. The server contains the CORBA-based location service agent, which places sessions in a cell. All RMI/IIOP IORs (for example, for enterprise beans) establish connections to the location service daemon first, then forward them to the target application server.

Daemon home directory

Directory in which the location service daemon resides. This is set to the configuration HFS mount point/Daemon and cannot be changed.

Daemon jobname

Specifies the jobname of the location service daemon, specified in the JOBNAME parameter of the MVS start command used to start the location service daemon.

Note: The same daemon jobname is used on all z/OS systems hosting the cell. Therefore, this name is generic and has no system indicator. When the deployment manager's controller started, its location service daemon was started automatically using this particular daemon jobname value.

Caution: When configuring a second cell, ensure you change the daemon jobname from the default or value you used for the first cell.

Note: A server automatically starts the location service daemon if it is not already running.

Procedure name
Name of the member in your procedure library to start the location service daemon.

Rule: Name must be seven or fewer characters.

User ID
The user ID associated with the location service daemon.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

IP Name
The fully qualified IP name, registered with the Domain Name Server (DNS), that the location service daemon uses.

Adhere to the following regarding your location service daemon IP name:

- It must be unique in the sysplex.
- It must not have the same value as any one system's node host name. (You can use the host name of the LPAR.)
- It should be a name that can be used in conjunction with a routing service that distributes requests among nodes in the cell (that is, systems in the sysplex).
- It should be a virtual IP address (VIPA) if you are operating in a sysplex.

Note:

- IBM recommends you use z/OS Sysplex Distributor by way of a ″Dynamic Virtual IP address″ (DVIPA). See related sections in the information center for more information.
- Select the IP name for the location service daemon carefully. Once chosen it is difficult to change, even in the middle of customization. This name must not be a numeric, such as, 3.7.25.43.

Daemon Listen IP
The default value is *.

Rule: The default is * or a numeric IP address.

Port The port number on which the location service daemon listens.

Note: Select the port number for the location service daemon carefully. You can choose any port you want; but, once chosen, it is difficult to change, even in the middle of customization.

SSL port
The port number on which the location service daemon listens for SSL connections.

Register daemon with WLM DNS
If you use the WLM DNS (connection optimization), you must select ″Y″ to register your location service daemon with it. Otherwise, select ″N.″

Note: Only one location service daemon per LPAR can register its domain name with WLM DNS. If you have multiple cells in the same LPAR and register one location service daemon and then a second, the second will fail to start.

SSL customization

If you plan to enable Administrative Security at some point, as is recommended, fill in the following SSL values:

Certificate authority keylabel
The name of the key label that identifies the certificate authority (CA) to be used in generating server certificates.

Generate authority certificate
Select ″Y″ to generate a new CA certificate. Select ″N″ to have an existing CA certificate generate server certificates.

Expiration date for certificates
The expiration date used for any X509 Certificate Authority certificates, as well as the expiration date for the personal certificates generated for WebSphere Application Server for z/OS servers. You must specify this even if you selected ″N″ for ″Generate Certificate Authority (CA) certificate.″

Default RACF keyring name
The default name given to the RACF keyring used by WebSphere Application Server for z/OS. The keyring names created for repertoires are all the same within a cell.

Enable SSL on location service daemon
Select ″Y″ if you want to support secure communications using Inter-ORB Request Protocol (IIOP) to the location service daemon using SSL. If you specify ″Y″, a RACF keyring will be generated for the location service daemon to use.

Security Customization

During the initial setup, you can choose one of the following three options for administrative security.

Option 1 - Use z/OS-managed security
Use the z/OS system's SAF-compliant security database to define WebSphere Application Server users. The EJBROLE profile will be used to control role-based access to applications. An administrator user ID and an unauthenticated user ID will be created and defined in the security database. Select this option if the WebSphere Application Server environment will run entirely on z/OS with a shared SAF-compliant (Local OS) user registry, or if you plan to implement an non-Local OS user registry (such as LDAP) with mapping to SAF user IDs.

Option 2 - Use WebSphere-managed security
Use a simple file-based registry to define WebSphere Application Server users. An administrator user ID will be created and defined in the file-based registry.

Option 3 - No security
Do not enable administrative security (not recommended). Your WebSphere Application Server environment will not be secured until you configure and enable security manually.

Depending on the security option you choose, there may be additional customization dialog values you need to set.

Security customization - z/OS-managed security

For this security option, you must decide whether to set a security domain name, and choose an administrator user ID and an unauthenticated (guest) user ID.

Use security domain identifier in RACF profiles
Set this to Y if you wish to include a security domain name in certain SAF security checks (APPL, CBIND, EJBROLE). Enter a 1-8 character security domain name.

Administrator user ID
Administrator user ID Enter a valid SAF user ID which will become the initial cell administrator. If this user ID already exists, it must have the WebSphere Application Server configuration group for this cell as its default Unix System Services group. Also specify a valid UID for this user ID.

Unauthenticated User ID
Enter a valid SAF user ID which will be associated with unauthenticated client requests. Also specify a valid UID for this user ID.

Security customization - WebSphere-managed security

For this security option, you must choose an administrator user ID and password.

Administrator user ID
Enter an alphanumeric user ID which you will use to log on to the administrative console and perform administrative tasks. This user ID and its password will initially be the only entry in the file-based user registry.

Administrator password
This password must not be blank.

Security Customization - no security

For this security option, there are no other choices to make. Your WebSphere Application Server environment will not be secured until you configure and enable security manually.

Customization Dialog worksheet: Network Deployment cell

Date: ______________________

Purpose of this Network Deployment cell:
____

System name: _________________________ Sysplex name: _________________________

Print out this worksheet and use it when collecting information about the WebSphere Application Server for z/OS Customization Dialog variables. The variables and defaults are provided, along with spaces for you to fill in your own value for each.

System Locations

Item	Default	Your value
PROCLIB	SYS1.PROCLIB	

WebSphere Application Server for z/OS product data sets

Run WebSphere Application Server from STEPLIB (Y/N)?	Y	
SBBOLPA	*was_hlq*.SBBOLPA	
SBBOLOAD	*was_hlq*.SBBOLOAD	
SBBGLOAD	*was_hlq*.SBBGLOAD	
SBBOLD2	*was_hlq*.SBBOLD2	
SBBOEXEC	*was_hlq*.SBBOEXEC	
SBBOMSG	*was_hlq*.SBBOMSG	

Locations of Resident File System Components

WebSphere Application Server product directory	/usr/lpp/zWebSphere/V6R1	

WebSphere configuration file system

Mount point	/WebSphere/V6R1	
Name	OMVS.WAS.CONFIG.HFS	
Volume, or '*' for SMS	*	
Primary allocation in cylinders	250	
Secondary allocation in cylinders	100	
File System used (HFS or ZFS)	HFS	

WebSphere error log stream

Name	WAS.ERROR.LOG	

Component Trace

Trace Parmlib member suffix	60	

Server customization

WebSphere Application Server home directory	*configmountpoint*/ DeploymentManager	
Cell name (short)	*sysplexname*	
Cell name (long)	*sysplexname*Network	
Node name (short)	*sysplexname*	
Node name (long)	*sysplexname*Manager	
Server name (short)	BBODMGR	
Server name (long)	dmgr	(cannot change)
Cluster transition name	BBODMGR	

Deployment manager definitions

Controller Information		
Jobname	*deploymentmanagershortname*	(cannot change)
Procedure name	BBO6DCR	
User ID	DMCR1	
UID	2421	
Servant Information		
Jobname	*deploymentmanagershortname*S	(cannot change)
Procedure name	BBO6DSR	

User ID	DMSR1	
UID	2422	

Deployment manager TCP/IP information

Node host name	(blank)	
SOAP JMX connector port	8879	
Cell Discovery Address port	7277	
ORB Listener IP name	*	
ORB port	9809	
ORB SSL port	0	
HTTP transport IP name	*	
Administrative console port	9060	
Administrative secure console port	9043	
High availability manager communication port	9352	

Note: The administrative console port for the deployment manager needs to be a different port from that used for the administrative console port for the stand-alone application server. Otherwise, you need to recycle the deployment manager before federating the application server.

Location service daemon

Daemon home directory	*configmountpoint*/Daemon	(cannot change)
Daemon jobname	BBODMNC	
Procedure name	BBO6DMN	
User ID	WSDMNCR1	
UID	2411	
IP name	(blank)	
Port	5755	
SSL port	5756	
Register daemon with WLM DNS	N	

SSL Customization

Certificate Authority Keylabel	WebSphereCA
Generate Certificate Authority (CA) certificate	Y
Expiration date for CA authority	2010/12/31
Default RACF keyring name	WASkeyring.<cellname>
Enable SSL on location service daemon	Y

Security configuration

Initial security: (1 = z/OS-managed security, 2 = WebSphere-managed security, 3 = No security)	1	

z/OS-managed security

Use security domain identifier in RACF profiles:	N	
Security domain identifier.		
WebSphere Application Server Administrator Information		
User ID	WSADMIN	
UID	2403	
WebSphere Application Server Unauthenticated User Information		
User ID	WSGUEST	
UID	2402	

WebSphere-managed security

WSADMIN	Administrator user ID	
	Password:	

Name of Network Deployment cell configuration variable data set:
____

Planning for a Network Deployment cell with an application server

A Network Deployment cell is a full-function WebSphere Application Server for z/OS configuration on which you can deploy and run applications. A Network Deployment cell with application server includes the following:

- A cell configuration.
- A deployment manager that runs the administrative console application.
- A single location service daemon on each z/OS system.
- One application server consisting of a node agent and one application server.
- One or more application server nodes (one is recommended) on each z/OS target system hosting portions of the cell. Each node consists of a node agent and some number of application servers.

This part of the configuration process creates the initial cell configuration, the deployment manager, and a location service daemon for the z/OS system, plus a node agent with an application server. Once the Network Deployment cell is created, you can add additional application server nodes by creating and federating new managed nodes, or by federating stand-alone application server nodes into the Network Deployment cell.

When configuring your deployment manager node, set up your file system such that the root file system is shared among all processors and the deployment manager's configuration is in a configuration file system on a system-generic mount point.

Note: This configuration scenario is the best for certain tasks, such as starting the deployment manager on another system, that you might want to perform in the future.

Decide on the security standards that you plan to use for this cell and complete the "Customization Dialog worksheet: Common MVS Groups and Users" on page 81.

1. Print a copy of "Customization Dialog worksheet: Network Deployment cell with an Application Server" on page 237.
2. Fill out the worksheet as described in "Customization Dialog variables: Network Deployment cell with an Application Server" on page 229.
3. Save the worksheet for use during Network Deployment cell with an application server customization.

Planning for a new managed node in a Network Deployment cell

You need to have already configured a Network Deployment cell and deployment manager.

Create a new managed node in a Network Deployment cell in order to add application servers to the cell.

This part of the configuration process creates an application server node structure, a node agent (for node administration), and a location service daemon (if one does not already exist) for the chosen z/OS system. This can be the same z/OS system on which the deployment manager was configured or a different z/OS system in the same sysplex. Once the managed node is created and federated into the Network Deployment cell, add application servers using the administrative console or scripting. You can use the configuration file system and user IDs created for the managed server node for the application servers in the node as well.

1. Print a copy of "Customization Dialog worksheet: Managed node" on page 118.
2. Fill out the worksheet as described in "Customization Dialog variables: Managed node."
3. Save the worksheet for use during managed node customization.

Customization Dialog variables: Managed node

This article lists definitions for the terms you will come across in the WebSphere Application Server for z/OS Customization Dialog.

System locations

This section identifies the target z/OS system on which you will configure the new stand-alone application server, along with system data set names.

System name
: The system name for the target z/OS system on which you will configure WebSphere Application Server for z/OS.

Sysplex name
: The sysplex name for the target z/OS system on which you will configure WebSphere Application Server for z/OS.

  Tip: If you are not sure what the system name (&SYSNAME) and sysplex name (&SYSPLEX) are, use the console command `D SYMBOLS` on the target z/OS system to display them.

For the following, specify the fully qualified data set names without quotes.

Rule: You can specify up to 44 characters for the data set names.

PROCLIB
: An existing procedure library where the WebSphere Application Server for z/OS cataloged procedures are added.

WebSphere Application Server for z/OS product data sets

Specify the following WebSphere Application Server for z/OS libraries so they can be accessed by the customized jobstreams the dialog produces. These data sets must be cataloged. See "Product data sets" on page 37 for more information.

Run WebSphere Application Server from STEPLIB (Y/N)?
Specifies whether to load WebSphere Application Server for z/OS load modules from STEPLIB (″Y″) or from the link pack area and link list (″N″). See "Link pack area, link list, and STEPLIB" on page 43 for more information.

Note: Specify ″Y″ if you have another instance of WebSphere Application Server for z/OS (Version 4 or later) in the system link pack area or link list.

SBBOLPA
WebSphere Application Server for z/OS load module library. It has modules that should go into LPA or the location service daemon STEPLIB.

SBBOLOAD
WebSphere Application Server for z/OS 32-bit load module library. It has members that should go into the link list or LPA, or into STEPLIB.

SBBGLOAD
WebSphere Application Server for z/OS 64-bit load module library. It has members that should go into the link list or LPA, or into STEPLIB.

SBBOLD2
WebSphere Application Server for z/OS load module library that you installed through SMP/E. It has members that should go into the link list, or into STEPLIB. **DO NOT** place them in LPA.

SBBOEXEC
WebSphere Application Server for z/OS CLIST library.

SBBOMSG
SBBOMSG WebSphere Application Server for z/OS message skeletons for language translation.

Locations of file system resident components

WebSphere Application Server product directory
The name of the directory where WebSphere Application Server for z/OS files reside after installation. See "Product file system" on page 39 for more information.

WebSphere configuration file system

This section defines the WebSphere configuration file system that you will use for the new managed node. See "Configuration file system" on page 59 for more information.

Mount point
Read/write file system directory mount point where application data and environment files are written. The customization process creates this mount point if it does not already exist.

Name File system data set you will create and mount at the above mount point.

Rule: You can specify up to 44 characters for the data set name.

Volume, or '*' for SMS
Specify either the DASD volume serial number to contain the above data set or ″*″ to let SMS select a volume. Using ″*″ requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle data set allocation automatically, list the volume explicitly.

Primary allocation in cylinders
Initial size allocation in cylinders for the above data set.

Recommendation: The minimum suggested size is 250 cylinders (3390).

Secondary allocation in cylinders
Size of each secondary extent in cylinders.

Recommendation: The minimum suggested size is 100 cylinders.

File system type (HFS or ZFS)
This is the type of file system that will be used when creating the WebSphere for z/OS configuration file system. The default is HFS.

WebSphere error log stream

This section defines the System Logger log stream that you will use for WebSphere Application Server for z/OS error logging if TRACEBUFFLOC=BUFFER. Having the error log in the log stream is optional. By default, the error log is written to JES SYSOUT.

Name Name of the WebSphere error log stream you will create.

Rules:
- The name must be 26 or fewer characters.
- Do NOT put quotes around the name.

Ctrace Writer Definitions

WebSphere application Server for z/OS uses component trace (CTRACE) to capture and to display trace data in trace data sets. WebSphere Application Server for z/OS identifies itself to CTRACE with the *cell short name.*

Trace Parmlib member suffix
Value that is appended to CTIBBO to form the member name for the Trace parmlib member.

Server customization

During this customization task, you will create a (temporary) cell configuration, a node configuration, and a (temporary) location service daemon.

Note: The cell configuration and location service daemon are temporary because they are replaced shortly after creation when the new managed node is federated.

The panels corresponding to the following tables set up the names, network configuration, start procedures, and user IDs for the future node agent and application servers.

WebSphere Application Server home directory
Directory in which the node's home directories will reside. See "Configuration file system" on page 59 for more information.

Note: You can alter only the portion after the configuration file system mount point.

Node Host Name
IP address or host name of the system on which the node resides.

Cell name (short)
Name that identifies the cell to z/OS facilities such as SAF. This name is used only until the managed node is federated into the Network Deployment cell.

Rules:
- Name must be eight or fewer characters and all uppercase.
- Name must be unique among all other cells in the sysplex.

Cell name (long)
Primary external identification of this WebSphere Application Server for z/OS cell. This name identifies the cell as displayed through the administrative console. This name is used only until the managed node is federated into the Network Deployment cell.

Rules:
- Name must be 50 or fewer characters and can be of mixed case.
- Name must be unique among all other cells in the sysplex.

Node name (short)
Name that identifies the node to z/OS facilities such as SAF. This name is **NOT** temporary, and will become the short name of the managed node in the Network Deployment cell. Choose a name that is unique in the Network Deployment cell.

Rules:
- Name must be eight or fewer characters and all uppercase.

- Name must be unique within the cell.

Node name (long)
Primary external identification of this WebSphere Application Server for z/OS node. This name identifies the node as displayed through the administrative console. This name is **NOT** temporary, and will become the short name of the managed node in the Network Deployment cell. Choose a name that is unique in the Network Deployment cell.

Rules:
- Name must be 50 or fewer characters and can be of mixed case.
- Name must be unique within the cell.

Admin asynch operations procedure name
Specifies the JCL procedure name of a started task that is launched by way of the START command by node agents or application servers to perform certain asynchronous administrative operations (such as node synchronization) and add and remove a node. See "Cataloged procedures" on page 72 for more information.

Server address space information

This section defines the cataloged procedures and associated user IDs for the node agent created during federation. See "Cataloged procedures" on page 72 for more information.

Rule: In the following, unless specified otherwise, names must be eight or fewer characters.

Controller Information

Procedure name
Name of member in your procedure library to start the node agent. It can also be used for application controllers in the node.

Rule: Name must be seven or fewer characters.

User ID
The user ID associated with the node agent. It can also be used as for application server controllers in the node.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Servant Information

Procedure name
Name of member in your procedure library to start the application server servants in the node. This procedure is not used by the node agent.

Rule: Name must be seven or fewer characters.

User ID
The user ID associated with the application servants in the node. This user ID is not used by the node agent.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Control Region Adjunct

Procedure name
Name of member in your procedure library to start application server control region adjuncts in the node. This procedure is not used by the node agent.

Rule: Name must be seven or fewer characters.

User ID
The user ID associated with application server control region adjuncts in the node. This user ID is not used by the node agent.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Note: Additional SAF definitions may be required for you to start the server successfully.

Location service daemon

The location service daemon is the initial point of client contact in WebSphere Application Server for z/OS. The server contains the CORBA-based location service agent, which places sessions in a cell. All RMI/IIOP IORs (for example, for enterprise beans) establish connections to the location service daemon first, then forward them to the target application server.

The location service daemon here is temporary and exists only until the managed node is federated into the Network Deployment cell.

Daemon home directory
Directory in which the location service daemon resides. This is set to the configuration file system mount point/Daemon and cannot be changed.

Daemon jobname
Specifies the jobname of the location service daemon, specified in the JOBNAME parameter of the MVS start command used to start the location service daemon.

Caution: When configuring a new managed node, ensure that this name is different from the daemon jobname for the Network Deployment cell into which you will federate the managed node.

Note: A server automatically starts the location service daemon if it is not already running.

Procedure name
Name of the member in your procedure library to start the location service daemon.

Rule: Name must be seven or fewer characters.

User ID
The user ID associated with the location service daemon.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

SSL customization

If you plan to enable Administrative Security at some point, as is recommended, fill in the following SSL values:

Certificate Authority Keylabel
The name of the key label that identifies the certificate authority (CA) to be used in generating server certificates.

Expiration date for CA authority
The expiration date used for any X509 Certificate Authority certificates, as well as the expiration date for the personal certificates generated for WebSphere Application Server for z/OS servers. You must specify this even if you selected ″N″ for ″Generate Certificate Authority (CA) certificate.″

Default RACF keyring name
The default name given to the RACF keyring used by WebSphere Application Server for z/OS. The keyring names created for repertoires are all the same within a cell.

Federation variables

WebSphere Application Server home directory
The home directory that will be passed to the addnode.sh command for federation. This is the same as the node home directory and cannot be changed here.

Node host name
The TCP/IP node name of the deployment manager for the Network Deployment cell.

JMX SOAP port
The JMX SOAP port of the deployment manager for the Network Deployment cell.

Deployment manager security is enabled
Specify ″Y″ if Administrative Security is enabled on the Network Deployment cell and the deployment manager.

User ID
A user ID with full administrative privileges for the Network Deployment cell. This is the security domain administrator user ID and cannot be changed.

Password
Password for User ID.

Node group name
The node group into which the node will be placed. Specify DefaultNodeGroup if the node is in the same sysplex as the deployment manager. See Node group for more information.

Node agent server name (short)
Name of the node agent server. This is the server's jobname, as specified in the MVS START command JOBNAME parameter. This value identifies the server to z/OS facilities such as SAF.

Rule: Name must usually contain seven or fewer all-uppercase characters. To change the jobname later to an eight-character value, you must follow the steps outlined in Converting a 7-character server short name to 8 characters.

Node agent server name (long)
Name of the node agent and the primary external identification of the node agent server. This name identifies the server as displayed through the administrative console . The node agent server long name is set to the fixed value ″nodeagent″.

JMX SOAP connector port
Port number for the JMX HTTP connection to this server based on the SOAP protocol. JMX is used for remote administrative functions and is invoked through scripts such as wsadmin.sh.

Rule: Value cannot be 0.

Node Discovery port
Defines the TCP/IP port to which the node agents listens for discovery requests that originate from the deployment manager.

Node Multicast Discovery port
Defines the multicast port through which the node agent sends discovery requests to its managed servers.

The multicast IP address on which the discovery port is opened is defaulted by WebSphere Application Server for z/OS to 232.133.104.73. This default address can be changed using the administrative console. This is a CLASS D address. The valid IP range is from 224.0.0.0 to 239.255.255.255.

High availability manager communication port
Port on which the High Availability Manager listens.

ORB listener IP name
IP address on which the server's ORB listens for incoming IIOP requests. The default is ″*″, which instructs the ORB to listen on all available IP addresses.

ORB port
Port for IIOP requests that acts as the bootstrap port for the server and also as the port through which the ORB accepts IIOP requests.

Rule: Value cannot be 0.

ORB SSL port
Port for secure IIOP requests. The default is ″0″, which allows the system to choose this port.

Security Customization

During the initial setup, you can select one of the following three options for administrative security. Choose the same security option as the Network Deployment cell into which the Managed node will be federated.

Option 1 - Use z/OS-managed security
Use the z/OS system's SAF-compliant security database to define WebSphere Application Server users. The EJBROLE profile will be used to control role-based access to applications. An administrator user ID and an unauthenticated user ID will be created and defined in the security database. Select this option if the WebSphere Application Server environment will run entirely on z/OS with a shared SAF-compliant (Local OS) user registry, or if you plan to implement an non-Local OS user registry (such as LDAP) with mapping to SAF user IDs.

Option 2 - Use WebSphere-managed security
Use a simple file-based registry to define WebSphere Application Server users. An administrator user ID will be created and defined in the file-based registry.

Option 3 - No security
Do not enable administrative security (not recommended). Your WebSphere Application Server environment will not be secured until you configure and enable security manually.

For the Managed node customization, there are no additional security prompts.

Customization Dialog worksheet: Managed node

Date: ______________________

Purpose of this managed node:
____

System name: _________________________ Sysplex name: _________________________

Print out this worksheet and use it when collecting information about the WebSphere Application Server for z/OS Customization Dialog variables. The variables and defaults are provided, along with spaces for you to fill in your own value for each.

System Locations

Item	Default	Your value
PROCLIB	SYS1.PROCLIB	

WebSphere Application Server for z/OS product data sets

Run WebSphere Application Server from STEPLIB (Y/N)?	Y	
SBBOLPA	*was_hlq*.SBBOLPA	
SBBOLOAD	*was_hlq*.SBBOLOAD	
SBBGLOAD	*was_hlq*.SBBGLOAD	
SBBOLD2	*was_hlq*.SBBOLD2	
SBBOEXEC	*was_hlq*.SBBOEXEC	

SBBOMSG	*was_hlq*.SBBOMSG	

Locations of Resident File System Components

WebSphere Application Server product directory	/usr/lpp/zWebSphere/V6R1	

WebSphere configuration file system

Mount point	/WebSphere/V6R1	
Name	OMVS.WAS.CONFIG.HFS	
Volume, or '*' for SMS	*	
Primary allocation in cylinders	250	
Secondary allocation in cylinders	100	

WebSphere error log stream

Name	WAS.ERROR.LOG	

Component Trace

Trace Parmlib member suffix	60	

Server customization

WebSphere Application Server home directory	*configmountpoint/AppServer*	Mount point can not be changed
Node host name	(blank)	
Cell name (short)	*systemname*	
Cell name (long)	*systemname*	
Node name (short)	*systemname*	
Node name (long)	*systemname*	
Admin asynch operations procedure name	BBOW6SH	

Server address space information

Controller Information		
Procedure name	BBO6ACR	
User ID	ASCR1	
UID	2431	
Servant Information		
Procedure name	BBO6ASR	
User ID	ASSR1	
UID	2432	
Control Region Adjunct		

Procedure name	BBO6CRA	
User ID	ASCRA1	
UID	2433	

Location service daemon

Daemon home directory	*configmountpoint*/Daemon	(cannot change)
Daemon jobname	BBODMNB	
Procedure name	BBO6DMN	
User ID	WSDMNCR1	
UID	2411	

SSL Customization

Certificate authority keylabel	WebSphereCA
Expiration date for certification	2010/12/31
Default RACF keyring name	WASKeyring.<nodename>

Federation variables

WebSphere Application Server home directory	(as set above)	(cannot change here)
Deployment Manager Access		
Node host name	(blank)	
JMX SOAP port	8879	
Deployment manager security is enabled	N	
User ID	(administrator user ID)	(cannot change)
Password		
Node group name	DefaultNodeGroup	
Node Agent Definitions		
Server name (short)	BBON001	
Server name (long)	nodeagent	(cannot change)
JMX SOAP connector port	9360	
Node Discovery port	7272	
Node Multicast Discovery port	5000	
High availability manager communication port	9354	
ORB listener IP name	*	
ORB port	2809	
ORB SSL port	0	

Security configuration

Initial security: (1 = z/OS-managed security, 2 = WebSphere-managed security, 3 = No security)	1	

z/OS-managed security

Use security domain identifier in RACF profiles:	N	
Security domain identifier.		
WebSphere Application Server Administrator Information		
User ID	WSADMIN	
UID	2403	
WebSphere Application Server Unauthenticated User Information		
User ID	WSGUEST	
UID	2402	

WebSphere-managed security

Administrator user ID	WSADMIN	
Password:		

Name of managed node configuration variable data set:
____

Planning to federate a stand-alone server into a Network Deployment cell

You must have configured a Network Deployment cell (and deployment manager) and a stand-alone application server. The two need to have a common MVS group and user domain and reside within the same z/OS sysplex.

Federate an existing stand-alone application server node into a Network Deployment cell in order to add application servers to the cell. The Deployment Manager needs to be at an equal or higher service level than the node being federated.

The cell structure and location service daemon for the stand-alone application server are discarded. The stand-alone application server node and its application servers become a new node in the Network Deployment cell. The stand-alone application server's configuration file system and home directory stay in use, but are modified to reflect the new cell name. New symbolic links for use during server startup are added.

1. Print a copy of "Customization Dialog worksheet: Federating an application server node" on page 124.
2. Fill out the worksheet as described in "Customization Dialog variables: Federating an application server node" on page 122.
3. Save the worksheet for use during federated application server node customization.

Customization Dialog variables: Federating an application server node

This article lists definitions for the terms you will come across in the WebSphere Application Server for z/OS Customization Dialog.

Rules:

- In the following, unless specified otherwise, names must be eight or fewer characters.
- In some of the following, specifying ″Y″ (yes) tells the dialog to define the profile or enable an option. Specifying ″N″ (no) tells the dialog to not define the profile or enable the option.

WebSphere Application Server for z/OS home directory

WebSphere Application Server home directory

Directory in which the stand-alone application server configuration resides.

Note: Depending on the variables you loaded in the ″Load customization variables″ step, you might need to modify this value to reflect the correct directory.

Deployment Manager Access

Node host name

IP name or address of the system on which the deployment manager server is configured. This value, equivalent to ″cell host″ in addNode.sh, is used by other WebSphere Application Server for z/OS functions to connect to this server in order to federate the designated node into the deployment manager cell.

Note: The node host name must always resolve to an IP stack on the system where the deployment manager runs. The node host name cannot be a DVIPA or a DNS name that, in any other way, causes the direction of requests to more than one system.

JMX SOAP port

JMX (Java Management Extensions) SOAP (formerly known as Simple Object Access Protocol) connector port that the add node request uses to connect to the deployment manager. It provides the federation process with knowledge of which deployment manager is the target of the federation.

Deployment manager security is enabled

Specify ″Y″ if administrative security is enabled on the Network Deployment cell and the deployment manager.

User ID

A user ID with full administrative privileges for the Network Deployment cell. This is set to the security domain administrator ID and cannot be changed.

Include Apps

Specify ″Y″ if you want to include applications with your deployment manager node. Enabling this option instructs the addNode program to include applications from the node, as it would remove them prior to federation otherwise. If the application already exists in the cell, a warning is printed and the application is not installed into the cell.

Note: You must use this option to migrate all the applications to the new cell. Federating the node to a cell using the addNode command does not merge any cell-level configuration information, including that from virtualHost.

Launch the node agent after node federation

Specify ″N″ if you do not want the node agent to start automatically after federating a node.

Application server's ORB port

Port for IIOP requests that acts as the bootstrap port for this server and also as the port through which the ORB accepts IIOP requests. Port value cannot be 0.

Note: The add node operation creates the none agent administrative server with a default ORB port equivalent to the INS CosNaming default bootstrap port. Because this same port was previously used by the node's initial stand-alone server, the initial stand-alone server's ORB port must change to a new port value. The default value to which the application server's ORB port is set is 9810. If you configure multiple cells that intersect the same systems, use of the default value will cause a port conflict between these cells. This option helps you set the port number in case port 9810 was previously assigned.

Node group name
The node group into which the node will be placed. Specify DefaultNodeGroup if the node is in the same sysplex as the deployment manager. See Node group for more information.

Node Agent Definitions

As part of the federation process, a node agent is created. This special server administers the node and its servers on behalf of the deployment manager.

Server name (short)
Name of the node agent server. This is the server's jobname, as specified in the MVS START command JOBNAME parameter. This value identifies the server to certain z/OS facilities used by WebSphere Application Server for z/OS (SAF for example).

Rule: Name must usually contain seven or fewer all-uppercase characters. To change the jobname later to an eight-character value, you must follow the steps outlined in Converting a 7-character server short name to 8 characters.

Server name (long)
Name of the node agent and the primary external identification of the node agent server. This name identifies the server as displayed through the administrative console. The node agent server long name is set to the fixed value ″nodeagent″.

JMX SOAP connector port
Port number for the JMX HTTP connection to this server based on the SOAP protocol. JMX is used for remote administrative functions and is invoked through scripts such as wsadmin.sh.

Rule: Value cannot be 0.

Node Discovery port
Defines the TCP/IP port to which the node agents listens for discovery requests that originate from the deployment manager.

Node Multicast Discovery port
Defines the multicast port through which the node agent sends discovery requests to its managed servers.

The multicast IP address on which the discovery port is opened is defaulted to 232.133.104.73 by WebSphere Application Server for z/OS. This default address can be changed using the administrative console. This is a CLASS D address. The valid IP range is from 224.0.0.0 to 239.255.255.255.

High Availability Manager communication port
Port on which the High Availability Manager listens.

ORB listener host name
IP address on which the server's ORB listens for incoming IIOP requests. The default is ″*″, which instructs the ORB to listen on all available IP addresses.

ORB port
Port for IIOP requests that acts as the bootstrap port for this server and also as the port through which the ORB accepts IIOP requests.

Rule: Value cannot be 0.

ORB SSL port
Port for secure IIOP requests. The default is ″0″, which allows the system to choose this port.

Service integration bus

Do you wish to federate service integration busses that exist on this node? (Y/N)
Specify ″Y″ to federate service integration busses that exist on this node.

Customization Dialog worksheet: Federating an application server node

Date: ______________________

Purpose of this federated node:

____

Print out this worksheet and use it when collecting information about the WebSphere Application Server for z/OS Customization Dialog variables. The variables and defaults are provided, along with spaces for you to fill in your own value for each.

WebSphere Application Server for z/OS home directory

Item	Default	Your value
WebSphere Application Server home directory	/WebSphere/V6R1/AppServer	

Deployment Manager Access

Node host name	(blank)	
JMX SOAP port	8879	
Deployment manager security is enabled	N	
User ID	(administrator user ID)	(cannot change)
Include Apps	Y	
Launch the node agent after node federation (**Note:** This field was added to the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.)	Y	
Application server's ORB port	9810	
Node group name	DefaultNodeGroup	

Node Agent Definitions

Server name (short)	BBON001	
Server name (long)	nodeagent	(cannot change)
JMX SOAP Connector port	9360	
Node Discovery port	7272	
Node Multicast Discovery port	5000	
High availability manager communication port	9354	
ORB listener host name	*	
ORB port	2809	
ORB SSL port	0	

Service integration bus

Do you wish to federate service integration busses that exist on this node? (Y/N)	N	

Planning for recovery

This article helps you plan for any recovery measures you might need to take.

1. Decide whether or not to implement automatic restart. See “Automatic restart management” on page 126 for more information.
2. Review the recommendations for starting a deployment manager on a different MVS image. See “Starting a deployment manager on a different MVS image” for more information.

Starting a deployment manager on a different MVS image

This describes steps you must follow to start your deployment manager on an MVS image different from the one on which it was originally configured.

The ability to start your deployment manager on an MVS image different from the one on which it was originally configured is handy if your original system becomes unavailable, either through a planned outage or a system failure. This way, you can still start and stop applications, make configuration updates, utilize monitors that use the PMI interface, perform other control functions, and so on. Perform the following steps to start your deployment manager on a different MVS image and ensure that client requests will successfully find the deployment manager at its new location.

Note: This works only if the deployment manager on the original MVS image is down. WebSphere Application Server for z/OS allows only one copy of the deployment manager to run at one time for any given cell.

1. Ensure that the MVS image to which you are moving the deployment manager contains a node that is already part of the cell of the deployment manager you want to move.
2. Ensure that the location service daemon on the MVS image to which you are moving the deployment manager is up and running before you move the deployment manager.
3. Set, in the Customization Dialog, your host names and ports appropriately:
   - Ensure that the host names and ports for the deployment manager are not specific to a particular system.
   - Ensure that you use a DVIPA generic host name, rather than a system-specific host name, for the ″Node host name″ value and an asterisk (″*″) for both the ″ORB Listener host name″ and ″HTTP Transport host name.″

     Note: After initial configuration, you can change the ″Node host name,″ ″ORB Listener host name″ and ″HTTP Transport host name″ values in the administrative console if necessary.
   - Consider configuring a secondary DVIPA in case the system with the primary VIPA is down.
4. Ensure that Sysplex Distributor is enabled so that, regardless of where the DVIPA has moved, it automatically routes any inbound traffic to the deployment manager.
5. Ensure that access to the PROCLIB is the same for both the original MVS image and the MVS image to which you want to move the deployment manager.
6. Start the deployment manager on the new system. There are three ways to accomplish this, depending on the configuration of your HFSs. Follow the scenario that matches your configuration.

   Scenario 1: Root HFS is shared among all processors, deployment manager’s configuration is in a configuration HFS on a system-generic mount point.

a. Issue the start command for the deployment manager on the system on which you want it to reside:

```
S <controller_procname>,JOBNAME=<jobname>,
ENV=<cell_shortname>.<node_shortname>.<server_shortname>
```

Scenario 2: Root HFS is shared among all processors, deployment manager's configuration HFS is mounted under a system-specific directory.

Note: This is an undesirable scenario that you should try to avoid from the start of your system configuration. If you find yourself with this setup, however, follow these steps for the workaround.

a. Create a symbolic link at the equivalent system-specific location on the target MVS image. The contents of the symbolic link should point back to the actual mount point, which means you should not use $SYSNAME anywhere.

b. Issue the start command for the deployment manager on the system on which you want it to reside:

```
S <controller_procname>,JOBNAME=<jobname>,
ENV=<cell_shortname>.<node_shortname>.<server_shortname>
```

Scenario 3: Root HFS is not shared among any processors, deployment manager's configuration HFS is mounted and accessible to only one system at a time.

a. Unmount the configuration HFS from the original MVS image and remount it (at a mount point with the same name) on the new MVS image.

b. Issue the start command for the deployment manager on the system on which you want it to reside:

```
S <controller_procname>,JOBNAME=<jobname>,
ENV=<cell_shortname>.<node_shortname>.<server_shortname>
```

You know you are done when your deployment manager is up and running on a different MVS image and you are able to use it to make configuration updates to your environment.

Automatic restart management

WebSphere Application Server for z/OS uses the z/OS Automatic Restart Management (ARM) to recover application servers. Each application server running on a z/OS system (including servers you create for your business applications) are automatically registered with an ARM group. Each registration uses a special element type called SYSCB, which ARM treats as restart level 3, assuring that RRS restarts before any application server.

If you have an application that is critical for your business, you need facilities to manage failures. z/OS provides rich automation interfaces, such as automatic restart management, that you can use to detect and recover from failures. Automatic restart management handles the restarting of servers when failures occur.

Tips:

- If you have automatic restart management (ARM) enabled on your system, you might want to disable ARM for the WebSphere Application Server for z/OS address spaces before you install and customize WebSphere Application Server for z/OS. During customization, job errors might cause unnecessary restarts of the WebSphere Application Server for z/OS address spaces. After installation and customization, consider enabling ARM.
- If you are ARM-enabled and you cancel or stop a server, it will restart in place using the armrestart command.
- It is a good idea to set up an ARM policy for your deployment manager and node agents.
- If you start the location service daemon on a system that already has one, it will terminate.
- Every other server will come up on a dynamic port unless the configuration has a fixed port. Therefore, the fixed ports must be unique in a sysplex.

- If you issue STOP, CANCEL, or MODIFY commands against server instances, be aware of how automatic restart management behaves regarding WebSphere Application Server for z/OS server instances:

Table 5. Behavior of automatic restart management regarding WebSphere Application Server for z/OS server instances

If you issue . . .	**Automatic restart management will . . .**
STOP *address_space*	not restart the address space
CANCEL *address_space*	not restart the address space
CANCEL *address_space*, ARMRESTART	restart the address space
MODIFY *address_space*, CANCEL	not restart the address space
MODIFY *address_space*, CANCEL,ARMRESTART	restart the address space

Activating automatic restart management:

You must have access to the couple data set format utility, IXCL1DSU, in SYS1.MIGLIB. If you plan to modify the automatic restart management policy, you must have access to the administrative data utility, IXCMIAPU, also in SYS1.MIGLIB, and have UPDATE authorization to the RACF FACILITY class MVSADMIN.XCF.ARM. To start a policy, you must have READ authorization to the RACF FACILITY class MVSADMIN.XCF.ARM.

Though servers automatically register with automatic restart management, you must activate the arm component itself, which means you must:

1. Allocate an ARM couple data set.
2. Start the automatic restart management policy.

If automatic restart management is not active, WebSphere Application Server for z/OS issues an error message to the hardcopy log.

You are not required to change the automatic restart management policy. However, you will have to modify this policy if you want to create custom restart groups. For instance, it is not required or recommended that you start the node agent or deployment manager servers on another system. These servers will never have any transactional recovery to perform. Therefore, they should only be set up for restart-in-place. For complete information about how to modify the policies, see *z/OS MVS Setting Up a Sysplex* (SA22-7625).

The following procedure is intended to give you enough information to get automatic restart management running. Defining automatic restart management policies would require the z/OS manual mentioned above.

1. If you have not already formatted a couple data set for policies, do so now. For details, see *z/OS MVS Setting Up a Sysplex*
2. Submit the job to format the ARM couple data set.
3. **Optional:** Modify the automatic restart management policy. To get started, you do not need to modify the policy. If you do want to modify the automatic restart management policy, go to *z/OS MVS Setting Up a Sysplex*, and follow the instructions in that manual.
4. Issue the following operator commands to start the automatic restart management policy:

   ```
   SETXCF COUPLE,TYPE=ARM,PCOUPLE=(dsname,vvvvvv)
   SETXCF START,POLICY,TYPE=ARM
   ```

   where

   dsname

   Is the data set name for the couple data set.

   vvvvvv

   Is the volume serial of the volume on which the couple data set resides.

You are done when the SETXCF commands complete successfully.

Displaying the status of ARM-registered address spaces:

WebSphere Application Server for the z/OS platform ships with all control regions issuing Automatic Restart Management (ARM) registration commands. If your installation enables ARM, you can use ARM to display the status of all ARM-registered address spaces, including the address spaces of server instances.

ARM is used to restart all address spaces that are registered with ARM if they go down. Address spaces that are canceled are not restarted even if they are registered.

Each WebSphere Application Server for z/OS controller registers with ARM. If a controller terminates abnormally or the system fails, ARM will try to restart the failing address spaces. In doing this, ARM will ensure that dependent address spaces are grouped together and will start in the appropriate order. In general, the default ARM policy will restart WebSphere Application Server for z/OS in place. If you are using a sysplex, see “Automatic restart management” on page 126 for setup guidelines to ensure that no cross-system restarts are performed.

Perform the following steps to use ARM to display the status of ARM registered address spaces (including the address spaces of server instances:

1. Initialize all servers.
2. Display all registered address spaces (including the address spaces of server instances). Issue the following command:

   ```
   d xcf,armstatus,detail
   ```

How to change the automatic restart management policies: Because server instances register with the default restart group, automatic restart management attempts to restart the entire default group on another system in the sysplex when a system failure occurs. If you want to create a restart group other than this default group, you must follow the following rules and restrictions that apply for z/OS automatic restart management policies. For more information about how to actually change these policies, see, *z/OS MVS Setting Up a Sysplex* (SA22-7625).

- To change the policy, you need to know the existing element names for the server instances and how to name new elements for additional instances. The element names for these server instances are formed by concatenating the cell short name and the servers specific short name.

  If you have a cell named PLEX1 and server named BBOS001, for example, the ARM element name would be PLEX1BBOS001.

  Since wildcard characters can be used in the ARM policy, it is possible to exclude an entire group of servers by using a common naming scheme within your cell.

  The following section of the ARM policy will prevent any node agents from starting, for example, assuming that each node agent in your cell has a name that adheres to the form BBON*xxx*:

  ```
  RESTART_GROUP(WEBSPHERE)
  ELEMENT(PLEX1BBON*)
  RESTART_ATTEMPTS(0,150)
  RESTART_TIMEOUT(600)
  READY_TIMEOUT(1200)
  TERMTYPE(ALLTERM)
  RESTART_METHOD(BOTH,PERSIST)
  ```

  This ARM policy will also prevent the node agent from restarting in place. This specification can be modified by changing the RESTART_METHOD and TERMTYPE parameters. *See z/OS MVS Setting Up a Sysplex* (SA22-7625) for more information.

- If you create a restart group, keep the following in the same restart group and set the restart order for the elements as indicated:

  1. RRS

2. DB2 with IRLM, IMS, CICS, and other transaction or resource managers if used by your application servers in the restart group
3. Your server instances

Either set up the location service daemon and node agent for restart-in-place or remove them from your ARM policy. Since WebSphere Application Server must be running on all systems that might be used to perform recovery, the application servers will use the location service daemon and node agent that are already running on the alternate system. If the location service daemon attempts to restart on the alternate system, it will fail. If the node agent restarts on the alternate system, it will have no recovery work to do.

Displaying the status of ARM-registered address spaces:

WebSphere Application Server for the z/OS platform ships with all control regions issuing Automatic Restart Management (ARM) registration commands. If your installation enables ARM, you can use ARM to display the status of all ARM-registered address spaces, including the address spaces of server instances.

ARM is used to restart all address spaces that are registered with ARM if they go down. Address spaces that are canceled are not restarted even if they are registered.

Each WebSphere Application Server for z/OS controller registers with ARM. If a controller terminates abnormally or the system fails, ARM will try to restart the failing address spaces. In doing this, ARM will ensure that dependent address spaces are grouped together and will start in the appropriate order. In general, the default ARM policy will restart WebSphere Application Server for z/OS in place. If you are using a sysplex, see “Automatic restart management” on page 126 for setup guidelines to ensure that no cross-system restarts are performed.

Perform the following steps to use ARM to display the status of ARM registered address spaces (including the address spaces of server instances:

1. Initialize all servers.
2. Display all registered address spaces (including the address spaces of server instances). Issue the following command:

   ```
   d xcf,armstatus,detail
   ```

Problem diagnostic plan strategy

This article describes:

- Component trace
- Error log stream
- Dump data sets

Overview of problem diagnosis

WebSphere Application Server for z/OS uses component trace (CTRACE) to capture and display trace data in trace data sets. WebSphere Application Server for z/OS identifies itself to CTRACE with the short cell name. CTRACE allows you to perform the following tasks:

- Merge multiple traces through the browse tool, including other components such as TCP/IP and z/OS UNIX.
- Write trace data to a data set rather than to sysprint, keeping spool space free.
- Better manage system resources by allowing trace data to wrap or not wrap.
- Use CTRACE to funnel trace data from multiple address spaces to one data set, or have CTRACE send the trace data from each address space to separate data sets.
- Start and stop tracing without stopping and restarting WebSphere Application Server for z/OS address spaces.
- Use one or more data sets for capturing trace data, thus allowing you to manage I/O more effectively.

WebSphere Application Server for z/OS also has an error log stream that records the following error information when WebSphere Application Server for z/OS detects an unexpected condition or failure within its own code:

- Assertion failures
- Unrecoverable error conditions
- Vital resource failures, such as memory
- Operating system exceptions
- Programming defects in WebSphere Application Server for z/OS code

Use the error log stream in conjunction with other facilities available to capture error or status information—such as an activity log, trace data, system logrec, and job log.

The WebSphere Application Server for z/OS error log stream is a system logger application. Because the error log stream uses the system logger, you can perform the following tasks:

- Have error information written to a coupling facility log stream, which provides sysplex-wide error logging, or to a DASD-only log stream, which provides single system-only error logging.

  Note: There is a significant performance penalty when using DASD-only error logging.
- Set up either a common log stream for all of WebSphere Application Server for z/OS or individual log streams servers.

  Local z/OS client ORBs can also log data in log streams. Because the system logger APIs are unauthorized, any application can use them. You should control access to the log streams through a security product such as RACF.

WebSphere Application Server for z/OS provides a REXX EXEC (BBORBLOG) that allows you to browse the error log stream. By default, the EXEC formats the error records to fit a 3270 display.

This article describes the error log stream and how to set it up. Information about using the error log stream to diagnose problems is in the Troubleshooting section of the WebSphere Application Server information center. General information and guidance about the system logger is in *z/OS MVS Setting Up a Sysplex*.

Table 6. Finding WebSphere Application Server for z/OS Error Log Stream Information

What is your goal?	**You should read:**
Learn about the system logger and understand its requirements	*z/OS MVS Setting Up a Sysplex*
Learn about the WebSphere Application Server for z/OS error log stream	This article
Size the coupling facility structure space needed for the WebSphere Application Server for z/OS error log stream	*z/OS MVS Setting Up a Sysplex*
Define the WebSphere Application Server for z/OS error log stream	“Using the Customization Dialog” on page 133
View the WebSphere Application Server for z/OS error log stream	The Troubleshooting section of the WebSphere Application Server information center
Learn about how Java applications can log messages and trace data in the error log stream	The Applications section of the WebSphere Application Server information center

For details about problem diagnosis, see the Troubleshooting section of the WebSphere Application Server information center.

Planning for component trace

To use CTRACE, perform the following tasks:

- Specify trace options for identifying trace data sets and connecting WebSphere Application Server for z/OS address spaces to the data sets in parmlib members.
- Update WebSphere Application Server for z/OS WebSphere variables to allow for initial trace parameters.
- Use IPCS-CTRACE to view the trace data because you cannot read the trace data in an ordinary editor.

Recommendation for dumps

Plan as you would normally for system dumps. Due to the size of WebSphere Application Server for z/OS address spaces, you might need to resize your system dump data sets and use dynamic dump data sets.

Configuring the product after installation

Use this task to configure WebSphere Application Server for z/OS application serving environments for your z/OS target systems.

Configuring a WebSphere Application Server for z/OS application serving environment consists of setting up the WebSphere Application Server for z/OS configuration directory for the environment, making any required changes to the z/OS target system that pertain to the particular application serving environment, and starting the new environment to verify the configuration. Configuring these application serving environments after product installation requires a fair amount of planning and coordination. If you have not previously configured WebSphere Application Server for z/OS, you should configure a ″practice″ stand-alone application server using the default options then proceed to configure the actual product configuration that you want. See “Building a practice WebSphere Application Server for z/OS cell” on page 84 for more information.

WebSphere Application Server for z/OS application serving environment nodes can be created using either the ISPF-based Customization Dialog (see “Configuring with the ISPF Customization Dialog” on page 132) or the workstation-based Profile Management tool (see “Configuring with the Profile Management tool” on page 224).

Once a node is configured and running, make further changes using the Web-based administrative console or scripting.

Once your application serving environment is up and running, you can install and test applications.

Downloading archive images from the z/OS platform to a distributed platform

Follow these steps to set up either JDBC or WebSphere Application Clients on a distributed platform workstation.

The product tape for the WebSphere Application Server product that runs on z/OS includes image archives for the following functions that can be run on a distributed platform:

- WebSphere Application Clients for distributed platforms supported by the WebSphere Application Server Version 6 products.
- The Data Direct Java Database Connectivity (JDBC) drivers for Microsoft Windows platforms.

1. Create a directory on the workstation or server where you want to install the image archive files for WebSphere Application Client or Data Direct Java Database Connectivity (JDBC) drivers for Microsoft Windows platforms.
2. For example, on the z/OS HFS containing the WebSphere Application Server product, locate the WebSphere Application Client image archive files.

   The image archives for each supported platform are contained in the following directory on the HFS.

```
usr/lpp/zWebSphere/V6R0/downloads2...
.../platform
     /zdownloads.inst.cd.image.part1.for_platform.archive-ending(s))
     /zdownloads.inst.cd.image.part2.for_platform.archive-ending(s))
```

There are two install image archives for each supported platform. Each pair of install images archives includes that includes the WebSphere Application Client archive files for that platform. The install image archives also contain the Data Direct Java Database Connectivity (JDBC) drivers for Microsoft Windows platforms.

Both install image archives must be downloaded to the same directory on the distributed platform.

3. Use FTP or some other process to download the two image archives to the distributed platform. These files are in binary format and must be downloaded in that format.
4. Archive and expand both of the files into the directory you created in Step 1.

See Installing Application Client for WebSphere Application Server for a description of how to install the Application Client on the distributed platform.

Configuring with the ISPF Customization Dialog

Use this task to configure WebSphere Application Server for z/OS application serving environments for your z/OS target systems.

- Choose a z/OS target system and complete the steps in the ″Installing the product and additional software″ and ″Preparing the base operating system″ sections of the *Installing your application serving environment* PDF.
- Choose a WebSphere Application Server for z/OS configuration (practice, stand-alone or Network Deployment cell) and complete the steps in the ″Planning for product configuration″ section of the *Installing your application serving environment* PDF.

Note: The ISPF Customization Dialog is deprecated; it will be removed in a future release.

Configuring a WebSphere Application Server for z/OS application serving environment consists of setting up the WebSphere Application Server for z/OS configuration directory for the environment, making any required changes to the z/OS target system that pertain to the particular application serving environment, and starting the new environment to verify the configuration. Configuring these application serving environments after product installation requires a fair amount of planning and coordination. If you have not previously configured WebSphere Application Server for z/OS, you should configure a ″practice″ stand-alone application server using the default options then proceed to configure the actual product configuration that you want.

If you have already created a Network Deployment cell, follow the instructions in this section to expand the cell by creating a new managed node or federating an existing stand-alone application server node into the Network Deployment cell.

WebSphere Application Server for z/OS application serving environment nodes are created using the ISPF-based Customization Dialog. Once a node is configured and running, make further changes using the Web-based administrative console or scripting.

After you have installed the WebSphere Application Server for z/OS product, prepared your z/OS target systems, and planned your new application serving environment, perform the tasks in this section to configure and start the application serving environment.

1. Review the use of the Customization Dialog. See “Using the Customization Dialog” on page 133 for more information.
2. If this application serving environment uses a new security domain, create the security domain before proceeding. See “Creating common MVS groups and users” on page 138 for detailed instructions.
3. Follow the directions for the type of application serving environment you want to configure:
   - “Creating a stand-alone application server cell” on page 150

- “Creating a Network Deployment cell” on page 176
- “Creating a managed server node” on page 197
- “Federating a stand-alone application server into a Network Deployment cell” on page 215

Once your application serving environment is up and running, you can install and test applications. You might also want to configure your Web servers to interact with WebSphere Application Server for z/OS.

See the Related Tasks section for additional tasks you can perform once your application serving environment is configured.

Using the Customization Dialog

This article provides general information on starting and using the Customization Dialog. See the instructions for each customization task for detailed directions on using the Customization Dialog to perform that particular task.

The WebSphere Application Server for z/OS Customization Dialog is an ISPF dialog, running under TSO, that you use for the initial setup of WebSphere Application Server for z/OS cells and nodes. The Customization Dialog itself does not create the cells and nodes. Instead, it creates batch jobs, scripts, and data files that you then use to perform WebSphere Application Server for z/OS customization tasks.

Note: In WebSphere Application Server for z/OS, you must use the Customization Dialog and the jobs it generates to create new cells and nodes. Once you have created a stand-alone application server or Network Deployment cell, however, you use the WebSphere Application Server for z/OS administrative console or scripting to administer it.

The Customization Dialog consists of a set of ISPF panels, file-tailoring skeletons, message libraries, CLISTs and REXX execs that are installed as part of the WebSphere Application Server for z/OS product. It is intended for use under TSO by a systems programmer or WebSphere Application Server for z/OS administrator who is familiar with the z/OS target system on which the resulting WebSphere Application Server for z/OS cells and nodes will run.

The Customization Dialog uses ISPF variables to hold the various values used to create WebSphere Application Server for z/OS customization jobs, scripts and files. See “Customization Dialog variables” on page 135 for more information.

Follow these steps to perform most Customization Dialog customization and migration tasks:

Note: See the *Migrating, coexisting, and interoperating* PDF for information on the migration portion of the Customization Dialog.

1. Start the Customization Dialog. See “Starting the Customization Dialog” for instructions.
2. Choose a set of configuration data sets to hold the generated jobs and files for this task.
3. Set the dialog variables to appropriate values. See “Customization Dialog variables” on page 135 for instructions.
4. Generate the customized jobs, scripts and files based on the dialog variable values you provided, and place them in the configuration data sets.
5. Move the configuration data sets to the z/OS system on which you will perform WebSphere Application Server for z/OS customization or migration tasks.
6. Follow the generated instructions in the .CNTL configuration data set to complete the customization or migration task.

Starting the Customization Dialog: Before you can start the Customization Dialog, you must install the WebSphere Application Server for z/OS product and make the product libraries available to the z/OS system on which the Customization Dialog will run. All product libraries should be cataloged. The

Customization Dialog does not use the product directory (/usr/lpp/zWebSphere/V6R1 by default), but you must mount it and make it available before the Customization Dialog's generated jobs can run.

You will need the ability to log on to TSO from a real or emulated 3270-type terminal. Your logon display must support 3270 emulation and be set to a minimum of 32 rows by 80 columns (32 x 80) in order for the ISPF Customization Dialog to run.

- If your terminal has exactly 32 display rows, be sure the PF keys are not shown. These can overlay Customization Dialog input fields. To hide the PF keys, issue the `PFSHOW OFF` command in ISPF.
- If you have a 32-row display and use the ISPF split screen function, deselect "Always show split line" on the ISPF Settings panel and split the screen at the extreme top or bottom of the display. This prevents the split screen line from displaying and lines in the Customization Dialog from being obscured. Other uses of split screen will obscure lines in the Customization Dialog.

The following steps outline how to change your display size setting if you use the IBM Personal Communications Workstation program. Complete all the steps before you start your TSO session.

1. In the menu bar of the session window, select **Communication**.
2. From the Communication window, select **Configure**.
3. In the window that appears, press the **Session Parameters** button.
4. Use the pull-down menu to select a screen-size setting between 32x80 and 62x160.
5. Press the **OK** button until you are back at the session window.

You will need a TSO user ID that has READ access to the WebSphere Application Server for z/OS product libraries.

- After you have logged on with your user ID, you will see the ISPF menu options. Select **Settings**, and under terminal characteristics find **screen format**, and select **max**. This will maximize your screen for the customization dialogs.

Starting the dialog: The BBOWSTRT command

To start the Customization Dialog, log on to TSO and invoke the BBOWSTRT exec from the SBBOCLIB product library using the following EXEC command.

Note: If you enter the command from TSO Option 6, just type in the command. If you enter the command on the ISPF command line, you must prefix it with ″TSO″.

```
EXEC 'was_hlq.SBBOCLIB(BBOWSTRT)'  'options'
```

where the parameter:

was_hlq
is the data set name high-level qualifier for your WebSphere Application Server for z/OS product libraries.

Note: The dialog libraries (SBBOCLIB, SBBOPxxx, SBBOMxxx, SBBOSLIB and SBBOSLB2) must have the same data set name high-level qualifier in order for the Customization Dialog to allocate the libraries correctly.

Be sure to use the *was_hlq* value that corresponds to the level of WebSphere Application Server for z/OS product code that you will use to run the resulting WebSphere Application Server for z/OS cells.

You can also specify the following options, separated by spaces and surrounded in a single set of quotes:

APPL(*value*)
Specifies the application name you intend to use. This provides for separate sets of Customization Dialog variables saved from one Customization Dialog session to the next. The default value is BBO6. See "Customization Dialog variables" on page 135 for details.

LANG(*value*)

Specifies the national language you want the Customization Dialog to use. Specify ″ENUS″ for English or ″JAPN″ for Japanese. The default value is ENUS.

Note: Some messages will still appear in English even if you specify another national language, using the `LANG` option.

PROD(*list*)

Specifies a list of WebSphere Application Server for z/OS add-on products that are also customized using the dialog. The list should consist of one or more three-character product identifiers from the following table, separated by spaces. If you do not want to customize any add-on products, omit the `PROD` option.

BBZ	WebSphere Business Integration

PRODHLQ(*list*)

Specifies a list of high-level qualifiers for the WebSphere Application Server for z/OS add-on products you specified with the `PROD` option. The list should consist of one high-level qualifier for each entry in the PROD list, given in the same order. If you do not want to customize any add-on products, omit the `PRODHLQ` option.

If you are starting the dialog for the first time or using a new `APPL` value, you will first see a copyright screen. Press ENTER to continue.

You should then see the Customization Dialog main menu, similar to the following:

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>                                                  Appl: BBO6

   Use this dialog to create WebSphere Application Server for z/OS
   cells and nodes. Specify an option and press Enter.


   1  Configure common MVS Groups and Users

   2  Create stand-alone application server nodes. You must complete
      Option 1 before starting this option.

   3  Create Network Deployment cells and nodes. You must complete
      Option 1 before starting this option.

   4  Migrate a node.

   5  License and Notices.
```

If you do not see the main menu, review the steps and requirements above.

Customization Dialog variables:

The Customization Dialog provides a means to save dialog variable values from one session to the next using the Customization Dialog save files. ISPF can save these variables from one session to the next; in addition, the Customization Dialog can save WebSphere Application Server customization variables in data sets that can be reloaded during later customization sessions.

The WebSphere Application Server for z/OS Customization Dialog uses ISPF variables to store customization values. Although it is possible to reenter these values every time you use the Customization Dialog, this process can be time-consuming and error-prone.

Customization Dialog save files

The Customization Dialog provides its own method for saving and reusing dialog variable values. From most task menus, you can type `S` on the command line to save the current dialog variables in a file, and type `L` on the command line to restore dialog variables from a previously-created save file. These files are ordinary z/OS sequential files (RECFM=VB, LRECL=255) with one variable-value pair per line.

Because values in save files are explicitly saved and restored, you are somewhat less likely to use them accidently when customizing a different cell than values stored in the ISPF profile. Also, you can use the save file name to identify the WebSphere Application Server for z/OS cell or node for which they are created, the WebSphere Application Server for z/OS version and release, and so on. You can use a Customization Dialog save file, together with the WebSphere Application Server for z/OS product files, to recreate the WebSphere Application Server for z/OS customization data sets (.CNTL and .DATA) at any time.

One save file that is particularly important is the security domain save file. You should save the security domain values during the creation of each WebSphere Application Server for z/OS security domain and import them whenever you create a new stand-alone application server or Network Deployment cell in that security domain. Within the dialog panels for cell creation, you can view the security domain settings but you can not change them. The panels have specific numbered options for saving the newly created security domain values and restoring them during cell creation so that you do not forget.

Once you have finishing setting customization variable values for a new stand-alone application server or Network Deployment cell, save the variable values before proceeding with customization. This allows you to more easily restart the customization process if you discover you have made an error during customization. You can also import the appropriate saved values when performing tasks, such as federating a stand-alone application server or creating a new managed node, on an existing Network Deployment cell.

The following are some cautions that you should take when working with Customization Dialog save files:

- Customization Dialog save files might contain unencrypted passwords and other sensitive information. Make sure that access to the save files is restricted.
- Save files created with the `S` command include the security domain variables as well. If you need to restore values from both a regular save file and a security domain save file, therefore, restore the security domain save file last.
- Do not edit Customization Dialog save files directly—-instead, restore them in a Customization Dialog session, make the appropriate changes, and save the values again.

You can also use Customization Dialog save files to check for typing errors and similar problems by sorting the save file on the columns containing the dialog variable values. This can help detect typographic errors in product data set names and so on as well as detect ports or UNIX UID/GID values that are accidently reused within a particular configuration.

The instructions for each configuration task provide guidance on restoring and saving Customization Dialog variable values.

ISPF variable pools

Variable pools are a feature of ISPF that allow the dialog variables for one application to reside separate from those for other applications. Whenever you start the dialog using the ISPSTART command, you can use the `NEWAPPL` option to assign a one to four character name to the application and its variable pool. Variable pool names must be one to four characters long, alphanumeric, and begin with an alphabetic letter. The default variable pool name for the WebSphere Application Server for z/OS Customization Dialog is BBO6.

When you exit ISPF normally, the current editor settings and dialog variable values for each variable pool are saved in a pair of members in the user ID's ISPPROF (ISPF profile) data set. The data set name of the ISPPROF data set is usually something like userid.ISPF.ISPPROF, but it can vary from installation to installation. When saving settings for application *xxxx*, the editor settings are saved in member *xxxx*EDIT and the variable settings are saved in member *xxxx*PROF.

You can use the `APPL` option when starting the WebSphere Application Server for z/OS Customization Dialog to specify the application name (variable pool) that you want to use:

```
EXEC 'was_hlq.SBBOCLIB(BBOWSTRT)' 'APPL(xxxx)'
```

Whenever you start the Customization Dialog for the first time with a particular application name, the product copyright page is displayed before the main menu appears. Each Customization Dialog task menu panel shows the application name in the upper right-hand corner, labelled ″Appl:″.

By using different application names, you can separate variable pools for different WebSphere Application Server for z/OS releases, target systems, or cells. Just be sure to specify the same APPL value whenever you start the Customization Dialog in order to work with a particular WebSphere Application Server for z/OS customization task. You should also use customization save files (described in the following section) to provide long-term, more easily documented backup of customization variable values.

To delete all information from an ISPF variable pool, delete the *xxxx*EDIT and *xxxx*PROF members from the ISPPROF data set.

Note: Ensure when you do this that the ISPF application that uses the variable pool is not active.

If an ISPF dialog such as the WebSphere Application Server for z/OS Customization Dialog terminates abnormally, the variable pool members in the ISPPROF data set will not be updated.

Customization Dialog commands and function keys: The WebSphere Application Server for z/OS Customization Dialog uses only a small number of application-specific primary commands that are listed on the panels on which you can enter them. Of these, the most important are the `S` (Save) and `L` (RESTORE) commands for working with WebSphere Application Server for z/OS customization variable save files. See "Customization Dialog variables" on page 135 for more information.

The following default function keys are used throughout the Customization Dialog:

PF1
Displays a help screen for the current panel.

Note: Within the help panels, slightly different PF key settings are used. Press PF1 from within any help panel for details.

PF2
Sets split screen mode. See "Starting the Customization Dialog" on page 133 for restrictions on split screen mode in the Customization Dialog.

PF3
Exits the current panel.

PF4
Exits the Customization Dialog.

PF7
Scrolls up.

PF8
Scrolls down.

PF9
Swaps screens in split screen mode.

PF10
Scrolls left.

PF11
Scrolls right.

PF12
Retrieves the previous command.

You might also find the following ISPF primary commands useful:

`EPDF`
Allows you to browse or edit a z/OS data set from the ISPF command line

`MSGID`
Turns the display of message identifiers on and off.

`PANELID`
Turns the display of panel names on and off.

`PFSHOW OFF`
Turns off the display of PF key settings.

Note: This is necessary when using a 32-row display.

`PFSHOW ON`
Turns on the display of PF key settings.

`TSO`
Executes a TSO command, CLIST, or REXX command procedure from the command line.

`ZKEYS`
Allows you to display or change the current function key settings.

Creating common MVS groups and users

Perform this task to set up the operating system security prerequisites for a WebSphere Application Server for z/OS cell. This ensures that all servers in the cell are using the same operating system security definitions.

Install the WebSphere Application Server for z/OS product code and review the instructions for using the Customization Dialog. Have available a copy of the worksheet that you completed as part of planning for common groups and users.

You need to perform this task before configuring any application serving environment that uses the common groups and users. If a new WebSphere Application Server for z/OS cell or server on a z/OS system will use the exact same common group and user definitions as an existing server or cell on the same z/OS system, you do not need to repeat this task.

You need to run the jobs generated as part of this task once per security database. If z/OS systems do not share a RACF or other security database, you are responsible for making sure identical security definitions are in place for all WebSphere Application Server for z/OS user IDs, groups, and profiles. See the *Securing applications and their environment* PDF for more information about preparing the security server (RACF).

1. Log on to TSO on the z/OS system on which you intend to configure the common groups and users. Use a user ID that has READ access to the WebSphere Application Server for z/OS product data sets.
2. Start the Customization Dialog. See “Starting the Customization Dialog” on page 133 for details.

3. Choose the configuration data sets in which you will store your customization jobs and data. See “Choosing configuration data sets” for details.
4. Set the customization variables according to the values recorded on your common groups and users worksheet. See “Setting the customization variables: Common groups and users” on page 144 for details.
5. Save the common groups and users customization variables in a data set that you will use in later customization steps. See “Saving the common group and user variables” on page 145 for details.
6. Create the customization jobs and files, based on the customization variable values you entered. See “Creating the customization jobs and files” on page 145 for details.
7. Follow the generated customization instructions. See “Following the generated customization instructions: Security domain” on page 147 for details, and a sample set of customization instructions.

You have finished when you have successfully completed the steps in the generated instructions. The common groups and users is in place on the chosen z/OS system. If any z/OS systems that interoperate with or host your planned application serving environment do not share the security database you updated as part of this task, update the security databases of the other systems accordingly.

Proceed with the configuration of the application serving environments that use this common group and user.

Choosing configuration data sets:

This article leads you through the ″Allocate target data sets″ option in the Customization Dialog.

You must start the Customization Dialog and select the ″Configure a security domain″ option.

Each option in the Customization Dialog saves customization jobs and files in a pair of customization data sets. While is it possible to reuse these data sets, it is safest to create separate data sets for each WebSphere Application Server for z/OS configuration. The best practice is to use the customization data set name prefix (sometimes referred to as ″*config_hlq*″) to indicate the version and release of WebSphere Application Server for z/OS, the task you are performing, and the cell (and, in some cases, the node name) you are configuring. For example, you might use the following data set name prefix for configuring a WebSphere Application Server for z/OS 6.0.1 security domain for cell PRODCELL:

SYSPROG1.WAS601.PRODCELL.SECD

Complete this task before generating the customization jobs and files.

1. On the main dialog panel, type the appropriate number in the *Option* field to select ″Allocate target data sets″.
2. Press Enter. **Result:** You see a panel that looks similar to the following:

```
-----------------     WebSphere Application Server for z/OS Customization      ------------------
Option  ===>

 Allocate Target Data Sets

 Specify a high level qualifier (HLQ) and press Enter to allocate the
 data sets to contain the generated jobs and instructions. You can
 specify multiple qualifiers (up to 39 characters).

 High level qualifier:                                        .CNTL
                                                              .DATA

 The Dialog will display data set allocation panels. You can make
 changes to the default allocations, however you should not change
 the DCB characteristics of the data sets.

    .CNTL  - a PDS with fixed block 80-byte records to
```

```
                 contain customization jobs.

        .DATA  - a PDS with variable length data to contain
                 other data produced by the Customization Dialog.
```

3. Fill in your chosen configuration data set name prefix value (*config_hlq*). If the data sets *config_hlq*.CNTL and *config_hlq*.DATA do not exist, you will be prompted for data set allocation information. If the data sets already exist, a message will inform you that they will be reused.

The data sets *config_hlq*.CNTL and *config_hlq*.DATA are allocated and will store customization jobs and files. These data set names will also be saved along with the customization variables.

Security domains and the customization dialog: A security domain provides cell-level granularity of security permissions, which:

- Provides cell-level granularity of roles
- Allows different administrators to be assigned for test and production
- Is also used as the APPL profile for servers in the cell

Configure a security domain by using the customization dialog to customize your settings. This provides:

- A set of saved variables to be used when creating base and deployment manager configurations
- A new sample set of Resource Access Control Facility (RACF) customization jobs that must only be run once when the domain is created

Security information is stored in a new SavedVariables file because security domain information can span multiple cells (including test and production). You should be able to use the existing variables you have defined previously. (Make sure you save the values and record the location where it is saved.)

The RACF profiles that are created and checked differently because of this are:

- CBIND
- EJBROLE
- APPL

Tip: When setting up base application server, save the general customization variables and security domain variables to different files.

Use CBIND profiles to restrict access to servers if no other specific profile is set. If there is no security domain identifier, enter the following RACF commands:

```
/*  CBIND profiles in case no server definition is set                */
"RDEFINE CBIND CB.BIND.* UACC(NONE)"
"RDEFINE CBIND CB.* UACC(NONE)"
```

If there is a security domain identifier defined as TESTSYS, enter:

```
/*  CBIND CB.BIND.domain_name.                                  */
"RDEFINE CBIND CB.BIND.TESTSYS.* UACC(NONE)"
"RDEFINE CBIND CB.TESTSYS.* UACC(NONE)"
```

Use an APPL profile to protect WebSphere Application Server for z/OS. Sample profiles can grant APPL access to everyone if you use the universal access authority, UACC(NONE), and grant access to the configuration group, unauthenticated user IDs, and all valid WebSphere Application Server for z/OS user IDs.

For example, if there is no security domain, enter the following RACF commands:

```
RDEFINE APPL CB390 UACC(NONE)
PERMIT CB390 CLASS(APPL) ID(TSCLGP) ACCESS(READ)
```

And if there is a security domain identifier defined as TESTSYS, for example, enter:

```
RDEFINE APPL TESTSYS UACC(NONE)
PERMIT TESTSYS CLASS(APPL) ID(TSCLGP) ACCESS(READ)
```

EJBROLE profiles are defined for role-based authorization checks if there is no security domain identifier and the configuration group is defined as TSTCFG. Note that these are default values set at bootstrap, which is the minimum set of users requiring access to naming and administrative roles for a Local OS registry when System Authorization Facility (SAF) authorization is selected.

The following roles must be defined for both operating system and application security. Enter the following RACF commands:

```
RDEFINE EJBROLE administrator UACC(NONE)
RDEFINE EJBROLE monitor       UACC(NONE)
RDEFINE EJBROLE configurator UACC(NONE)
RDEFINE EJBROLE  operator     UACC(NONE)
RDEFINE EJBROLE  deployer     UACC(NONE)
RDEFINE EJBROLE  AdminSecurityManager     UACC(NONE)

PERMIT administrator  CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT monitor        CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT configurator  CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT operator        CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT deployer        CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT AdminSecurityManager       CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)

/* Setting up EJBRoles Profiles for Naming roles                        */
RDEFINE EJBROLE CosNamingRead   UACC(NONE)
PERMIT CosNamingRead  CLASS(EJBROLE)  ID(TSGUEST) ACCESS(READ)
RDEFINE EJBROLE CosNamingWrite  UACC(NONE)
RDEFINE EJBROLE CosNamingCreate UACC(NONE)
RDEFINE EJBROLE CosNamingDelete UACC(NONE)
```

If there is a security domain identifier defined as TESTSYS and the configuration group is defined as TSTCFG, enter the following RACF commands:

```
RDEFINE EJBROLE TESTSYS.administrator UACC(NONE)
RDEFINE EJBROLE TESTSYS.monitor       UACC(NONE)
RDEFINE EJBROLE TESTSYS.configurator UACC(NONE)
RDEFINE EJBROLE TESTSYS.operator     UACC(NONE)
RDEFINE EJBROLE TESTSYS.deployer     UACC(NONE)
RDEFINE EJBROLE TESTSYS.AdminSecurityManager     UACC(NONE)

PERMIT TESTSYS.administrator  CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT TESTSYS.monitor        CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT TESTSYS.configurator  CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT TESTSYS.operator        CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT TESTSYS.deployer        CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)
PERMIT TESTSYS.AdminSecurityManager        CLASS(EJBROLE)  ID(TSTCFG) ACCESS(READ)

/* Setting up EJBRoles Profiles for Naming roles                        */
RDEFINE EJBROLE TESTSYS.CosNamingRead   UACC(NONE)
PERMIT TESTSYS.CosNamingRead  CLASS(EJBROLE)  ID(TSGUEST) ACCESS(READ)
RDEFINE EJBROLE TESTSYS.CosNamingWrite  UACC(NONE)
RDEFINE EJBROLE TESTSYS.CosNamingCreate UACC(NONE)
RDEFINE EJBROLE TESTSYS.CosNamingDelete UACC(NONE)
```

Security server definition

A security domain definition in z/OS provides WebSphere Application Server for z/OS with a set of cell-wide z/OS Security Server (RACF) security definitions.

Assign distinct MVS user IDs to servers in isolatable security domains

The user IDs assigned to the control and servant tasks are defined by the SAF STARTED profile. A specific STARTED profile can be created for all controller procedures with a given procedure name. For example, enter the following RACF commands: RDEFINE STARTED TST5ACR.* STDATA(USER(TSSYMCR1) GROUP(TSTCFG) TRACE(YES))

Tip: Your controller procedure should be given a unique procedure name for each security domain.

Servant regions use different procedure names. In order to avoid having to customize the security for each new server created, use a generic profile with a generic server prefix (BBO*) to assign the user ID of all servers whose procedure names begin with BBO to a specific profile for other server names by entering the following RACF commands:

```
RDEFINE STARTED BBO*.* STDATA(USER(TSSYMSR1) GROUP(TSTCFG) TRACE(YES))
RDEFINE STARTED TSTS001S.* STDATA(USER(TSSYMSR1) GROUP(TSTCFG) TRACE(YES))
```

Note: You cannot modify the server and generic server prefix name BBO using the customization dialog. If you require isolation and do not want to require security customization when you add new servers, note that there is an implicit relationship between the server name and the MVS user ID of the servant task. If you do not need to use security customization when adding new servers, and you do not require a unique user ID for an authorization request to a particular server, create an alternate STARTED profile with a unique generic server prefix name by entering the following RACF commands:

```
RDEFINE STARTED TST*.* STDATA(USER(TSSYMSR1) GROUP(TSTCFG) TRACE(YES))
```

Each servant procedure in the TEST security domain must be defined with TST as the server prefix name used for all servers in this domain. Create a specific STARTED profile for each server for more control over the authorization of requests on different server by entering the following RACF commands:

```
RDEFINE STARTED TSTS001S.* STDATA(USER(TSSYMSR1) GROUP(TSTCFG) TRACE(YES))
RDEFINE STARTED TSTS002S.* STDATA(USER(TSSYMSR2) GROUP(TSTCFG) TRACE(YES))
```

Restrict SERVER access to security domains. In addition, the server class profiles are used to indicate which servant identities can access the appropriate Workload Manager (WLM) queues WebSphere Application Server for z/OS uses. In order to clearly isolate the security domains sets, note the relationship between server names and servant region MVS user IDs.

The SERVER profile checked is either in the form CB.servername.clustername.cellname or CB.servername.clustername, depending upon whether or not WLM Dynamic Application Environment support is enabled. For example, if your server name is TSTC001, the definitions are set by entering the following RACF commands:

```
RDEFINE SERVER CB.* UACC(NONE)
RDEFINE SERVER CB.*.BBO* UACC(NONE)
RDEFINE SERVER CB.*.BBO*.* UACC(NONE)
RDEFINE SERVER CB.*.TSTC001 UACC(NONE)(READ)
RDEFINE SERVER CB..*.TSTC001.*  UACC(NONE)
```

Permissions to access this server name are given by entering the following RACF commands:

```
PERMIT CB.*.TSTC001 CLASS(SERVER) ID(TSSYMSR1) ACC(READ)
PERMIT CB.*.TSTC001.* CLASS(SERVER) ID(TSSYMSR1) ACC(READ)
```

You can create additional SERVER definitions to accommodate a new server transition prefix, or create specific profiles per servant (similarly to how you can use the STARTED profile). Use a server transition prefix of TST to restrict the access of the TSSYMSR user ID to queues from TST servers. For example, to set this up enter the following RACF commands:

```
RDEFINE SERVER CB.*.TST* UACC(NONE)(READ)
RDEFINE SERVER CB.*.TST.*  UACC(NONE)
PERMIT CB.*.TST* CLASS(SERVER) ID(TSSYMSR1) ACC(READ)
PERMIT CB.*.TST*.* CLASS(SERVER) ID(TSSYMSR1) ACC(READ)
```

CBIND profile definitions for servers

If there is no security domain identifier, information is defined during bootstrap. Enter the following RACF commands:

```
RDEFINE CBIND CB.BIND.BBO* UACC(NONE)
RDEFINE CBIND CB.BIND.TSTC001 UACC(NONE)
PERMIT CB.BIND.BBO* CLASS(CBIND) ID(TSTCFG) ACCESS(CONTROL)
PERMIT CB.BIND.TSTC001 CLASS(CBIND) ID(TSTCFG) ACCESS(CONTROL)
RDEFINE CBIND CB.BBO* UACC(NONE)
RDEFINE CBIND CB.TSTC001 UACC(NONE)
```

If there is a security domain identifier defined as TESTSYS, enter:

```
RDEFINE CBIND CB.BIND.TESTSYS.BBO* UACC(NONE)
RDEFINE CBIND CB.BIND.TESTSYS.TSTC001 UACC(NONE)
PERMIT CB.BIND.TESTSYS.BBO* CLASS(CBIND) ID(TSTCFG) ACCESS(CONTROL)
PERMIT CB.BIND.TESTSYS.TSTC001 CLASS(CBIND) ID(TSTCFG) ACCESS(CONTROL)
RDEFINE CBIND CB.TESTSYS.BBO* UACC(NONE)
RDEFINE CBIND CB.TESTSYS.TSTC001 UACC(NONE)
```

Note:

- If you wish to create a new specific server with a prefix other than BBO*, define a specific CBIND profile by entering the following RACF commands:

  ```
  RDEFINE CBIND CB.BIND.TSTC002 UACC(NONE)
  PERMIT CB.BIND.TSTC002 CLASS(CBIND) ID(TSTCFG) ACCESS(CONTROL)
  RDEFINE CBIND CB.TSTC002 UACC(NONE)
  ```

- The samples create server definitions with specific server names (but a generic profile with a server prefix of BBO). If you have created an alternative server prefix and wish to avoid additional CBIND definitions, add generic CBIND profiles that reflect the new name by entering the following RACF commands:

  ```
  RDEFINE CBIND CB.BIND.TESTSYS.TST* UACC(NONE)
  PERMIT CB.BIND.TESTSYS.TST* CLASS(CBIND) ID(TSTCFG) ACCESS(CONTROL)
  RDEFINE CBIND CB.TESTSYS.TST* UACC(NONE)
  ```

Note:

- Different security domains should have different users and groups. While the domain identifier separates the RACF classes (CBIND, EJBROLE, APPL), it does not separate the file permissions for configuration files in the Hierarchical File System (HFS). For example, if:
  - The administrator is WSADMIN in group WSCFG
  - The Servant region identity is WASSRV (which must also belong to the WSCFG group)
  - The user TOM has READ access to the TEST.administrator EJBROLE but not to the PROD.administrator EJBROLE,

  TOM cannot use the administration application to make changes to the PROD cell.
- A rogue application running in the TEST application server can modify HFS files in the PROD cell. This is because the TEST server runs with the WASSRV user ID that belongs to the WSCFG group. Both the TEST and PROD HFS files can be modified by the WSCFG group. For maximum protection, PROD should be created and associated with a different RACF group from TEST.

Security customization dialog settings:

The Customization Dialog enables you to create a security domain for your WebSphere Application Server for z/OS configuration.

Note:

- You must set up a base Application Server using the dialogs before using this one to set up a Network Deployment node, which is managed by the deployment manager process (dmgr). It is critical that you **LOAD** saved environment variables from the base Application Server into the deployment manager node that federates the base node. Do this before performing security customization on the deployment manager node.
- If the APPL class is active and you have defined a profile for WebSphere Application Server, make sure that all z/OS identities using WebSphere Application Server services have READ permission to the WebSphere Application Server APPL profile. This includes all WebSphere Application Server identities, WebSphere Application Server unauthenticated identities, WebSphere Application Server administrative identities, user IDs based on role-to-user mappings, and all user identities for system users. If you have not defined a security domain, the APPL profile used is CBS390 or the name used as the security domain identifier. If you have defined a security domain, the APPL profile used is the security domain name.
- When adding an administrator to the administrative console using local operating system security, if the APPL class is activated, the administrator's user ID must be authorized to the CBS390 (or the name specified as the security domain identifier) APPL class for RACF as well. If the administrator's user ID is not authorized to CBS390 APPL, message BBOS0108E is issued, indicating that the credential-handling function (RunAsGetSpecCred) failed in routine because the user is not authorized.

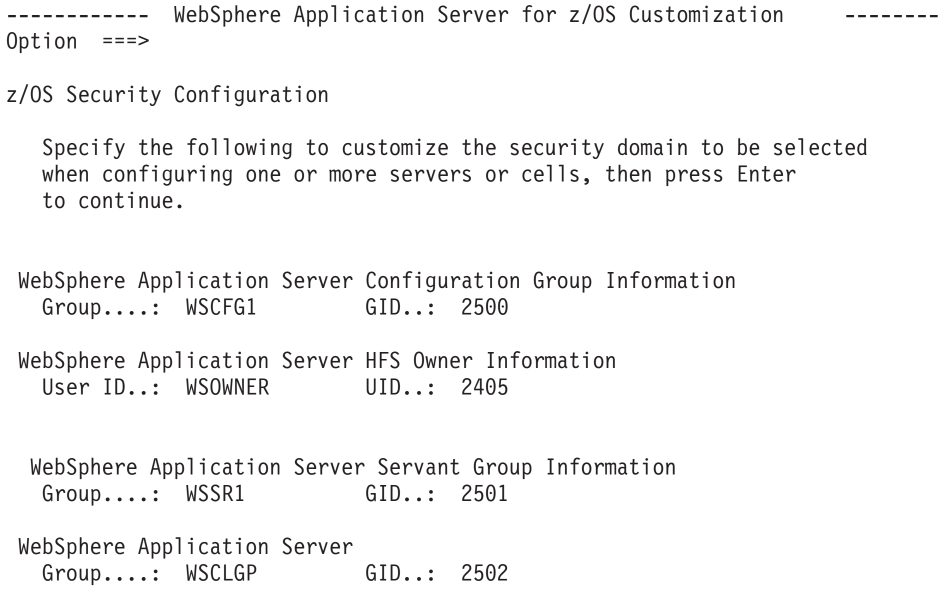
Setting the customization variables: Common groups and users:

This article describes how to complete the ″Define variables″ option for a WebSphere Application Server for z/OS security domain.

You must start the Customization Dialog and select the ″Create stand-alone Application Server nodes″ option. Have the Common MVS Groups and Users Customization Dialog worksheet completed and at hand. A copy of this worksheet is available in the *Installing your application serving environment* PDF.

1. On the ″Configure common groups and users″ panel, type the appropriate number in the *Option* field to select ″Define variables″ and press **Enter**.
2. Fill in the ″Common groups and users define variables″ panels using the following screen shots and tips as your guides. When you are done with each panel, press **Enter**.

Common groups and users define variables panel



```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

z/OS Security Configuration

   Specify the following to customize the security domain to be selected
   when configuring one or more servers or cells, then press Enter
   to continue.


 WebSphere Application Server Configuration Group Information
   Group....:  WSCFG1         GID..:  2500

 WebSphere Application Server HFS Owner Information
   User ID..:  WSOWNER        UID..:  2405


  WebSphere Application Server Servant Group Information
   Group....:  WSSR1          GID..:  2501

 WebSphere Application Server
   Group....:  WSCLGP         GID..:  2502
 WebSphere Application Server User ID home directory:
     /var/WebSphere/home
```

Note: The WebSphere Application Server user ID home directory field was added to the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.1.

Saving the common group and user variables:
You must start the Customization Dialog or Profile Management tool and fill in the common group and user variables.

The common group and user settings are used in the customization of every WebSphere Application Server for z/OS cell. By saving the common group and user variables, you create a saved security configuration that you can use consistently across all nodes in the cells you create for a given security domain.

Complete this task after setting the common group and user variables and before moving on to later customization steps. If you encounter problems during customization and change the common group and user variable values, be sure to re-save them.

Note: This procedure applies to only the ″Configure a common group and user″ dialog option. For information about saving variables for all the other dialog options, see “Saving the cell variables” on page 183

1. On the ″Configure Security Domain″ panel, type the appropriate number in the *Option* field to select ″Save security domain variables″ and press **Enter**. You will see a panel that looks similar to the following:

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   Save common group and user variables

     Specify the name of a sequential data set to contain the common group and
     user variables, then press Enter to continue. If the data set does not
     exist, the dialog displays the Allocate New Data Set panel, with which you
     can allocate a data set.


    Data set name:
   ```

2. Fill in the name of the sequential data set you will use to hold the common group and user variable values.

   Choose a data set name that identifies the sysplex, cell, or group of cells for which this security domain is defined. If the data set does not exist, you will be prompted for data set allocation information.

3. Record the name of the common group and user variable data set on your common group and user worksheet.

The common group and user settings are saved in the data set you selected.

Creating the customization jobs and files:
You must select configuration data sets to use and complete the process of defining variables for this task. See “Choosing configuration data sets” on page 139 and for more information.

The Customization Dialog creates customization batch jobs and data files, based on the variable values you specified in the dialog. The batch jobs and data sets will be written to the *config_hlq*.CNTL and *config_hlq*.DATA configuration data sets that you created with the ″Allocate target data sets″ option.

1. Ensure that the configuration data sets are allocated and not in use.

Note: Editing a member in *config_hlq*.CNTL or *config_hlq*.DATA will cause this task to fail.

2. On the ″Configure Security Domain″ panel, type the appropriate number in the *Option* field to select ″Generate customization jobs″ and press **Enter**. You will have one of two results:
   - **Result A:** If all variables are defined correctly, you see the ″Specify Job Cards″ panel, which looks similar to this:

     ```
     ------------  WebSphere Application Server for z/OS Customization     --------
     Option  ===>

     Generate Customization Jobs

      This portion of the Customization Dialog generates the jobs you must
      run after you complete this Dialog process. You must complete the
      customization process before you generate the jobs with this step.
      If you have not done this, please return to that step.

      Jobs and data files will get generated into data sets:
        'hlq.CNTL'
        'hlq.DATA'
      If you wish to generate customization jobs using other data sets, then
      exit from this panel and select the "Allocate target data sets" option.

      All the jobs that will be tailored for you will need a job card.
      Please enter a valid job card for your installation below. The
      file tailoring process will update the jobname for you in all the
      generated jobs, so you need not be concerned with that portion of
      the job cards below. If continuations are needed, replace the
      comment cards with continuations.

      Specify the job cards, then press Enter to continue.

      //jobname  JOB (ACCTNO,ROOM),'userid',CLASS=A,REGION=0M
      //*
      //*
      //*
     ```

     Note: Pay particular attention to the displayed target data sets. Make sure that they are the ones you intend to use.
   - **Result B:** If the variables are not defined correctly, you will see the ″Verification″ panel. Decide whether the warnings or errors are serious enough to warrant returning to the ″Define variables″ option.

     Note: If the return code is 8 or greater, return to the ″Define variables″ option and fix the uncovered problems. If you saved the variables previously, be sure to re-save them after making any updates.
3. Fill in the job card information, according to your installation requirements. For each job, the dialog generates a jobname and the ″JOB″ keyword to match the member name of the PDS, but you specify the rest.

   Note: If you need to run these jobs on a particular system in the sysplex (for example, JES2 MAS or JES3 complex), you should specify the necessary Scheduling Environment (SCHENV), JES2 JOBPARM, or JES3 //*MAIN statement at this time.

   Example of a job card entry:

   ```
   //jobname JOB 1234,USER1,NOTIFY=????,MSGCLASS=O,REGION=0M
   //*            USER=SYSADM1,PASSWORD=SYSADM1
   /*JOBPARM SYSAFF=SYSB
   ```

   Note: This example is useful for jobs that require a user ID other than that of the logged-on TSO user. (This is typically a user ID with UID=0.) In that case, you can just put a comma at the end of the first line, put in the correct user ID on the second line, then uncomment that second line. You might want to use RACF SUBMIT authority to avoid having to keep passwords in your configuration data sets.

4. Fix any errors. If there are errors anywhere, you will see the ″Error″ panel. Press PF3 to exit the error panel, then enter the correct panel to fix the errors. Then return to the ″Generate Customization Jobs″ option and pick up where you left off. If necessary, you can update the variables and rerun this option. The generation process will delete and re-tailor all the members.

   Note: Compress the configuration data sets before you rerun this option.

You are done when all the jobs are generated. You can move ahead to viewing the generated jobs. See “Following the generated customization instructions: Security domain” for more information.

Following the generated customization instructions: Security domain:
You must generate the customization jobs and files for this task.

The Customization Dialog creates a set of instructions for each customization task. Follow these instructions to tailor and customize a security domain on your system.

Note: Do not attempt to fix a typographical error or make a change by modifying the generated output. Many of the variables are used in multiple members of the target data sets; if you do not change them all, therefore, you will run into problems that are very difficult to diagnose.

1. On the ″Configure Security Domain″ panel, type the appropriate number in the *Option* field to select ″View instructions″ and press **Enter**. ISPF Browse will open and you will see the BBOSDINS member of *config_hlq*.CNTL.
2. Read the instructions carefully, both to preview the customization process and to find any typographical or other errors you might have made while entering the customization variable values.
3. Follow the instructions as given. There are two ways to follow the directions:
   - Follow the instructions while remaining in ISPF Browse.
   - Record the data set name and member at the top of the screen and either print the instructions or use ISPF split screen and browse or edit the instructions while you follow them.
4. Fix any problems. If you encounter problems caused by your Customization Dialog values, modify your variables using the dialog, regenerate the instructions, and restart the customization process.

   Note: Remember that you cannot generate new customization jobs while either configuration data set is open!

You are done with this customization task when you have successfully followed the generated instructions.

Sample generated instructions: Common MVS groups and users:

This article presents a sample of what the Customization Dialog’s generated instructions might look like. This is a sample only--you must use the instructions generated from your own variables when configuring your system.

```
-----------------------------------------------
Instructions for customizing a WebSphere for z/OS common MVS groups and
users.

The customization dialog has created jobs based on the information you
provided. These instructions tell you how to modify the operating
system and run the jobs to customize WebSphere for z/OS.

RULES:

1.  If you created the target data sets (*.CNTL and *.DATA) on another
    (driving) system, you must copy them to the target system and give
    them the same data set names.

2.  You must perform these instructions on your target system.

3.  You will have saved the z/OS security definition values
    in a data set. These values will need to be loaded and used
    when creating a stand-alone application server or a Network
    Deployment environment.

    ------------------------------------------------------------------

Running the customized jobs
---------------------------

The customization dialog built a number of batch jobs with the
variables you supplied. You must run the jobs in the order listed
below using user IDs with the appropriate authority.

The customization dialog for WebSphere for z/OS does not attempt to
update configuration data for your base operating system or existing
subsystems.

BEFORE YOU BEGIN: You must copy the target data sets (*.CNTL and
*.DATA) to your target system and give them the same data set names,
and you must be running on your target system.

Follow the table below, which lists in order the jobs you must submit
and the commands you must enter. Special handling notes are included
in the table. All jobs are members of

HOLLOW.WASBB61.CNTL.

Attention: After submitting each job, carefully check the output.
Errors may exist even when all return codes are zero.

+-----------+--------------------------------------------------------+
|           | The following three jobs (BBOSBRAJ, BBOSBRAK, BBOSBRAM) |
|           | do not need to be run if the indicated groups, user IDs |
|           | and directories already exist with the correct gid,    |
|           | uid and ownership permission values, as given below.   |
+-----------+--------------------------------------------------------+
| BBOSBRAJ  | User ID requirement: Authority to update data set      |
+-----------+                                                        |
| Done:     | USERID.WASBB61.DATA.                                   |
|           |                                                        |
|           | This job builds (but does not execute) the RACF commands |
| By:       | to create common WebSphere for z/OS groups and user IDs: |
|           |                                                        |
|           |  Configuration group:   WSCFG1 (gid 2500)              |
|           |  Servant group:         WSSR1 (gid 2501)               |
|           |  Local user group:      WSCLGP (gid 2502)              |
|           |  HFS owner user ID:     WSOWNER (uid 2405)             |
|           |                                                        |
|           | The commands are placed into member BBOSBRAK of data set |
|           | USERID.WASBB61.DATA.                                   |
|           |                                                        |
|           | Carefully review these definitions with your security  |
|           | administrator.                                         |
+-----------+--------------------------------------------------------+
| BBOSBRAK  | User ID requirement: RACF special authority.           |
+-----------+                                                        |
| Done:     | This job executes the RACF commands created by the     |
|           | previous job.  If the group and user IDs named above   |
|           | have already been created during a previous WebSphere  |
|           | for z/OS configuration and are in all target system RACF |
|           | database(s), you do not need to rerun this job.        |
|           |                                                        |
|           | RESULT: You may receive errors, such as INVALID USER     |
|           | messages, from this job because a user ID, group  or     |
|           | profile is already defined.  Make sure the existing      |
|           | user ID, group or profile has the same characteristics   |
|           | as the user ID, group or profile being created by        |
|           | BBOSBRAK.  If not, then change the values in the         |
|           | customization dialog which are causing the conflict,     |
|           | regenerate the customization jobs, and restart the       |
|           | process.                                                 |
|           |                                                          |
|           | When this step is complete, all groups and user IDs      |
|           | listed above for job BBOSBRAJ should be defined in the   |
|           | RACF database on each target system for the cell.        |
|           | Note: the WAS owner user ID WSOWNER MUST have the WAS    |
|           | configuration group WSCFG1 as its default OMVS group.    |
|           |                                                          |
| By:       |                                                          |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOSBRAM  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job creates home directories for WebSphere for z/OS |
|           | user IDS. These home directories will be subdirectories  |
|           | of /var/WebSphere/home                                   |
| By:       |                                                          |
|           | This job will:                                           |
|           |                                                          |
|           | Create the following directory with permission bits 755: |
|           |  /var/WebSphere/home                                     |
|           |                                                          |
|           |  Create the following directory with ownership           |
|           |  WSOWNER:WSCFG1 and permission bits 770:                 |
|           |                                                          |
|           |   /var/WebSphere/home/WSCFG1                             |
|           |                                                          |
|           |  Create the following directory with ownership           |
|           |  WSOWNER:WSSR1 and permission bits 770:                  |
|           |                                                          |
|           |   /var/WebSphere/home/WSSR1                              |
|           |                                                          |
|           |  Create the following directory with ownership           |
|           |  WSOWNER:WSCLGP and permission bits 770:                 |
|           |                                                          |
|           |   /var/WebSphere/home/WSCLGP                             |
|           |                                                          |
|           | This job should be run on each z/OS system that will     |
|           | host WebSphere Application Server nodes using these      |
|           | WebSphere for z/OS common groups and owner user ID.      |
|           | After execution, verify that the directories have been   |
|           | created with the correct permissions on each system.     |
|           |                                                          |
|           | If these directories already exist with the specified    |
|           | ownership and permission on a target system, then this   |
|           | job does not need to be run on that system.              |
|           |                                                          |
|           | ATTENTION: If the directory                              |
|           |  /var/WebSphere/home                                     |
|           | is used by applications other than WebSphere Application |
|           | Server, make sure that the permissions set by            |
|           | BBOSBRAM (755) are appropriate, or change them manually. |
|           | This directory must be world-readable for Websphere      |
|           | Application Server to run correctly.                     |
+-----------+----------------------------------------------------------+

+-----------+----------------------------------------------------------+
```

Creating a stand-alone application server cell

This article leads you through the tasks involved in setting up a WebSphere Application Server for z/OS stand-alone application server environment.

Ensure that the security domain was successfully created on the z/OS target system for the new stand-alone application server. Have available a copy of the worksheet that you completed as a part of your planning for a stand-alone application server cell.

Follow the steps below to set up a new WebSphere Application Server for z/OS stand-alone application server cell.

1. Log on to TSO on the z/OS system on which you intend to configure the stand-alone application server cell. Use a user ID that has READ access to the WebSphere Application Server for z/OS product data sets. You will also need access to a user ID with authority to make security system updates and a user ID with UID 0. (These can all be the same user ID.)
2. Start the Customization Dialog. See “Starting the Customization Dialog” on page 133 for details.
3. Choose the configuration data sets in which you will store your customization jobs and data. See “Choosing configuration data sets” on page 152 for details.
4. Load the security domain variables saved from the security domain you intend to use for this cell. See “Loading the common MVS groups and users variables” on page 151 for details.
5. Set the customization variables according to the values recorded on your stand-alone application server worksheet. See “Setting the customization variables: Stand-alone application server cell” on page 153 for details.
6. (Optional but recommended.) Save the stand-alone application server customization variables in a data set. See “Saving the cell variables” on page 183 for details.
7. Create the customization jobs and files, based on the customization variable values you entered. See “Creating the customization jobs and files” on page 162 for details.
8. Follow the generated customization instructions. See “Following the generated customization instructions: Stand-alone application server cell” on page 163 for details, and a sample set of customization instructions.

You are done when you have successfully completed the steps in the generated instructions. The new stand-alone application server is up and running on the chosen z/OS system. See “Working with your new server” on page 174 for more information. You can now deploy and test applications on your new stand-alone application server.

Loading the security domain variables:

This article describes how to complete the ″Load security domain variables″ option for a WebSphere Application Server for z/OS stand-alone application server cell.

Create the security domain you will use for the new stand-alone application server node and know the name of the saved security domain configuration variable file that you recorded on the security domain worksheet.

The security domain settings are used in the customization of every WebSphere Application Server for z/OS cell. By loading the security domain variables at the start of node or cell creation, you ensure that the security domain configuration you use is consistent and matches the RACF definitions that have already been set as part of security domain configuration.

Complete this task as the first step in configuring a new stand-alone application server node. If you encounter problems during customization and change the security domain variable values, be sure to re-save them.

1. On the ″Create a stand-alone application server node″ panel, type the appropriate number in the *Option* field to select ″Load security domain variables″ and press **Enter**. You will see a panel that looks similar to the following:

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Load Security Domain Variables

 Specify the name of a data set containing the security domain variables,
 then press Enter to continue.

 IBM-supplied defaults are in ''


 Data set name:


 If this data set is not cataloged, specify the volume.

 Volume:
```

2. Fill in the name of the sequential data set you used to hold the security domain variable values and press **Enter**. The security domain variables will load.

The security domain settings are loaded. You can display these variables, but not change them.

Loading the common MVS groups and users variables:

This article describes how to complete the ″Load common MVS groups and users variables″ option for a WebSphere Application Server for z/OS stand-alone application server cell.

Create the common MVS groups and users you will use for the new stand-alone application server node and know the name of the saved security domain configuration variable file that you recorded on the common MVS groups and users worksheet.

The common MVS groups and users settings are used in the customization of every WebSphere Application Server for z/OS cell. By loading the common MVS groups and users variables at the start of node or cell creation, you ensure that the configuration you use is consistent and matches the RACF definitions that have already been set as part of common MVS groups and users configuration.

Complete this task as the first step in configuring a new stand-alone application server node. If you encounter problems during customization and change the common MVS groups and users variable values, be sure to re-save them.

1. On the ″Create a stand-alone application server node″ panel, type the appropriate number in the *Option* field to select ″Load common MVS groups and users variables″ and press **Enter**. You will see a panel that looks similar to the following:

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Load common MVS groups and users variables

 Specify the name of a data set containing the common MVS groups and users
 variables, then press Enter to continue.

 IBM-supplied defaults are in ''


 Data set name:
```

```
 If this data set is not cataloged, specify the volume.

 Volume:
```

2. Fill in the name of the sequential data set you used to hold the variable values and press **Enter**. The common MVS groups and users variables will load.

The common MVS groups and users settings are loaded. You can display these variables, but not change them.

Choosing configuration data sets:

This article leads you through the ″Allocate target data sets″ option in the Customization Dialog.

You must start the Customization Dialog and select the ″Create stand-alone application server nodes″ option.

Each option in the Customization Dialog saves customization jobs and files in a pair of customization data sets. While is it possible to reuse these data sets, it is safest to create separate data sets for each WebSphere Application Server for z/OS configuration. The best practice is to use the customization data set name prefix (sometimes referred to as ″*config_hlq*″) to indicate the version and release of WebSphere Application Server for z/OS, the task you are performing, and the cell (and, in some cases, the node name) you are configuring.

For example, you might use the following data set name prefix for configuring a WebSphere Application Server for z/OS stand-alone application server with cellname SYSA and servername BB6QA1:

```
SYSPROG.WAS602.SYSA.BB6QA1.SAPPSRVR
```

Complete this task before generating the customization jobs and files.

1. On the main dialog panel, type the appropriate number in the *Option* field to select ″Allocate target data sets″.
2. Press Enter. **Result:** You see a panel that looks similar to the following:

```
-----------------     WebSphere Application Server for z/OS Customization      ------------------
Option  ===>

 Allocate Target Data Sets

 Specify a high level qualifier (HLQ) and press Enter to allocate the
 data sets to contain the generated jobs and instructions. You can
 specify multiple qualifiers (up to 39 characters).

 High level qualifier:                                          .CNTL
                                                                .DATA

 The Dialog will display data set allocation panels. You can make
 changes to the default allocations, however you should not change
 the DCB characteristics of the data sets.

    .CNTL  - a PDS with fixed block 80-byte records to
             contain customization jobs.

    .DATA  - a PDS with variable length data to contain
             other data produced by the Customization Dialog.
```

3. Fill in your chosen configuration data set name prefix value (*config_hlq*). If the data sets *config_hlq*.CNTL and *config_hlq*.DATA do not exist, you will be prompted for data set allocation information. If the data sets already exist, a message will inform you that they will be reused.

The data sets *config_hlq*.CNTL and *config_hlq*.DATA are allocated and will store customization jobs and files. These data set names will also be saved along with the customization variables.

Setting the customization variables: Stand-alone application server cell:

This article describes how to complete the ″Define variables″ option for a WebSphere Application Server for z/OS stand-alone application server node.

You must start the Customization Dialog and select the ″Create stand-alone Application Server nodes″ option. Have the Stand-alone application server cell Customization Dialog worksheet completed and at hand. A copy of this worksheet is available in the *Installing your application serving environment* PDF.

1. On the ″Create a stand-alone application server node″ panel, type the appropriate number in the *Option* field to select ″Define variables″ and press **Enter**.
2. On the ″Define Variables to Configure stand-alone Application Server Node″ panel, type the appropriate number in the *Option* field to select ″System Locations (directories, HLQs, etc.)″ and press **Enter**.
3. Fill in the ″System Locations″ panels using the following screen shots as your guides. When you are done with each panel, press **Enter**.

   System Locations

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   System Locations (1 of 2)

      Specify the following for the system on which you are installing
      WebSphere Application Server for z/OS, then press Enter to continue.
      For some data sets, specify "Y" if they are in STEPLIB.

      System name.:  AQTS      Sysplex name :  MCLXCF01

    Full Names of Data Sets

      PROCLIB:  SYS1.PROCLIB
      PARMLIB:  SYS1.PARMLIB
      SYSEXEC:

      Run WebSphere Application Server from STEPLIB (Y/N)?  Y
      SBBOLPA.:  BOSS.VICOM.W000170.SBBOLPA
      SBBOLOAD:  BOSS.VICOM.W000170.SBBOLOAD
      SBBGLOAD:  BOSS.VICOM.WOOO170.SBBGLOAD
      SBBOLD2.:  BOSS.VICOM.W000170.SBBOLD2
      SBBOEXEC:  BOSS.VICOM.W000170.SBBOEXEC
      SBBOMSG.:  BOSS.VICOM.W000170.SBBOMSG
   ```

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   System Locations (2 of 2)

      Specify the following for your customization, then press Enter
      to continue.

    Locations of HFS Resident Components

      WebSphere Application Server product directory:
        /usr/lpp/zWebSphere/V6R1
   ```
4. On the ″Define Variables to Configure stand-alone Application Server Node″ panel, type the appropriate number in the *Option* field to select ″System Environment Customization″ and press **Enter**.

5. Fill in the ″System Environment Customization″ panels using the following screen shots as your guides. When you have completed each panel, press **Enter**.
   - **System Environment Customization for WebSphere Application Server for z/OS Version 6.0.1**

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (1 of 4)

   Specify the following to customize your system environment, then
   press Enter to continue.

 WebSphere Application Server for z/OS Configuration HFS Information

   Mount point....:  /WebSphere/V6R0
   Name...........:  OMVS.WAS.CONFIG.HFS
   Volume, or '*' for SMS.:  *
   Primary allocation in cylinders...:  250
   Secondary allocation in cylinders.:  100
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (2 of 4)

   Specify the following to customize your system environment, then
   press Enter to continue.

 WebSphere Error Log Stream Information

        Name.................:  WAS.ERROR.LOG
        Data class ..........:  STANDARD
        Storage class........:
        HLQ for data sets....:  IXGLOGR

   Is log stream CF resident (Y|N):  Y

     If yes, specify structure name.:  WAS_STRUCT
     If no, specify: log stream size:  3000
                     staging size...:  3000

 RRS Log Stream Information

        Group name...........:  MCLXCF01
        Data class...........:  STANDARD
        Storage class........:
        HLQ for data sets....:  IXGLOGR

   Is log stream CF resident (Y|N):  Y

 Create RRS PROC (Y|N).......:  Y
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (3 of 4)

   Specify the following to customize your system environment, then
   press Enter to continue.

 CTRACE Writer Definitions

   Procedure name:  BBOWTR
   User ID.......:  STCRACF
   Group.........:  SYS1

   Trace Data Set Information
```

```
     Name...................:  SYS1.AQTS.WAS390.CTRACE
     Volume, or "*" for SMS.:  *
     Primary space in cylinders...:  10
     Secondary space in cylinders.:  0

   Trace Parmlib member suffix...:  60
```

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (4 of 4)

   Specify the following to customize your server, then press Enter
   to continue.

 Logging Details for Transaction XA Partner Log

 Use Log stream (Y|N):  N

   Log Stream Information

     Name HLQ......................:  WASTXA
     Data class ...................:
     Storage class.................:
     HLQ for data sets.............:  IXGLOGR

     Is log stream CF resident (Y|N):  Y

        If yes, specify structure name.:  WAS_STRUCT
        If no, specify: log stream size:  256
                        staging size...:  256
```

- **System Environment Customization for WebSphere Application Server for z/OS Versions 6.0.2 and Later**

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (1 of 4)

   Specify the following to customize your system environment, then
   press Enter to continue.

 WebSphere Application Server for z/OS Configuration HFS Information

   Mount point....:  /WebSphere/V6R0
   Name...........:  OMVS.WAS.CONFIG.HFS
   Volume, or '*' for SMS.:  *
   Primary allocation in cylinders...:  250
   Secondary allocation in cylinders.:  100
```

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (2 of 4)

   Specify the following to customize your system environment, then
   press Enter to continue.

 WebSphere Error Log Stream Information

       Name.................:  WAS.ERROR.LOG
       Data class ..........:  STANDARD
       Storage class........:
       HLQ for data sets....:  IXGLOGR

   Is log stream CF resident (Y|N):  Y
```

```
     If yes, specify structure name.:  WAS_STRUCT
     If no, specify: log stream size:  3000
                     staging size...:  3000
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (3 of 4)

   Specify the following to customize your system environment, then
   press Enter to continue.

 CTRACE Writer Definitions

   Trace Parmlib member suffix...:  60
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (4 of 4)

   Specify the following to customize your server, then press Enter
   to continue.

 Logging Details for Transaction XA Partner Log

 Use Log stream (Y|N):  N

   Log Stream Information

     Name HLQ......................:  WASTXA
     Data class ...................:
     Storage class.................:
     HLQ for data sets.............:  IXGLOGR

     Is log stream CF resident (Y|N):  Y

        If yes, specify structure name.:  WAS_STRUCT
        If no, specify: log stream size:  256
                        staging size...:  256
```

- **System Environment Customization for WebSphere Application Server for z/OS Versions 6.1**

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (1 of 3)

   Specify the following to customize your system environment, then
   press Enter to continue.

 WebSphere Application Server for z/OS Configuration HFS Information

   Mount point....:  /WebSphere/V6R1
   Name...........:  OMVS.WAS.CONFIG.HFS
   Volume, or '*' for SMS.:  *
   Primary allocation in cylinders...:  250
   Secondary allocation in cylinders.:  100
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (2 of 3)

   Specify the following to customize your system environment, then
   press Enter to continue.

 WebSphere Error Log Stream Information
```

```
        Log stream Dataset Name:  WAS.ERROR.LOG


 Transaction XA Partner Log Information

   Use Log stream (Y|N):  N

     Log stream HLQ....:    WASTXA


------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (3 of 3)

   Specify the following to customize your system environment, then
   press Enter to continue.

 CTRACE Writer Definitions

   Trace Parmlib member suffix...:  60
```

6. On the ″Define Variables to Configure stand-alone Application Server Node″ panel, type the appropriate number in the *Option* field to select ″Server Customization″ and press **Enter**.
7. Fill in the ″Server Customization″ panels using the following screen shots as your guides. When you are done with each panel, press **Enter**.

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (1 of 6)

   Specify the following to customize your server, then press Enter
   to continue.

 Application Server Definitions

   WebSphere Application Server home directory:
     /WebSphere/V6R0
         / AppServer

   Cell name (short)......:  AQTS
   Cell name (long).......:  AQTS

   Node name (short)......:  AQTS
   Node name (long).......:  AQTS

   Server name (short)....:  BBOS001
   Server name (long).....:  server1

   Cluster transition name:  BBOC001

   Admin asynch operations procedure name:  BBOW6SH

   Install samples?  (Y/N):  Y
```

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (2 of 6)

   Specify the following to customize your server, then press Enter
   to continue.

 Application Server Definitions

   Controller Information
```

```
     Jobname.......:  BBOS001
     Procedure name:  BBO6ACR
     User ID.......:  ASCR1
     UID...........:  2431

   Servant Information

     Jobname.......:  BBOS001S
     Procedure name:  BBO6ASR
     User ID.......:  ASSR1
     UID...........:  2432

   Control Region Adjunct

     Jobname.......:  BBOS001A
     Procedure name:  BBO6CRA
     User ID.......:  ASCRA1
     UID...........:  2433
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (3 of 6)

   Specify the following to customize your server, then press Enter
   to continue.

 Application Server Definitions

   Node host name..........:

     SOAP JMX Connector port...............:  8880

   ORB Listener host name..:  *

     ORB port..............................:  2809
     ORB SSL port..........................:  0

   HTTP transport host name:  *

     HTTP port.............................:  9080
     HTTP SSL port.........................:  9443

   Service Integration port...........................:  7276
   Service Integration Secure port....................:  7286
   Service Integration MQ Interoperability port.......:  5558
   Service Integration MQ Interoperability Secure port:  5578
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (4 of 6)

   Specify the following to customize your server, then press Enter
   to continue.

 Application Server Definitions

 Specify your High Availability Manager Host here.  This MUST
 resolve to a single IP address; it cannot be a multihomed host

 High Availability Manager Host:

 High Availability Manager Communication Port:  9353
```

Note: The ″High Availability Manager Host″ field was removed from the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (5 of 6)

   Specify the following to customize your server, then press Enter
   to continue.

 Location Service Daemon Definitions

   Daemon home directory:
     /WebSphere/V6R0/Daemon

   Daemon jobname:  BBODMNB

   Procedure name.:  BBO6DMN
   User ID........:  WSDMNCR1
   UID............:  2411

   IP name........:  sdf
   Port...........:  5655
   SSL port.......:  5656

   Register daemon with WLM DNS:  N
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (6 of 6)

   (Note:  This panel is optional if you are not configuring a
    database for the Scheduler component)

   Specify the following for the system on which you wish to
   configure your Scheduler database, then press Enter to continue.

 Full Names of Datasets

   SBPXEXEC...........:  SYS1.SBPXEXEC
   DB2 RUNLIB Location:  DB2HLQ.RUNLIB.LOAD

 Scheduler Database Definitions

   DB2 Subsystem Name.:  DSN
   Plan Name..........:  DSNTIA81

   Scheduler Database Name:  SCHEDDB

   Storage Group Name.....:  SYSDEFLT
   Tablespace Name........:  SCHEDTS
   Table Prefix...........:  TBLPRFIX
```

Server Customization

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (1 of 5)

   Specify the following to customize your server, then press Enter
   to continue.

 Application Server Definitions

   WebSphere Application Server home directory:
     /WebSphere/V6R1
         / AppServer

   Cell name (short)......:  AQTS
   Cell name (long).......:  AQTS

   Node name (short)......:  AQTS
   Node name (long).......:  AQTS

   Server name (short)....:  BBOS001
   Server name (long).....:  server1

   Cluster transition name:  BBOC001

   Admin asynch operations procedure name:  BBOW7SH

   WebSphere Application Server Asynchronous Administration Task
      User ID:  WSADMSH        UID:   2504
   Install samples?  (Y/N):  Y
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (2 of 5)

   Specify the following to customize your server, then press Enter
   to continue.

 Application Server Definitions

   Controller Information

     Jobname.......:  BBOS001
     Procedure name:  BBO7ACR
     User ID.......:  ASCR1
     UID...........:  2431

   Servant Information

     Jobname.......:  BBOS001S
     Procedure name:  BBO7ASR
     User ID.......:  ASSR1
     UID...........:  2432

   Control Region Adjunct

     Jobname.......:  BBOS001A
     Procedure name:  BBO7CRA
     User ID.......:  ASCRA1
     UID...........:  2433
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (3 of 5)

   Specify the following to customize your server, then press Enter
   to continue.

 Application Server Definitions

   Node host name..........:  *

     SOAP JMX Connector port...............:  8880

   ORB Listener host name..:  *

     ORB port..............................:  2809
     ORB SSL port..........................:  0

   HTTP transport host name:  *
```

```
     Administrative console port...........:  9060
     Administrative console secure port....:  9043
     HTTP transport port...................:  9080
     HTTPS transport port..................:  9443

   High Availability Manger Communication Port........:  9353
   Service Integration port...........................:  7276
   Service Integration Secure port....................:  7286
   Service Integration MQ Interoperability port.......:  5558
   Service Integration MQ Interoperability Secure port:  5578
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (4 of 5)

   Specify the following to customize your server, then press Enter
   to continue.

 Application Server Definitions

 Specify your High Availability Manager Host here.  This MUST
 resolve to a single IP address; it cannot be a multihomed host

 High Availability Manager Host:

 High Availability Manager Communication Port:  9353
```

Note: The ″High Availability Manager Host″ field was removed from the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (5 of 5)

   Specify the following to customize your server, then press Enter
   to continue.

 Location Service Daemon Definitions

   Daemon home directory:
     /WebSphere/V6R1/Daemon

   Daemon jobname:  BBODMNB

   Procedure name.:  BBO6DMN
   User ID........:  WSDMNCR1
   UID...........:  2411

   IP name........:  xxx
   Port...........:  5655
   SSL port.......:  5656

   Register daemon with WLM DNS:  N
```

8. Fill in the ″Web Server Configuration″ panel using the following screen shot as your guide. When you are done with the panel, press **Enter**.

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

 Web Server Configuration (1 of 1)

 If you are running IBM HTTP Server for z/OS on a local or remote z/OS system
 and wish to have WebSphere Application Server for z/OS manage the
 plugin-cfg.xml file, fill in the following values.

  Web Server Name:  webserver1
   The name used in defining the Web server in the admin console.
```

```
Host.......... :  myhost.acme.com
 IP name or address of the z/OS system on which the Web server is located.

Port.......... :  80
 HTTP port on which the Web server is listening.

Application Mapping ((A)ll, (N)one, (D)efault): A
 Determines whether you want to map all applications, none of
 the applications, or the Default Application to the web server
```

9. On the ″Define Variables to Configure stand-alone Application Server Node″ panel, type the appropriate number in the *Option* field to select ″View Security Domain Configuration Panels″ and press **Enter**. These panels display values you previously set in the ″Configure a security domain″ option--you cannot change any of the values here. If you do want to make changes, you must go back to the main dialog panel and run through the ″Configure a security domain″ option again.

Creating the customization jobs and files:
You must select configuration data sets to use and complete the process of defining variables for this task. See “Choosing configuration data sets” on page 152 and “Setting the customization variables: Stand-alone application server cell” on page 153 for more information.

The Customization Dialog creates customization batch jobs and data files, based on the variable values you specified in the dialog. The batch jobs and data sets will be written to the *config_hlq*.CNTL and *config_hlq*.DATA configuration data sets that you created with the ″Allocate target data sets″ option.

1. Ensure that the configuration data sets are allocated and not in use.

   Note: Editing a member in *config_hlq*.CNTL or *config_hlq*.DATA will cause this task to fail.
2. On the ″Create a stand-alone application server node″ panel, type the appropriate number in the *Option* field to select ″Generate customization jobs″ and press **Enter**. You will have one of two results:
   - **Result A:** If all variables are defined correctly, you see the ″Specify Job Cards″ panel, which looks similar to this:

     ```
     ------------  WebSphere Application Server for z/OS Customization     --------
     Option  ===>

     Generate Customization Jobs

      This portion of the Customization Dialog generates the jobs you must
      run after you complete this Dialog process. You must complete the
      customization process before you generate the jobs with this step.
      If you have not done this, please return to that step.

      Jobs and data files will get generated into data sets:
        'hlq.CNTL'
        'hlq.DATA'
      If you wish to generate customization jobs using other data sets, then
      exit from this panel and select the "Allocate target data sets" option.

      All the jobs that will be tailored for you will need a job card.
      Please enter a valid job card for your installation below. The
      file tailoring process will update the jobname for you in all the
      generated jobs, so you need not be concerned with that portion of
      the job cards below. If continuations are needed, replace the
      comment cards with continuations.

      Specify the job cards, then press Enter to continue.

      //jobname  JOB (ACCTNO,ROOM),'userid',CLASS=A,REGION=0M
      //*
      //*
      //*
     ```

Note: Pay particular attention to the displayed target data sets. Make sure that they are the ones you intend to use.

- **Result B:** If the variables are not defined correctly, you will see the ″Verification″ panel. Decide whether the warnings or errors are serious enough to warrant returning to the ″Define variables″ option.

  Note: If the return code is 8 or greater, return to the ″Define variables″ option and fix the uncovered problems. If you saved the variables previously, be sure to re-save them after making any updates.

3. Fill in the job card information, according to your installation requirements. For each job, the dialog generates a jobname and the ″JOB″ keyword to match the member name of the PDS, but you specify the rest.

   Note: If you need to run these jobs on a particular system in the sysplex (for example, JES2 MAS or JES3 complex), you should specify the necessary Scheduling Environment (SCHENV), JES2 JOBPARM, or JES3 //*MAIN statement at this time.

   Example of a job card entry:

   ```
   //jobname JOB 1234,USER1,NOTIFY=????,MSGCLASS=O,REGION=0M
   //*             USER=SYSADM1,PASSWORD=SYSADM1
   /*JOBPARM SYSAFF=SYSB
   ```

   Note: This example is useful for jobs that require a user ID other than that of the logged-on TSO user. (This is typically a user ID with UID=0.) In that case, you can just put a comma at the end of the first line, put in the correct user ID on the second line, then uncomment that second line.

   You might want to use RACF SUBMIT authority to avoid having to keep passwords in your configuration data sets.

4. Fix any errors. If there are errors anywhere, you will see the ″Error″ panel. Press PF3 to exit the error panel, then enter the correct panel to fix the errors. Then return to the ″Generate Customization Jobs″ option and pick up where you left off. If necessary, you can update the variables and rerun this option. The generation process will delete and re-tailor all the members.

   Note: Compress the configuration data sets before you rerun this option.

You are done when all the jobs are generated. You can move ahead to viewing the generated jobs. See “Following the generated customization instructions: Stand-alone application server cell” for more information.

Following the generated customization instructions: Stand-alone application server cell: You must generate the customization jobs and files for this task.

The Customization Dialog creates a set of instructions for each customization task. Follow these instructions to tailor and customize a stand-alone application server node on your system.

Note: Do not attempt to fix a typographical error or make a change by modifying the generated output. Many of the variables are used in multiple members of the target data sets; if you do not change them all, therefore, you will run into problems that are very difficult to diagnose.

1. On the ″Create a stand-alone application server node″ panel, type the appropriate number in the *Option* field to select ″View instructions″ and press **Enter**. ISPF Browse will open and you will see the BBOSSINS member of *config_hlq*.CNTL.
2. Read the instructions carefully, both to preview the customization process and to find any typographical or other errors that you might have made while entering the customization variable values.
3. Follow the instructions as given.

   There are two ways to follow the directions:

   - Follow the instructions while remaining in ISPF Browse.

- Record the data set name and member at the top of the screen and either print the instructions or use ISPF split screen and browse or edit the instructions while you follow them.

4. Fix any problems.

   If you encounter problems caused by your Customization Dialog values, modify your variables using the dialog, regenerate the instructions, and restart the customization process.

   Note: Remember that you cannot generate new customization jobs while either configuration data set is open!

You are done with this customization task when you have successfully followed the generated instructions.

Sample generated instructions: Stand-alone application server cell:

This article presents a sample of what the Customization Dialog's generated instructions might look like. This is a sample only--you must use the instructions generated from your own variables when configuring your system.

Note: This sample is based on the installation of WebSphere Application Server for z/OS Version 6.0.2. Generated instructions from different releases of Version 6 would have different content.

Instructions for customizing WebSphere for z/OS for a stand-alone Application Server node for 6.1.

```
----------------------------------------------


The customization dialog has created jobs based on the information you
provided. These instructions tell you how to modify the operating
system and run the jobs to customize WebSphere for z/OS.

RULES:

1.  If you created the target data sets (*.CNTL and *.DATA) on another
    (driving) system, you must copy them to the target system and give
    them the same data set names.

2.  You must perform these instructions on your target system.

Doing manual configuration updates
----------------------------------

The customization dialog for WebSphere for z/OS does not attempt to
update configuration data for your base operating system or existing
subsystems. You must do the following manual steps prior to running
the WebSphere for z/OS configuration jobs.

Perform these steps to do manual configuration updates:

1.  Update BLSCUSER. Refer to member BBOIPCSP in

    'MINGLW.TODAY.CNTL'

    In order to use the IPCS support provided by the product, append
    the contents of this member to the BLSCUSER member in your IPCSPARM
    or system PARMLIB datasets.

    -------------------------------------------------------------------

2.  Update SCHEDxx. Refer to member BBOSCHED in

    'USERID.TODAY.CNTL'

    In order to set the correct program properties for the WebSphere
    for z/OS run-time executables, append the contents of this member
    to the SCHEDxx member in your system PARMLIB concatenation.

    Note: When you are finished, issue the command SET SCH=(xx,xx)
    to activate SCHEDxx and load a new program properties table.

    --------------------------------------------------------------------

3.  Make sure the following data sets are APF-authorized:

        DBEL.WASX.SBBOLPA
        DBEL.WASX.SBBOLOAD
        DBEL.WASX.SBBGLOAD
        DBEL.WASX.SBBOLD2

    Add these datasets to your PROGxx or IEAAPFxx parmlib members, as
    appropriate, ensuring you specify the correct volsers.

    --------------------------------------------------------------------

4.  If you want to collect the SMF120 records created by the run-time
    servers, update SMFPRMxx via the following:
    EXAMPLE:

       SUBSYS(STC,EXITS(IEFU29,IEFACTRT),INTERVAL(SMF,SYNC),
                          TYPE(0,30,70:79,88,89,120,245))
                                                ---

    For details on the SMF records, see related topics in the
    WebSphere for z/OS Information Center at
    http://www.ibm.com/software/webservers/appserv/zos_os390/library/

    --------------------------------------------------------------------

5.  Update your active BPXPRMxx member to have the following WebSphere
    for z/OS configuration file system:

    OMVS.WAS.CONFIG.HFS

    mounted at:

    /WebSphere/V6R1

    in read/write mode.

    EXAMPLE:

       MOUNT FILESYSTEM('OMVS.WAS.CONFIG.HFS')
         MOUNTPOINT('/WebSphere/V6R1')
          TYPE(HFS)
          MODE(RDWR)

    --------------------------------------------------------------------

6.  Update TCP/IP by reserving the following ports for WebSphere for
    z/OS:

       SOAP JMX Connector port                              - 8880
       ORB port                                             - 2809
       Administrative console port                          - 9060
       Administrative console secure port                   - 9043

       HTTP Transport port                                  - 9080
       HTTPS Transport port                                 - 9443

       High availability manager communication port         - 9353
       Service Integration port                             - 7276
       Service Integration Secure port                      - 7286
       Service Integration MQ Interoperability port        - 5558
       Service Integration MQ Interoperability Secure port - 5578
       Session Initiation Protocol (SIP) port              - 5060
       Session Initiation Protocol (SIP) secure port       - 5061

       Daemon IP port                                      - 5655
       Daemon SSL port                                     - 5656

    View member BBOTCPIP in

    'USERID.TODAY.CNTL'

    Add the contents of this member to the PORT section of the file
    referenced by the DD statement for the TCP/IP profile in the
    TCP/IP start procedure. Cut and paste from this member into the
    data set used by your installation.

    ATTENTION: If another application has already reserved any of these
    ports for its own use, you must resolve the resulting conflict
    before you continue. If you update the WebSphere for z/OS
    customization dialog with new port specifications, be sure to
    regenerate the customization jobs, data, and instructions.

    ---------------------------------------------------------------------

7.  The WebSphere product libraries will be placed in STEPLIB as
    needed, rather than in the system link pack area and system link
    list. Make sure that the target MVS system has at least 8MB
    of free storage in extended CSA for the daemon and for EACH
    node (deployment manager node or application server node).

    SBBOLOAD and SBBOLD2:
    ====================
    The following data sets will be placed in the STEPLIB concatenation
    for the location service daemon, controller and servant regions,
    and in the setupCmdLine.sh script in the WebSphere Configuration
    file system. You must not remove these STEPLIB statements.

    DBEL.WASX.SBBOLOAD
    DBEL.WASX.SBBOLD2


    BBORTS61:
    =========

    The BBORTS61 module is used by WebSphere Application Server for
    component trace support. A copy of this module (any maintenance
    level) must be in the system link pack area in order for CTRACE
    to work correctly.

    If a copy of BBORTS61 is currently loaded into LPA, you need take
    no further action.

    Otherwise, issue the following MVS console command to load BBORTS61
    into dynamic LPA:

      SETPROG LPA,ADD,MODNAME=BBORTS61,
              DSNAME=DBEL.WASX.SBBOLPA

    Alternatively, you can place the following statement in a parmlib
    PROGxx member which is activated with the SET PROG= command after
    system IPL is complete:

    LPA ADD MODNAME(BBORTS61)
      DSNAME(DBEL.WASX.SBBOLPA)

    Make sure that the BBORTS61 module is loaded into LPA after each
    system IPL.

    --------------------------------------------------------------------

8.  WebSphere for z/OS customization assumes that the following system
    data sets are in the system link list or link pack area:

    Language Environment     SCEERUN
                             SCEERUN2

    System SSL               SGSKLOAD (z/OS 1.5 and below)
                             SIEALNKE (z/OS 1.6 and above)

    Placing these data sets in the link list or link pack area improves
    performance and insulates your WebSphere for z/OS configuration
    from changes in data set names (for example, when migrating to z/OS
    1.6).

    If the Language Environment or System SSL load module libraries are
    not in your system link list or link pack area, you must perform
    the following steps before starting any WebSphere Application
    Server for z/OS servers:

    - Make sure the data sets are APF-authorized
    - Complete the optional step below to add the data sets to STEPLIB
      in the server JCL and setupCmdLine.sh script(s).

    If you regenerate server cataloged procedures at any point, make
    sure the data sets are added to the new cataloged procedures.

    --------------------------------------------------------------------

9.  If the error logstream
    WAS.ERROR.LOG
    does not already exist on your target system, make a copy of the
    appropriate job in the SBBOJCL data set, customize it according
    to the comments in the job, and run it:

       BBOERRLC    Create an error logstream in a coupling facility
       BBOERRLD    Create a DASD-only error logstream

    --------------------------------------------------------------------

10. WebSphere for z/OS regions open a large number of files (more than
    1024). Make sure your BPXPRMxx parmlib member(s) specify a value of
    MAXFILEPROC that is greater than or equal to 2000. Use the
    following MVS console command to see your current MAXFILEPROC
    setting:

      D OMVS,OPTIONS

    --------------------------------------------------------------------

Running the customized jobs
---------------------------

The customization dialog built a number of batch jobs with the
variables you supplied. You must run the jobs in the order listed
below using user IDs with the appropriate authority.

BEFORE YOU BEGIN: Complete the section above entitled "Doing manual
configuration updates".

Follow the table below, which lists in order the jobs you must submit
and the commands you must enter. Special handling notes are included
in the table. All jobs are members of
```

```
USERID.TODAY.CNTL

Attention: After submitting each job, carefully check the output.
Errors may exist even when all return codes are zero.

+-----------+----------------------------------------------------------+
|           | The next three jobs (BBOSBRAJ, BBOSBRAK, BBOSBRAM)       |
|           | do not need to be run if the indicated groups, user IDs  |
|           | and directories already exist with the correct gid,      |
|           | uid and ownership permission values, as given below.     |
+-----------+----------------------------------------------------------+
| BBOSBRAJ  | User ID requirement: Authority to update data set        |
+-----------+                                                          |
| Done:     | USERID.TODAY.DATA.                                       |
|           |                                                          |
|           | This job builds (but does not execute) the RACF commands |
| By:       | to create common WebSphere for z/OS groups and user IDs: |
|           |                                                          |
|           |  Configuration group:       WSCFG1 (gid 2500)            |
|           |  Servant group:             WSSR1 (gid 2501)             |
|           |  Local user group:          WSCLGP (gid 2502)            |
|           |  File system owner user ID: WSOWNER (uid 2405)           |
|           |                                                          |
|           | The commands are placed into member BBOSBRAK of data set |
|           | USERID.TODAY.DATA.                                       |
|           |                                                          |
|           | Carefully review these definitions with your security    |
|           | administrator.                                           |
+-----------+----------------------------------------------------------+
| BBOSBRAK  | User ID requirement: RACF special authority.             |
+-----------+                                                          |
| Done:     | This job executes the RACF commands created by the       |
|           | previous job.  If the group and user IDs named above     |
|           | have already been created during a previous WebSphere    |
|           | for z/OS configuration and are in all target system RACF |
|           | database(s), you do not need to rerun this job.          |
|           |                                                          |
|           | RESULT: You may receive errors, such as INVALID USER     |
|           | messages, from this job because a user ID, group  or     |
|           | profile is already defined.  Make sure the existing      |
|           | user ID, group or profile has the same characteristics   |
|           | as the user ID, group or profile being created by        |
|           | BBOSBRAK.  If not, then change the values in the         |
|           | customization dialog which are causing the conflict,     |
|           | regenerate the customization jobs, and restart the       |
|           | process.                                                 |
|           |                                                          |
|           | When this step is complete, all groups and user IDs      |
|           | listed above for job BBOSBRAJ should be defined in the   |
|           | RACF database on each target system for the cell.        |
|           | Note: the WAS owner user ID WSOWNER MUST have the WAS    |
|           | configuration group WSCFG1 as its default OMVS group.    |
|           |                                                          |
| By:       |                                                          |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOSBRAM  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job creates home directories for WebSphere for z/OS |
|           | user IDS. These home directories will be subdirectories  |
|           | of /var/WebSphere/home                                   |
| By:       |                                                          |
|           | This job will:                                           |
|           |                                                          |
|           | Create the following directory with permission bits 755: |
|           |                                                          |
|           |  /var/WebSphere/home                                     |
|           |                                                          |
|           | Create the following directory with ownership            |
|           | WSOWNER:WSCFG1 and permission bits 770:                  |
|           |                                                          |
|           |  /var/WebSphere/home/WSCFG1                              |
|           |                                                          |
|           | Create the following directory with ownership            |
|           | WSOWNER:WSSR1 and permission bits 770:                   |
|           |                                                          |
|           |  /var/WebSphere/home/WSSR1                               |
|           |                                                          |
|           | Create the following directory with ownership            |
|           | WSOWNER:WSCLGP and permission bits 770:                  |
|           |                                                          |
|           |  /var/WebSphere/home/WSCLGP                              |
|           |                                                          |
|           | This job should be run on each z/OS system that will     |
|           | host WebSphere Application Server nodes using these      |
|           | WebSphere for z/OS common groups and owner user ID.      |
|           | After execution, verify that the directories have been   |
|           | created with the correct permissions on each system.     |
|           |                                                          |
|           | If these directories already exist with the specified    |
|           | ownership and permission on a target system, then this   |
|           | job does not need to be run on that system.              |
|           |                                                          |
|           | ATTENTION: If the directory                              |
|           |  /var/WebSphere/home                                     |
|           | is used by applications other than WebSphere Application |
|           | Server, make sure that the permissions set by            |
|           | BBOSBRAM (755) are appropriate, or change them manually. |
|           | This directory must be world-readable for Websphere      |
|           | Application Server to run correctly.                     |
+-----------+----------------------------------------------------------+
| BBOMSGC   | User ID requirement: Update authority for data set       |
+-----------+ SYS1.MSGENU and/or SYS1.MSGJPN.                          |
| Done:     |                                                          |
|           | ATTENTION: This is optional unless you require message   |
|           | translation.                                             |
| By:       |                                                          |
|           | This job sets up MMS to translate messages for WebSphere |
|           | for z/OS.                                                |
|           |                                                          |
|           | There are two steps to update SYS1.MSGENU and            |
|           | SYS1.MSGJPN. Remove the unneeded step and change the     |
|           | target libraries, if necessary.                          |
+-----------+----------------------------------------------------------+
| BBOCBRAJ  | User ID requirement: Authority to update data set        |
+-----------+                                                          |
| Done:     | USERID.TODAY.DATA                                        |
|           |                                                          |
|           | This job builds (but does not execute) the RACF commands |
| By:       | for the WebSphere for z/OS run-time clusters and places  |
|           | them into member BBOWBRAK of data set                    |
|           |                                                          |
|           | USERID.TODAY.DATA                                        |
|           |                                                          |
|           | Carefully review these definitions with your security    |
|           | administrator.                                           |
+-----------+----------------------------------------------------------+
| BBOCBRAK  | User ID requirement: RACF special authority.             |
+-----------+                                                          |
| Done:     | This job executes the RACF commands set up in the        |
|           | previous job.                                            |
|           |                                                          |
|           | This job creates the WebSphere administrator ID WSADMIN  |
|           | without a password.  You must assign this user ID a      |
|           | password that complies with your institution standards;  |
|           | this is also the password that will be used when logging |
|           | on to the WebSphere Application Server administrative    |
|           | console.                                                 |
|           |                                                          |
|           | Enter the following RACF command to assign a password:   |
|           |                                                          |
|           |   ALTUSER WSADMIN PASSWORD(password) NOEXPIRED           |
|           |                                                          |
|           | If you are using a different security system, make sure  |
|           | that the  user ID has a password.                        |
|           |                                                          |
|           |                                                          |
| By:       | RESULT: You may receive errors, such as INVALID USER     |
|           | messages, from this job because a user ID, group  or     |
|           | profile is already defined.  Make sure the existing      |
|           | user ID, group or profile has the same characteristics   |
|           | as the user ID, group or profile being created by        |
|           | BBOCBRAK.  If not, then change the values in the         |
|           | customization dialog which are causing the conflict,     |
|           | regenerate the customization jobs, and restart the       |
|           | process.                                                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | Check user ID authorizations.                            |
+-----------+                                                          |
| Done:     | Make sure the WSCFG1 group has read access to all        |
|           | WebSphere product data sets, as well as to any other     |
|           | data sets which will be placed in WebSphere for z/OS     |
|           | cataloged procedure STEPLIB concatenations.              |
|           |                                                          |
|           | Make sure the following user IDs have read access to     |
| By:       | the resolver configuration file in use on your system.   |
|           | Depending on your IP setup, this file may be             |
|           | /etc/resolv.conf, SYS1.TCPPARMS(TCPDATA), or another     |
|           | data set.                                                |
|           |                                                          |
|           | ASCR1                                                    |
|           | ASSR1                                                    |
|           |                                                          |
|           | See the z/OS eNetwork Communication Server IP            |
|           | Configuration manual for the resolver search order.      |
|           |                                                          |
|           | Ensure the following user ID has read access to the data |
|           | sets in your system parmlib concatenation:               |
|           |                                                          |
|           | WSDMNCR1                                                 |
|           |                                                          |
|           | ATTENTION:                                               |
|           |                                                          |
|           |  If operator commands are protected by the z/OS security |
|           |  server at your installation, you must ensure that       |
|           |  sufficient authority is given to WebSphere tasks to     |
|           |  control operations.                                     |
|           |                                                          |
|           |  The Application Server Controller user ID (ASCR1)       |
|           |  needs the ability to perform operations on started      |
|           |  tasks belonging to WebSphere Application Server for     |
|           |  z/OS.                                                   |
|           |                                                          |
|           |  The asynchronous administrator user ID, and any user    |
|           |  ID used to run the federation job when the node agent   |
|           |  is started automatically, need the authority to issue   |
|           |  the MVS START command.                                  |
|           |                                                          |
|           |  If you are currently controlling MVS console command    |
|           |  authority with SAF OPERCMDS profiles, grant the         |
|           |  following authorities as indicated, substituting your   |
|           |  own profile names:                                      |
|           |                                                          |
|           |  PERMIT  START_profile_name  CLASS(OPERCMDS)             |
|           |          ID (ASCR1  WSADMSH)  ACCESS(UPDATE)             |
|           |                                                          |
|           |  PERMIT  STOP_profile_name  CLASS(OPERCMDS)              |
|           |          ID (ASCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  MODIFY_profile_name  CLASS(OPERCMDS)            |
|           |          ID (ASCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  CANCEL_profile_name  CLASS(OPERCMDS)            |
|           |          ID (ASCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  FORCE_profile_name  CLASS(OPERCMDS)             |
|           |          ID (ASCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |                                                          |
|           |  You must also grant the appropriate console command     |
|           |  authority to any user ID that executes the              |
|           |  startServer.sh or stopServer.sh script.                 |
+-----------+----------------------------------------------------------+
| BBOWCHFS  | User ID requirement: UID=0 and authority to allocate     |
+-----------+                                                          |
| Done:     | OMVS.WAS.CONFIG.HFS                                      |
|           |                                                          |
|           | This job:                                                |
|           |                                                          |
| By:       | o   Creates a mount point directory                      |
|           |                                                          |
|           |     /WebSphere/V6R1                                      |
|           |                                                          |
|           | o   Allocates the configuration file system using        |
|           |     the Hierarchical File System (HFS)                   |
|           |                                                          |
|           |     OMVS.WAS.CONFIG.HFS                                  |
|           |                                                          |
|           |     and mounts it at the above mount point.              |
|           |                                                          |
|           | DO NOT RUN THIS JOB IF:                                  |
|           |   1. The configuration file system already exists and is |
|           |      mounted at the desired mountpoint, or if            |
|           |                                                          |
|           |   2. The mount point directory is controlled by          |
|           |      automount.  Either disable the automount rule for   |
|           |      the configuration mount point while running this    |
|           |      job, or perform the following steps manually:       |
|           |                                                          |
|           |      a. Allocate the configuration file system data set. |
|           |      b. Issue the following shell commands, which will   |
|           |         also cause automount to mount the file system    |
|           |                                                          |
|           |      chmod 775 /WebSphere/V6R1                           |
|           |      chown WSOWNER:WSCFG1 /WebSphere/V6R1                |
|           |                                                          |
|           | BEFORE YOU BEGIN: The BBOWCHFS job assumes your root     |
|           | file system is mounted in read/write mode.  If the root  |
|           | file system is not mounted in read/write mode, manually  |
|           | create the directory                                     |
|           |                                                          |
|           | /WebSphere/V6R1                                          |
|           |                                                          |
|           | and any needed higher directories, set file permissions  |
|           | to 775, and set the owning user ID and group to WSOWNER  |
|           | and WSCFG1 before running BBOWCHFS.                      |
|           |                                                          |
|           | EXAMPLE: If you plan to use /WebSphere/V6R0 as your      |
|           | directory, issue the following commands from within the  |
|           | OMVS shell:                                              |
|           |                                                          |
|           |   mkdir -p -m 775 /WebSphere/V6R0                        |
|           |   chown -R WSOWNER:WSCFG1 /WebSphere                     |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOWHFSA  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job populates the previously-created file system.   |
|           |                                                          |
|           | Upon completion, examine the job output. Success is      |
| By:       | indicated with a RC=0 in the job output.                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOWCPY1  | User ID requirement: update authority for:               |
+-----------+                                                          |
| Done:     |                                                          |
|           |                                                          |
|           |     SYS1.PROCLIB                                         |
| By:       |                                                          |
|           |                                                          |
|           |                                                          |
|           | This job copies the tailored start procedures,           |
|           | parameters, and EXECs to the run-time libraries.         |
|           |                                                          |
|           | ATTENTION: Be aware that you may overlay existing        |
|           | members in the above data sets.                          |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOWWPFA  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job sets up the runtime file system.                |
|           |                                                          |
|           |                                                          |
| By:       | Upon completion, examine the job output. Success is      |
|           | indicated by rc=0.                                       |
|           |                                                          |
|           | Note: If the BBOWWPFA (profile creation) job fails,      |
|           | delete the WAS_HOME/profiles/default directory and all   |
|           | its contents before rerunning BBOWWPFA.                  |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOWHFSB  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job will complete the file system initialization.   |
|           |                                                          |
|           | Upon completion, examine the job output. Success is      |
| By:       | indicated with a RC=0 in the job output.                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | All WebSphere Application Server processes require       |
+-----------+ access to the Language Environment and System SSL load   |
| Done:     | modules.                                                 |
|           |                                                          |
|           | If the SCEERUN, SCEERUN2 and System SSL load module      |
|           | libraries are not in the system link list or link pack   |
|           | area, add them to the STEPLIB DD concatenation in each   |
|           | of the following cataloged procedures in                 |
|           | SYS1.PROCLIB:                                            |
|           |                                                          |
|           |     BBO6ACRZ                                             |
|           |     BBO6ASRZ                                             |
|           |     BBO6CRAZ                                             |
|           |     BBO6DMNZ                                             |
|           |                                                          |
|           | and also add the full data set names, separated by     |
|           | colons (:), to the STEPLIB variable in the shell script |
|           |                                                         |
|           |     /WebSphere/V6R1/                                    |
|           |      AppServer/                                         |
|           |       profiles/default/bin/setupCmdLine.sh              |
| By:       |                                                         |
|           | When modifying the setupCmdLine.sh script, do not       |
|           | remove lines or comment them out, as this may cause     |
|           | problems with automated updates to the script.          |
|           |                                                         |
|           | Add only those data sets which are NOT in the link list |
|           | or link pack area.                                      |
|           |                                                         |
+-----------+---------------------------------------------------------+
| --------  | Make sure Resource Recovery Services (RRS) is active.   |
+-----------+ (See the InfoCenter for setup instructions if necessary.)|
| Done:     | Look for the following console message to verify that   |
|           | RRS was successfully started:                           |
|           |                                                         |
| By:       |                                                         |
|           |   ASA2011I RRS INITIALIZATION COMPLETE. COMPONENT       |
|           |     ID=SCRRS                                            |
|           |                                                         |
+-----------+---------------------------------------------------------+
| --------  | If your system is busy, you may want to include a rule  |
+-----------+ in your WLM policy that OMVS work for job BBOS001       |
| Done:     | (such as the postinstaller step) is to run in a service |
|           | class with a high service objective.                    |
| By:       |                                                         |
+-----------+---------------------------------------------------------+
| --------  | Start the Application Server                            |
+-----------+                                                         |
| Done:     | Issue the MVS command                                   |
|           |                                                         |
|           |   START BBO6ACR,JOBNAME=BBOS001,                        |
|           |         ENV=AQFT.AQFT.BBOS001                           |
|           |                                                         |
| By:       | This command starts the Application Server. Wait until  |
|           | the server is finished initializing before proceeding.  |
|           |                                                         |
|           | RESULT: The following message appears on the console and |
|           | in the job log of                                       |
|           |                                                         |
|           | BBOS001                                                 |
|           |                                                         |
|           |   BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR   |
|           |     z/OS CONTROL PROCESS BBOS001                        |
+-----------+---------------------------------------------------------+
| BBOWIVT   | User ID requirement: UID=0 or WSADMIN                   |
+-----------+                                                         |
| Done:     | This job runs the IVT application. See related topics in |
|           | the WebSphere for z/OS Information Center at            |
|           | http://www.ibm.com/software/webservers/appserv/zos_os390/|
|           | library/                                                |
|           |                                                         |
|           | for information about how to run this job.              |
| By:       |                                                         |
+-----------+---------------------------------------------------------+
| The product is now configured and verified.                         |
|                                                                     |
| To start the application server, issue the MVS command:             |
|                                                                     |
|   START BBO6ACR,JOBNAME=BBOS001,                                    |
|         ENV=AQFT.AQFT.BBOS001                                       |
|                                                                     |
| To stop the application server, enter the MVS command:              |
|                                                                   |
|   STOP BBODMNB                                                    |
+-------------------------------------------------------------------+

The following is a useful script that helps you define security
controls for clusters. It is in data set

'USERID.TODAY.DATA'

+-----------+-------------------------------------------------------+
| BBOWBRAC  | This is a sample exec you can modify to include       |
+-----------+ installation-specific RACF controls. This exec defines |
| Done:     | all the user IDs and groups that are necessary and    |
|           | sufficient for installing WebSphere for z/OS.         |
|           |                                                       |
| By:       | Additionally, there are commented sections for other  |
|           | components that might be used (SSL, for example).     |
+-----------+-------------------------------------------------------+
```

Working with your new server:

Once you complete the customization instructions, you will have a WebSphere Application Server for z/OS stand-alone application server. The application server includes a (simple) cell and node structure. This article provides useful information for when you work with your new server.

Before starting your server

Make sure that the WebSphere Application Server for z/OS product file system and configuration file system are mounted. If you chose to load the SBBOLPA (and possibly SBBOLOAD) into the system link pack area, make sure that these libraries are loaded into LPA before starting the server.

Starting the stand-alone application server

To start your stand-alone application server, issue the following MVS console command:

```
START server_proc,JOBNAME=server_name,ENV=cell_name.node_name.server_name
```

where:

- *server_proc* is the stand-alone application server controller cataloged procedure.
- *server_name* is the server short name.
- *node_name* is the node short name.
- *cell_name* is the cell short name.

If you chose default values and your system is named MVSA, for example, you would enter the following START command:

```
START BBO6ACR,JOBNAME=BBOS001,ENV=MVSA.MVSA.BBOS001
```

The START command brings up the controller. The controller starts the location service daemon, then uses WLM to start the servant. You should see a message like the following when the entire server is up and running:

```
BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR Z/OS CONTROL PROCESS BBOS001
```

Accessing the server administrative console

Once the server is successfully started, access the administrative console port by pointing a Web browser to the following URL:

```
http://hostname:http_port/ibm/console
```

where:

- *hostname* is the HTTP transport host name you specified during customization.

  Note: If you specified "*" for the HTTP host name, this is actually the node host name.
- *http_port* is the HTTP port you specified during customization.

  Note: The default Administrative console port for the stand-alone application server is 9060. This value has changed since Version 6.0.

Until global security is enabled, you will see a sign-on screen that asks you for a user ID but no password.

The user ID can be anything and is used only to provide basic tracking of changes. Be aware that, until you enable global security, anyone with a Web browser and access to the HTTP port can modify your application serving environment.

You can use the administrative console, scripting, or both to manage the application server and deploy and manage J2EE applications. See the *Using the administrative clients* PDF for more information.

Accessing the Samples Gallery

If you chose to install the sample applications, you can access them by pointing a Web browser to the following URL:

```
http://hostname:http_port/WSsamples/
```

where:

- *hostname* is the HTTP transport host name you specified during customization.

  Note: If you specified "*" for the HTTP host name, this is actually the node host name.
- *http_port* is the HTTP port that you specified during customization.

  Note: The default HTTP port for the stand-alone application server is 9080.

Stopping your stand-alone application server

The easiest way to stop the stand-alone application server is to stop the location service daemon. The location service daemon holds pointers to modules in common storage, and stopping it forces the rest of the cell to shut down. To stop the location service daemon, enter the following MVS console command:

```
STOP daemon_jobname
```

where `daemon_jobname` is the location service daemon jobname. The default location service daemon jobname for a stand-alone application server is BBODMNB.

Using the installation verification test:

You initially run the installation verification test (IVT), which verifies that WebSphere Application Server is configured correctly for your system, during ISPF customization of each of your systems. If you want to run the IVT at a time other than during initial customization, however, there are two methods from which you can choose.

Note: These options are now available when you are running a stand-alone application server configuration as well as after federating an application server.

Select either method to invoke the IVT:

- “Running the installation verification test with a job” on page 176

- “Running the installation verification test from a command line”

Running the installation verification test with a job:
The application server must be running.

Follow these steps to run the installation verification test (IVT) using the BBOWIVT job.

1. Verify that the application server is running.
2. Confirm that the ivtApp application is installed and started.
3. Submit the job BBOWIVT.

The IVT runs a series of verification tests and reports pass or fail status for each in the messages generated by the BBOWIVT job. The output is written to standard output and to the `WAS_HOME/profiles/default/logs/ivtClient.log` file.

Running the installation verification test from a command line:
The application server must be running.

Follow these steps to run the installation verification test (IVT) from a command line.

1. Verify that the application server is running.
2. Confirm that the ivtApp application is installed and started.
3. From a command line, navigate to the `WAS_HOME/bin` directory.
4. Issue the following command:

   ```
   ivt.sh server_name profile_name -p port_number [-host host_name]
   ```

   where

   - *server_name* is the short name of the server.
   - *profile_name* is the name of the profile.
   - `-p` *port_number* is an argument that specifies the port number.
   - `-host` *host_name* is an optional argument that specifies the host name. If you do not specify a host name, the program will use the host-name value that is set in your TCP/IP hosts file.

   Example:

   ```
   /WebSphere/V6R0/AppServer/bin> ivt.sh serverj default -p 9080 -host myhost
   ```

The IVT runs a series of verification tests and reports pass or fail status for each in the messages generated by the BBOWIVT job. The output is written to standard output and to the `WAS_HOME/profiles/default/logs/ivtClient.log` file.

Creating a Network Deployment cell

This article leads you through the tasks involved in setting up a WebSphere Application Server for z/OS Network Deployment environment.

Ensure that the security domain was successfully created on the z/OS target systems for the new Network Deployment cell. Have available a copy of the worksheet that you completed as a part of your planning for a Network Deployment cell.

Perform this task to set up a new WebSphere Application Server for z/OS Network Deployment cell. These steps will set up the cell and the deployment manager node.

1. Log on to TSO on the z/OS system on which you intend to configure the Network Deployment cell’s deployment manager. Use a user ID that has READ access to the WebSphere Application Server for z/OS product data sets. You will also need access to a user ID with authority to make security system updates and a user ID with UID 0. (These can all be the same user ID.)
2. Start the Customization Dialog. See “Starting the Customization Dialog” on page 133 for details.

3. Choose the configuration data sets in which you will store your customization jobs and data. See “Choosing configuration data sets” on page 178 for details.
4. Load the security domain variables saved from the security domain you intend to use for this cell. See “Configure common MVS groups and users” on page 178 for details.
5. Set the customization variables according to the values recorded on your Network Deployment cell worksheet. See “Setting the customization variables: Network Deployment cell” on page 179 for details.
6. Save the Network Deployment cell customization variables in a data set. See “Saving the cell variables” on page 183 for details. You will use these variables when creating managed nodes for the cell or federating stand-alone application servers into it.
7. Create the customization jobs and files, based on the customization variable values you entered. See “Creating the customization jobs and files” on page 183 for details.
8. Follow the generated customization instructions. See “Following the generated customization instructions: Network Deployment cell” on page 185 for details, and a sample set of customization instructions.

You are done when you have successfully completed the steps in the generated instructions. The new deployment manager is up and running on the chosen z/OS system. See “Working with your new deployment manager” on page 195 for more information.

Add application server nodes to your cell using one of two methods:

- Create a new managed node using the Customization Dialog and add application servers to it using the administrative console or scripting.
- Federate existing stand-alone application servers into your Network Deployment cell to create managed nodes with application servers.

Loading the security domain variables:

This article describes how to complete the ″Load security domain variables″ option for a WebSphere Application Server for z/OS Network Deployment cell.

Create the security domain you will use for the new Network Deployment cell and know the name of the saved security domain configuration variable file that you recorded on the security domain worksheet.

The security domain settings are used in the customization of every WebSphere Application Server for z/OS cell. By loading the security domain variables at the start of node or cell creation, you ensure that the security domain configuration you use is consistent and matches the RACF definitions that have already been set as part of security domain configuration.

Complete this task as the first step in configuring a new Network Deployment cell. If you encounter problems during customization and change the security domain variable values, be sure to re-save them.

1. On the ″Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″Load security domain variables″ and press **Enter**. You will see a panel that looks similar to the following:

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Load Security Domain Variables

 Specify the name of a data set containing the security domain variables,
 then press Enter to continue.

 IBM-supplied defaults are in ''


 Data set name:
 If this data set is not cataloged, specify the volume.

 Volume:
```

2. Fill in the name of the sequential data set you used to hold the security domain variable values and press **Enter**. The security domain variables will load.

The security domain settings are loaded. You can display these variables, but not change them.

Configure common MVS groups and users:

This article describes how to complete the ″Configure common MVS groups and users″ option for a WebSphere Application Server for z/OS Network Deployment cell.

Create the common MVS groups and users you will use for the new Network Deployment cell and know the name of the saved configuration variable file that you recorded on the worksheet.

The common MVS groups and users settings are used in the customization of every WebSphere Application Server for z/OS cell. By loading the variables at the start of node or cell creation, you ensure that the configuration you use is consistent and matches the RACF definitions that have already been set as part of the configuration.

Complete this task as the first step in configuring a new Network Deployment cell. If you encounter problems during customization and change the common MVS groups and users variable values, be sure to re-save them.

1. On the ″Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″Configure common MVS groups and users″ and press **Enter**. You will see a panel that looks similar to the following:
2. Fill in the name of the sequential data set you used to hold the common MVS group and user variable values and press **Enter**. The variables will load.

The common MVS groups and users settings are loaded. You can display these variables, but not change them.

Choosing configuration data sets:

This article leads you through the ″Allocate target data sets″ option in the Customization Dialog.

You must start the Customization Dialog and select the ″Create a Network Deployment cell″ option.

Each option in the Customization Dialog saves customization jobs and files in a pair of customization data sets. While is it possible to reuse these data sets, it is safest to create separate data sets for each WebSphere Application Server for z/OS configuration. The best practice is to use the customization data set name prefix (sometimes referred to as ″*config_hlq*″) to indicate the version and release of WebSphere Application Server for z/OS, the task you are performing, and the cell (and, in some cases, the node name) you are configuring.

For example, you might use the following data set name prefix for configuring a WebSphere Application Server for z/OS Network Deployment cell with cellname AZCELL:

```
SYSPROG.WAS602.AZCELL.NDCONFIG
```

Complete this task before generating the customization jobs and files.

1. On the main dialog panel, type the appropriate number in the *Option* field to select ″Allocate target data sets″.

2. Press Enter. **Result:** You see a panel that looks similar to the following:

```
-----------------     WebSphere Application Server for z/OS Customization      ------------------
Option  ===>

 Allocate Target Data Sets

 Specify a high level qualifier (HLQ) and press Enter to allocate the
 data sets to contain the generated jobs and instructions. You can
 specify multiple qualifiers (up to 39 characters).

 High level qualifier:                                        .CNTL
                                                              .DATA

 The Dialog will display data set allocation panels. You can make
 changes to the default allocations, however you should not change
 the DCB characteristics of the data sets.

    .CNTL  - a PDS with fixed block 80-byte records to
             contain customization jobs.

    .DATA  - a PDS with variable length data to contain
             other data produced by the Customization Dialog.
```

3. Fill in your chosen configuration data set name prefix value (*config_hlq*). If the data sets *config_hlq*.CNTL and *config_hlq*.DATA do not exist, you will be prompted for data set allocation information. If the data sets already exist, a message will inform you that they will be reused.

The data sets *config_hlq*.CNTL and *config_hlq*.DATA are allocated and will store customization jobs and files. These data set names will also be saved along with the customization variables.

Setting the customization variables: Network Deployment cell:

This article describes how to complete the ″Define variables″ option for a WebSphere Application Server for z/OS Network Deployment cell.

You must start the Customization Dialog and select the ″Create stand-alone Application Server nodes″ option. Have the Network Deployment cell Customization Dialog worksheet completed and at hand. A copy of this worksheet is available in the *Installing your application serving environment* PDF.

1. On the ″Create a Network Deployment Cell″ panel, type the appropriate number in the *Option* field to select ″Define variables″ and press **Enter**.
2. On the ″Define Variables to Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″System Locations (directories, HLQs, etc.)″ and press **Enter**.
3. Fill in the ″System Locations″ panels using the following screen shots as your guides. When you are done with each panel, press **Enter**.

   System Locations

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Locations (1 of 2)

   Specify the following for the system on which you are installing
   WebSphere Application Server for z/OS, then press Enter to continue.
   For some data sets, specify "Y" if they are in STEPLIB.

   System name.:  AQTS      Sysplex name :  MCLXCF01

 Full Names of Data Sets

   PROCLIB:  SYS1.PROCLIB
   PARMLIB:  SYS1.PARMLIB
   SYSEXEC:
```

```
   Run WebSphere Application Server from STEPLIB (Y/N)?  Y
   SBBOLPA.:  BOSS.VICOM.W000170.SBBOLPA
   SBBOLOAD:  BOSS.VICOM.W000170.SBBOLOAD
   SBBOLD2.:  BOSS.VICOM.W000170.SBBOLD2
   SBBOEXEC:  BOSS.VICOM.W000170.SBBOEXEC
   SBBOMSG.:  BOSS.VICOM.W000170.SBBOMSG

                                                       Use STEPLIB?
   SCEERUN.:  CEE.SCEERUN                                    N
   SCEERUN2:  CEE.SCEERUN2                                   N
   System SSL: GSK.SGSKLOAD                                  N
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Locations (2 of 2)

   Specify the following for your customization, then press Enter
   to continue.

 Locations of the HFS Resident Components

   WebSphere Application Server product directory:
     /usr/lpp/zWebSphere/V6R1
```

4. On the ″Define Variables to Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″System Environment Customization″ and press **Enter**.
5. Fill in the ″System Environment Customization″ panel using the following screen shot as your guide. When you are done, press **Enter**.

System Environment Customization

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

System Environment Customization (1 of 1)

   Specify the following to customize your system environment, then
   press Enter to continue.

 WebSphere Application Server for z/OS Configuration HFS Information

   Mount point....:  /WebSphere/V6R1
   Name..........:  OMVS.WAS.CONFIG.HFS
   Volume, or '*' for SMS.:  *
   Primary allocation in cylinders...:  250
   Secondary allocation in cylinders.:  100
```

6. On the ″Define Variables to Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″Server Customization″ and press **Enter**.
7. Fill in the ″Server Customization″ panels using the following screen shots as your guides. When you are done with each panel, press **Enter**.

Server Customization

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (1 of 4)

   Specify the following to customize your server, then press Enter
   to continue.

 Deployment Manager Definitions

   WebSphere Application Server home directory:
     /WebSphere/V6R1
         / DeploymentManager
   Cell name (short)......:  MCLXCF01
   Cell name (long).......:  MCLXCF01Network

   Node name (short)......:  MCLXCF01
   Node name (long).......:  MCLXCF01Manager

   Server name (short)....:  BBODMGR
   Server name (long).....:  dmgr

   Cluster transition name:  BBODMGR

   Add ability to administer Proxy Server(s) (Y/N) ?:  N
```

Note: The ″Add ability to administer Proxy Server(s) (Y/N)″ field was added to the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (2 of 4)

   Specify the following to customize your server, then press Enter
   to continue.

 Deployment Manager Definitions

   Controller Information

     Jobname.......:  BBODMGR
     Procedure name:  BBO6DCR
     User ID.......:  DMCR1
     UID...........:  2421

   Servant Information

     Jobname.......:  BBODMGRS
     Procedure name:  BBO6DSR
     User ID.......:  DMSR1
     UID...........:  2422
------------  WebSphere Application Server for z/OS Customization     -------
Option  ===>

Server Customization (3 of 4)

   Specify the following to customize your server, the press Enter
   to continue.

 Deployment Manager Definitions

   Node host name..........:

     SOAP JMX connector port...................:  8879
     Cell Discovery Address port...............:  7277
     DRS CLIENT Address port...................:  7989

   ORB Listener host name..:  *

     ORB port..................................:  9809
     ORB SSL port..............................:  0

   HTTP transport host name:  *

     HTTP port.................................:  9060
     HTTP SSL port.............................:  9043

 The High Availability Manager Host MUST resolve to a single
 IP address.  It cannot be a multihomed host.
```

```
   High Availability Manager Host:

   High Availability Manager Communication Port:  9352
```

Note: The ″High Availability Manager Host″ field was removed from the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (4 of 4)

   Specify the following to customize your server, then press Enter
   to continue.

 Location Service Daemon Definitions

   Daemon home directory:
     /WebSphere/V6R1/Daemon

   Daemon jobname:  BBODMNC

   Procedure name.:  BBO6DMN
   User ID........:  WSDMNCR1
   UID............:  2411

   IP name........:
   Port...........:  5755
   SSL port.......:  5756

   Register daemon with WLM DNS:  N
```

8. On the ″Define Variables to Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″Web Server Configuration″ and press **Enter**.
9. Fill in the ″Web Server Configuration″ panel using the following screen shot as your guide. When you are done with the panel, press **Enter**.

Web Server Configuration

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

 Security Customization (1 of 1)

  Specify a number and press Enter to customize security for
  WebSphere Application Server for z/OS.  You should review all of the
  variables in the section you select, even if you are using all of the
  IBM-supplied defaults.

  Press PF3 to return to the main menu.

  How do you want to manage user identities and authorization policy?

   The name used in defining the Web server in the admin console.

   1 - Use a z/OS security product.
       The z/OS security product manages users, groups, and
       the authorization policy.

   2 - Use WebSphere Application Server.
       WebSphere Application Server manages users, groups, and
       the authorization policy.

   3 - Do not enable security.
```

10. On the ″Define Variables to Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″View Security Customization Configuration Panels″ and press **Enter**.

Saving the cell variables:
You must start the Customization Dialog and fill in the variables for the chosen task.

Saving your Network Deployment cell variables allows you to load the same consistent set of values when configuring a new managed node for the cell.

Complete this task after setting the variables for your chosen task. If you encounter problems during customization and change the variable values, be sure to re-save them.

Note: This procedure applies to all dialog options except ″Configure a common group and user.″ For information about saving those variables, see “Saving the common group and user variables” on page 145

1. On the main panel for your chosen task, type ″S″ in the *Option* field to select ″Save customization variables″ and press **Enter**. You will see a panel that looks similar to the following:

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   Save Customization Variables

     Specify the name of a sequential data set to contain the
     customization variables, then press Enter to continue. If the
     data set does not exist, the dialog displays the Allocate New
     Data Set panel, with which you can allocate a data set.


    Data set name:
   ```

2. Fill in the name of the sequential data set you will use to hold the variable values. Choose a data set name that identifies the sysplex, cell or group of cells affected by your chosen task. Enclose the data set name in single quotes. If the data set does not exist, you will be prompted for data set allocation information.
3. Record the name of the data set on the applicable worksheet.

The settings are saved in the data set you selected.

Creating the customization jobs and files:
You must select configuration data sets to use and complete the process of defining variables for this task. See “Choosing configuration data sets” on page 178 and “Setting the customization variables: Network Deployment cell” on page 179 for more information.

The Customization Dialog creates customization batch jobs and data files, based on the variable values you specified in the dialog. The batch jobs and data sets will be written to the *config_hlq*.CNTL and *config_hlq*.DATA configuration data sets that you created with the ″Allocate target data sets″ option.

1. Ensure that the configuration data sets are allocated and not in use.

   Note: Editing a member in *config_hlq*.CNTL or *config_hlq*.DATA will cause this task to fail.
2. On the ″Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″Generate customization jobs″ and press **Enter**. You will have one of two results:
   - **Result A:** If all variables are defined correctly, you see the ″Specify Job Cards″ panel, which looks similar to this:

     ```
     ------------  WebSphere Application Server for z/OS Customization     --------
     Option  ===>

     Generate Customization Jobs
     ```

```
This portion of the Customization Dialog generates the jobs you must
run after you complete this Dialog process. You must complete the
customization process before you generate the jobs with this step.
If you have not done this, please return to that step.

Jobs and data files will get generated into data sets:
  'hlq.CNTL'
  'hlq.DATA'
If you wish to generate customization jobs using other data sets, then
exit from this panel and select the "Allocate target data sets" option.

All the jobs that will be tailored for you will need a job card.
Please enter a valid job card for your installation below. The
file tailoring process will update the jobname for you in all the
generated jobs, so you need not be concerned with that portion of
the job cards below. If continuations are needed, replace the
comment cards with continuations.

Specify the job cards, then press Enter to continue.

//jobname  JOB (ACCTNO,ROOM),'userid',CLASS=A,REGION=0M
//*
//*
//*
```

Note: Pay particular attention to the displayed target data sets. Make sure that they are the ones you intend to use.

- **Result B:** If the variables are not defined correctly, you will see the ″Verification″ panel. Decide whether the warnings or errors are serious enough to warrant returning to the ″Define variables″ option.

  Note: If the return code is 8 or greater, return to the ″Define variables″ option and fix the uncovered problems. If you saved the variables previously, be sure to re-save them after making any updates.

3. Fill in the job card information, according to your installation requirements. For each job, the dialog generates a jobname and the ″JOB″ keyword to match the member name of the PDS, but you specify the rest.

   Note: If you need to run these jobs on a particular system in the sysplex (for example, JES2 MAS or JES3 complex), you should specify the necessary Scheduling Environment (SCHENV), JES2 JOBPARM, or JES3 //*MAIN statement at this time.

   Example of a job card entry:

   ```
   //jobname JOB 1234,USER1,NOTIFY=????,MSGCLASS=O,REGION=0M
   //*            USER=SYSADM1,PASSWORD=SYSADM1
   /*JOBPARM SYSAFF=SYSB
   ```

   Note: This example is useful for jobs that require a user ID other than that of the logged-on TSO user. (This is typically a user ID with UID=0.) In that case, you can just put a comma at the end of the first line, put in the correct user ID on the second line, then uncomment that second line.

   You might want to use RACF SUBMIT authority to avoid having to keep passwords in your configuration data sets.

4. Fix any errors. If there are errors anywhere, you will see the ″Error″ panel. Press PF3 to exit the error panel, then enter the correct panel to fix the errors. Then return to the ″Generate Customization Jobs″ option and pick up where you left off. If necessary, you can update the variables and rerun this option. The generation process will delete and re-tailor all the members.

   Note: Compress the configuration data sets before you rerun this option.

You are done when all the jobs are generated. You can move ahead to viewing the generated jobs. See “Following the generated customization instructions: Network Deployment cell” for more information.

Following the generated customization instructions: Network Deployment cell:

You must generate the customization jobs and files for this task.

The Customization Dialog creates a set of instructions for each customization task. Follow these instructions to tailor and customize a Network Deployment cell on your system.

Note: Do not attempt to fix a typographical error or make a change by modifying the generated output. Many of the variables are used in multiple members of the target data sets; if you do not change them all, therefore, you will run into problems that are very difficult to diagnose.

1. On the ″Configure Deployment Manager Node″ panel, type the appropriate number in the *Option* field to select ″View instructions″ and press **Enter**. ISPF Browse will open and you will see the BBOCCINS member of *config_hlq*.CNTL.
2. Read the instructions carefully, both to preview the customization process and to find any typographical or other errors you might have made while entering the customization variable values.
3. Follow the instructions as given. There are two ways to follow the directions:
   - Follow the instructions while remaining in ISPF Browse.
   - Record the data set name and member at the top of the screen and either print the instructions or use ISPF split screen and browse or edit the instructions while you follow them.
4. Fix any problems. If you encounter problems caused by your Customization Dialog values, modify your variables using the dialog, regenerate the instructions, and restart the customization process.

   Note: Remember that you cannot generate new customization jobs while either configuration data set is open!

You are done with this customization task when you have successfully followed the generated instructions.

Sample generated instructions: Network Deployment cell:

This article presents a sample of what the Customization Dialog’s generated instructions might look like. This is a sample only--you must use the instructions generated from your own variables when configuring your system.

Note: This sample is based on the installation of WebSphere Application Server for z/OS Version 6.0.2. Generated instructions from different releases of Version 6 would have different content.

Instructions for customizing WebSphere for z/OS for a Deployment Manager node for 6.1.

```
-----------------------------------------------

The customization dialog has created jobs based on the information you
provided. These instructions tell you how to modify the operating
system and run the jobs to customize WebSphere for z/OS.

RULES:

1.  If you created the target data sets (*.CNTL and *.DATA) on another
    (driving) system, you must copy them to the target system and give
    them the same data set names.

2.  You must perform these instructions on your target system.

Doing manual configuration updates
---------------------------------
```

```
The customization dialog for WebSphere for z/OS does not attempt to
update configuration data for your base operating system or existing
subsystems. You must do the following manual steps prior to running
the WebSphere for z/OS configuration jobs.

Perform these steps to do manual configuration updates:

1.  Update BLSCUSER. Refer to member BBOIPCSP in

    'USERID.TODAY.CNTL'

    In order to use the IPCS support provided by the product, append
    the contents of this member to the BLSCUSER member in your IPCSPARM
    or system PARMLIB datasets.

    -------------------------------------------------------------------

2.  Update SCHEDxx. Refer to member BBOSCHED in

    'USERID.TODAY.CNTL'

    In order to set the correct program properties for the WebSphere
    for z/OS run-time executables, append the contents of this member
    to the SCHEDxx member in your system PARMLIB concatenation.

    Note: When you are finished, issue the command SET SCH=(xx,xx)
    to activate SCHEDxx and load a new program properties table.

    -------------------------------------------------------------------

3.  Make sure the following data sets are APF-authorized:

        DBEL.WASX.SBBOLPA
        DBEL.WASX.SBBOLOAD
        DBEL.WASX.SBBGLOAD
        DBEL.WASX.SBBOLD2

    Add these datasets to your PROGxx or IEAAPFxx parmlib members, as
    appropriate, ensuring you specify the correct volsers.

    -------------------------------------------------------------------

4.  Update your active BPXPRMxx member to have the following WebSphere
    for z/OS configuration file system:

    OMVS.WAS.CONFIG.HFS

    mounted at:

    /WebSphere/V6R1

    in read/write mode.

    EXAMPLE:

       MOUNT FILESYSTEM('OMVS.WAS.CONFIG.HFS')
         MOUNTPOINT('/WebSphere/V6R1')
          TYPE(HFS)
          MODE(RDWR)

    -------------------------------------------------------------------

5.  Update TCP/IP by reserving the following ports for WebSphere for
    z/OS:

       SOAP JMX Connector port                       - 8879
       CELL DISCOVERY ADDRESS port                   - 7277
       ORB port                                  - 9809
       Administrative console port               - 9060
       Administrative console secure port        - 9043

       High Availability Manager Communications port - 9352

       Daemon IP port                            - 5755
       Daemon SSL port                           - 5756

    View member BBOTCPID in

    'USERID.TODAY.CNTL'

    Add the contents of this member to the PORT section of the file
    referenced by the DD statement for the TCP/IP profile in the
    TCP/IP start procedure. Cut and paste from this member into the
    data set used by your installation.

    ATTENTION: If another application has already reserved any of these
    ports for its own use, you must resolve the resulting conflict
    before you continue. If you update the WebSphere for z/OS
    customization dialog with new port specifications, be sure to
    regenerate the customization jobs, data, and instuctions.

    -------------------------------------------------------------------

6.  The WebSphere product libraries will be placed in STEPLIB as
    needed, rather than in the system link pack area and system link
    list. Make sure that the target MVS system has at least 8MB
    of free storage in extended CSA for the daemon and for EACH
    node (deployment manager node or application server node).

    SBBOLOAD and SBBOLD2:
    ====================
    The following data sets will be placed in the STEPLIB concatenation
    for the location service daemon, controller and servant regions,
    and in the setupCmdLine.sh script in the WebSphere Configuration
    file system. You must not remove these STEPLIB statements.

    DBEL.WASX.SBBOLOAD
    DBEL.WASX.SBBOLD2


    BBORTS61:
    =========

    The BBORTS61 module is used by WebSphere Application Server for
    component trace support. A copy of this module (any maintenance
    level) must be in the system link pack area in order for CTRACE
    to work correctly.

    If a copy of BBORTS61 is currently loaded into LPA, you need take
    no further action.

    Otherwise, issue the following MVS console command to load BBORTS61
    into dynamic LPA:

      SETPROG LPA,ADD,MODNAME=BBORTS61,
              DSNAME=DBEL.WASX.SBBOLPA

    Alternatively, you can place the following statement in a parmlib
    PROGxx member which is activated with the SET PROG= command after
    system IPL is complete:

    LPA ADD MODNAME(BBORTS61)
      DSNAME(DBEL.WASX.SBBOLPA)
    Make sure that the BBORTS61 module is loaded into LPA after each
    system IPL.

    -------------------------------------------------------------------

7.  WebSphere for z/OS customization assumes that the following system
    data sets are in the system link list or link pack area:

    Language Environment     SCEERUN
                             SCEERUN2

    System SSL               SGSKLOAD (z/OS 1.5 and below)
                             SIEALNKE (z/OS 1.6 and above)

    Placing these data sets in the link list or link pack area improves
    performance and insulates your WebSphere for z/OS configuration
    from changes in data set names (for example, when migrating to z/OS
    1.6).

    If the Language Environment or System SSL load module libraries are
    not in your system link list or link pack area, you must perform
    the following steps before starting any WebSphere Application
    Server for z/OS servers:

    - Make sure the data sets are APF-authorized
    - Complete the optional step below to add the data sets to STEPLIB
      in the server JCL and setupCmdLine.sh script(s).

    If you regenerate server cataloged procedures at any point, make
    sure the data sets are added to the new cataloged procedures.

    -------------------------------------------------------------------

8.  If the error logstream
    WAS.ERROR.LOG
    does not already exist on your target system, make a copy of the
    appropriate job in the SBBOJCL data set, customize it according
    to the comments in the job, and run it:
       BBOERRLC    Create an error logstream in a coupling facility
       BBOERRLD    Create a DASD-only error logstream

    -------------------------------------------------------------------

9.  WebSphere for z/OS regions open a large number of files (more than
    1024). Make sure your BPXPRMxx parmlib member(s) specify a value of
    MAXFILEPROC that is greater than or equal to 2000. Use the
    following MVS console command to see your current MAXFILEPROC
    setting:

      D OMVS,OPTIONS


Running the customized jobs
---------------------------

The customization dialog built a number of batch jobs with the
variables you supplied. You must run the jobs in the order listed
below using user IDs with the appropriate authority.

BEFORE YOU BEGIN: Complete the section above entitled "Doing manual
configuration updates".

Follow the table below, which lists in order the jobs you must submit
and the commands you must enter. Special handling notes are included
in the table. All jobs are members of

USERID.TODAY.CNTL
```

```
Attention: After submitting each job, carefully check the output.
Errors may exist even when all return codes are zero.

+-----------+---------------------------------------------------------+
|           | The next three jobs (BBOSBRAJ, BBOSBRAK, BBOSBRAM)      |
|           | do not need to be run if the indicated groups, user IDs |
|           | and directories already exist with the correct gid,     |
|           | uid and ownership permission values, as given below.    |
+-----------+---------------------------------------------------------+
| BBOSBRAJ  | User ID requirement: Authority to update data set       |
+-----------+                                                         |
| Done:     | USERID.TODAY.DATA.                                      |
|           |                                                         |
|           | This job builds (but does not execute) the RACF commands|
| By:       | to create common WebSphere for z/OS groups and user IDs:|
|           |                                                         |
|           |  Configuration group:       WSCFG1 (gid 2500)           |
|           |  Servant group:             WSSR1 (gid 2501)            |
|           |  Local user group:          WSCLGP (gid 2502)           |
|           |  File system owner user ID: WSOWNER (uid 2405)          |
|           |                                                         |
|           | The commands are placed into member BBOSBRAK of data set|
|           | USERID.TODAY.DATA.                                      |
|           |                                                         |
|           | Carefully review these definitions with your security   |
|           | administrator.                                          |
+-----------+---------------------------------------------------------+
| BBOSBRAK  | User ID requirement: RACF special authority.            |
+-----------+                                                         |
| Done:     | This job executes the RACF commands created by the      |
|           | previous job.  If the group and user IDs named above    |
|           | have already been created during a previous WebSphere   |
|           | for z/OS configuration and are in all target system RACF|
|           | database(s), you do not need to rerun this job.         |
|           |                                                         |
|           | RESULT: You may receive errors, such as INVALID USER    |
|           | messages, from this job because a user ID, group  or    |
|           | profile is already defined.  Make sure the existing     |
|           | user ID, group or profile has the same characteristics  |
|           | as the user ID, group or profile being created by       |
|           | BBOSBRAK.  If not, then change the values in the        |
|           | customization dialog which are causing the conflict,    |
|           | regenerate the customization jobs, and restart the      |
|           | process.                                                |
|           |                                                         |
|           | When this step is complete, all groups and user IDs     |
|           | listed above for job BBOSBRAJ should be defined in the  |
|           | RACF database on each target system for the cell.       |
|           | Note: the WAS owner user ID WSOWNER MUST have the WAS   |
|           | configuration group WSCFG1 as its default OMVS group.   |
|           |                                                         |
| By:       |                                                         |
|           |                                                         |
+-----------+---------------------------------------------------------+
| BBOSBRAM  | User ID requirement: UID=0.                             |
+-----------+                                                         |
| Done:     | This job creates home directories for WebSphere for z/OS|
|           | user IDS. These home directories will be subdirectories |
|           | of /var/WebSphere/home                                  |
| By:       |                                                         |
|           | This job will:                                          |
|           |                                                         |
|           | Create the following directory with permission bits 755:|
|           |                                                         |
|           |  /var/WebSphere/home                                    |
|           |                                                         |
|           | Create the following directory with ownership              |
|           | WSOWNER:WSCFG1 and permission bits 770:                    |
|           |                                                            |
|           |  /var/WebSphere/home/WSCFG1                                |
|           |                                                            |
|           | Create the following directory with ownership              |
|           | WSOWNER:WSSR1 and permission bits 770:                     |
|           |                                                            |
|           |  /var/WebSphere/home/WSSR1                                 |
|           |                                                            |
|           | Create the following directory with ownership              |
|           | WSOWNER:WSCLGP and permission bits 770:                    |
|           |                                                            |
|           |  /var/WebSphere/home/WSCLGP                                |
|           |                                                            |
|           | This job should be run on each z/OS system that will       |
|           | host WebSphere Application Server nodes using these        |
|           | WebSphere for z/OS common groups and owner user ID.        |
|           | After execution, verify that the directories have been     |
|           | created with the correct permissions on each system.       |
|           |                                                            |
|           | If these directories already exist with the specified      |
|           | ownership and permission on a target system, then this     |
|           | job does not need to be run on that system.                |
|           |                                                            |
|           | ATTENTION: If the directory                                |
|           |  /var/WebSphere/home                                       |
|           | is used by applications other than WebSphere Application   |
|           | Server, make sure that the permissions set by              |
|           | BBOSBRAM (755) are appropriate, or change them manually.   |
|           | This directory must be world-readable for Websphere        |
|           | Application Server to run correctly.                       |
+-----------+------------------------------------------------------------+
| BBODBRAJ  | User ID requirement: Authority to update data set          |
+-----------+                                                            |
| Done:     | USERID.TODAY.DATA                                          |
|           |                                                            |
|           | This job builds (but does not execute) the RACF commands   |
|           | for the WebSphere for z/OS run-time clusters and places    |
| By:       | them into member BBODBRAK of data set                      |
|           |                                                            |
|           | USERID.TODAY.DATA                                          |
|           |                                                            |
|           | Carefully review these definitions with your security      |
|           | administrator.                                             |
+-----------+------------------------------------------------------------+
| BBODBRAK  | User ID requirement: RACF special authority.               |
+-----------+                                                            |
| Done:     | This job executes the RACF commands set up in the          |
|           | previous job.                                              |
|           |                                                            |
|           | This job creates the WebSphere administrator ID WSADMIN    |
|           | without a password.  You must assign this user ID a        |
|           | password that complies with your institution standards;    |
|           | this is also the password that will be used when logging   |
|           | on to the WebSphere Application Server administrative      |
|           | console.                                                   |
|           |                                                            |
|           | Enter the following RACF command to assign a password:     |
|           |                                                            |
|           |   ALTUSER WSADMIN PASSWORD(password) NOEXPIRED             |
|           |                                                            |
|           | If you are using a different security system, make sure    |
|           | that the WSADMIN user ID has a password.                   |
|           |                                                            |
|           |                                                            |
| By:       | RESULT: You may receive errors, such as INVALID USER       |
|           | messages, from this job because a user ID, group  or     |
|           | profile is already defined.  Make sure the existing      |
|           | user ID, group or profile has the same characteristics   |
|           | as the user ID, group or profile being created by        |
|           | BBODBRAK.  If not, then change the values in the         |
|           | customization dialog which are causing the conflict,     |
|           | regenerate the customization jobs, and restart the       |
|           | process.                                                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | Check user ID authorizations.                            |
+-----------+                                                          |
| Done:     | Make sure the WSCFG1 group has read access to all        |
|           | WebSphere product data sets, as well as to any other     |
|           | data sets which will be placed in WebSphere for z/OS     |
|           | cataloged procedure STEPLIB concatenations.              |
|           |                                                          |
|           | Make sure the following user IDs have read access to     |
| By:       | the resolver configuration file in use on your system.   |
|           | Depending on your IP setup, this file may be             |
|           | /etc/resolv.conf, SYS1.TCPPARMS(TCPDATA), or another     |
|           | data set.                                                |
|           |                                                          |
|           | DMCR1                                                    |
|           | DMSR1                                                    |
|           |                                                          |
|           | See the z/OS eNetwork Communication Server IP            |
|           | Configuration manual for the resolver search order.      |
|           |                                                          |
|           | Ensure the following user ID has read access to the data |
|           | sets in your system parmlib concatenation:               |
|           |                                                          |
|           | WSDMNCR1                                                 |
|           |                                                          |
|           | ATTENTION:                                               |
|           |                                                          |
|           |  If operator commands are protected by the z/OS security |
|           |  server at your installation, you must ensure that       |
|           |  sufficient authority is given to WebSphere tasks to     |
|           |  control operations.                                     |
|           |                                                          |
|           |  The Deployment Manager controller user ID (DMCR1)       |
|           |  needs the ability to perform operations on started      |
|           |  tasks belonging to WebSphere Application Server for     |
|           |  z/OS.                                                   |
|           |                                                          |
|           |  The asynchronous administrator user ID, and any user    |
|           |  ID used to run the federation job when the node agent   |
|           |  is started automatically, need the authority to issue   |
|           |  the MVS START command.                                  |
|           |                                                          |
|           |  If you are currently controlling MVS console command    |
|           |  authority with SAF OPERCMDS profiles, grant the         |
|           |  following authorities as indicated, substituting your   |
|           |  own profile names:                                      |
|           |                                                          |
|           |  PERMIT  START_profile_name  CLASS(OPERCMDS)             |
|           |          ID (DMCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  STOP_profile_name  CLASS(OPERCMDS)              |
|           |          ID (DMCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  MODIFY_profile_name  CLASS(OPERCMDS)            |
|           |          ID (DMCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  CANCEL_profile_name  CLASS(OPERCMDS)            |
|           |          ID (DMCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  FORCE_profile_name  CLASS(OPERCMDS)             |
|           |          ID (DMCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  You must also grant the appropriate console command    |
|           |  authority to any user ID that executes the             |
|           |  startServer.sh or stopServer.sh script.                |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBODCHFS  | User ID requirement: UID=0 and authority to allocate     |
+-----------+                                                          |
| Done:     | OMVS.WAS.CONFIG.HFS                                      |
|           |                                                          |
|           | ATTENTION: Skip this step if the mount point is already  |
|           | created, such as for the stand-alone Application Server. |
|           |                                                          |
| By:       | This job:                                                |
|           |                                                          |
|           | o   Creates a mount point directory                      |
|           |                                                          |
|           |     /WebSphere/V6R1                                      |
|           |                                                          |
|           | o   Allocates the configuration file system using        |
|           |     the Hierarchical File System (HFS)                   |
|           |                                                          |
|           |     OMVS.WAS.CONFIG.HFS                                  |
|           |                                                          |
|           |     and mounts it at the above mount point.              |
|           |                                                          |
|           | DO NOT RUN THIS JOB IF:                                  |
|           |   1. The configuration file system already exists and is |
|           |      mounted at the desired mountpoint, or if            |
|           |                                                          |
|           |   2. The mount point directory is controlled by          |
|           |      automount.  Either disable the automount rule for   |
|           |      the configuration mount point while running this    |
|           |      job, or perform the following steps manually:       |
|           |                                                          |
|           |      a. Allocate the configuration file system data set. |
|           |      b. Issue the following shell commands, which will   |
|           |         also cause automount to mount the file system    |
|           |                                                          |
|           |      chmod 775 /WebSphere/V6R1                           |
|           |      chown WSOWNER:WSCFG1 /WebSphere/V6R1                |
|           |                                                          |
|           |                                                          |
|           | BEFORE YOU BEGIN: The BBODCHFS job assumes your root     |
|           | file system is mounted in read/write mode.  If the root  |
|           | file system is not mounted in read/write mode, manually  |
|           | create the directory                                     |
|           |                                                          |
|           | /WebSphere/V6R1                                          |
|           |                                                          |
|           | and any needed higher directories. Set file permissions  |
|           | to 775 and set the owning user ID and group to WSOWNER   |
|           | and WSCFG1 before running BBODCHFS.                      |
|           |                                                          |
|           | EXAMPLE: If you plan to use /WebSphere/V6R0 as your      |
|           | directory, issue the following commands from within the  |
|           | OMVS shell:                                              |
|           |                                                          |
|           |   mkdir -p -m 775 /WebSphere/V6R0                        |
|           |   chown -R WSOWNER:WSCFG1 /WebSphere                     |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBODHFSA  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job populates the previously-created file system.   |
|           |                                                          |
|           | Upon completion, examine the job output. Success is      |
| By:       | indicated with a RC=0 in the job output.                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBODCPY1  | User ID requirement: update authority for:               |
+-----------+                                                          |
| Done:     |                                                          |
|           |                                                          |
|           |     SYS1.PROCLIB                                         |
| By:       |                                                          |
|           |                                                          |
|           | This job copies the tailored start procedures,           |
|           | parameters, and EXECs to the run-time product libraries. |
|           |                                                          |
|           | ATTENTION: Be aware that you may overlay existing        |
|           | members in the above data set.                           |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOWWPFD  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     |  This job sets up the runtime file system.               |
|           |                                                          |
|           |                                                          |
| By:       | Upon completion, examine the job output. Success is      |
|           | indicated by rc=0.                                       |
|           |                                                          |
|           | Note: If the BBOWWPFD (profile creation) job fails,      |
|           | delete the WAS_HOME/profiles/default directory and all   |
|           | its contents before rerunning BBOWWPFD.                  |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBODHFSB  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job will complete the file system initialization.   |
|           |                                                          |
|           | Upon completion, examine the job output. Success is      |
| By:       | indicated with a RC=0 in the job output.                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | All WebSphere Application Server processes require       |
+-----------+ access to the Language Environment and System SSL load   |
| Done:     | modules.                                                 |
|           |                                                          |
|           | If the SCEERUN, SCEERUN2 and System SSL load module      |
|           | libraries are not in the system link list or link pack   |
|           | area, add them to the STEPLIB DD concatenation in each   |
|           | of the following cataloged procedures in                 |
|           | SYS1.PROCLIB:                                            |
|           |                                                          |
|           |     BBO6DCRZ                                             |
|           |     BBO6DSRZ                                             |
|           |     BBO6DMNZ                                             |
|           |                                                          |
|           | and also add the full data set names, separated by       |
|           | colons (:), to the STEPLIB variable in the shell script  |
|           |                                                          |
|           |     /WebSphere/V6R1/                                     |
|           |      DeploymentManager/                                  |
|           |       profiles/default/bin/setupCmdLine.sh               |
| By:       |                                                          |
|           | When modifying the setupCmdLine.sh script, do not        |
|           | remove lines or comment them out, as this may cause      |
|           | problems with automated updates to the script.           |
|           |                                                          |
|           | Add only those data sets which are NOT in the link list  |
|           | or link pack area.                                        |
|           |                                                           |
+-----------+-----------------------------------------------------------+
| --------  | Make sure Resource Recovery Services (RRS) is active.     |
+-----------+ (See the InfoCenter for setup instructions if necessary.) |
| Done:     | Look for the following console message to verify that     |
|           | RRS was successfully started:                             |
|           |                                                           |
| By:       |                                                           |
|           |   ASA2011I RRS INITIALIZATION COMPLETE. COMPONENT         |
|           |     ID=SCRRS                                              |
|           |                                                           |
+-----------+-----------------------------------------------------------+
| --------  | If your system is busy, you may want to include a rule    |
+-----------+ in your WLM policy that OMVS work for job BBODMGR         |
| Done:     | (such as the postinstaller step) is to run in a service   |
|           | class with a high service objective.                      |
| By:       |                                                           |
+-----------+-----------------------------------------------------------+
| --------  | Start the Deployment Manager.                             |
+-----------+                                                           |
| Done:     | Issue the MVS command                                     |
|           |                                                           |
|           |   START BBO6DCR,JOBNAME=BBODMGR,                          |
|           |         ENV=MCLXCF01.MCLXCF01.BBODMGR                     |
|           |                                                           |
| By:       | This command starts the Deployment Manager and also       |
|           | starts the location service daemon. Wait until the        |
|           | server is finished initializing before proceeding.        |
|           |                                                           |
|           | RESULT: The following message appears on the console and  |
|           | in the job log of BBODMGR                                 |
|           |                                                           |
|           |   BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR     |
|           |     z/OS CONTROL PROCESS BBODMGR                          |
|           |                                                           |
+-----------------------------------------------------------------------+
| The product is now configured for a Deployment Manager.               |
|                                                                       |
+-----------+-----------------------------------------------------------+

Note: At this point of the configuration, if you want to federate a
managed node or stand-alone Application Server node, you will need to
do the following steps:

Choose option 3  Create Network Deployment cells and nodes, from the
WebSphere Application Server for z/OS Customization main menu

Choose option 3  Federate an existing stand-alone application server
node into an existing Network Deployment cell

+-----------+-----------------------------------------------------------+
| To start the Deployment Manager, issue the following MVS command:     |
|                                                                       |
|   START BBO6DCR,JOBNAME=BBODMGR,                                      |
|         ENV=MCLXCF01.MCLXCF01.BBODMGR                                 |
|                                                                       |
| To stop the WebSphere for z/OS servers, enter the MVS command:        |
|                                                                       |
|   STOP BBODMNC                                                        |
|                                                                       |
+-----------------------------------------------------------------------+

The following is a useful script that helps you define security
controls. It is in data set

'USERID.TODAY.DATA'
```

```
+-----------+-------------------------------------------------------+
| BBODBRAC  | This is a sample exec that you may modify to include  |
+-----------+ installation-specific RACF controls. This exec defines |
| Done:     | all the user IDs and groups that are necessary and    |
|           | sufficient for installing WebSphere for z/OS.         |
|           |                                                       |
| By:       | Additionally, there are commented sections for other  |
|           | components that might be used (SSL, for example).     |
+-----------+-------------------------------------------------------+
```

Working with your new deployment manager:

Once you complete the customization instructions, you will have a WebSphere Application Server for z/OS Network Deployment cell. The Network Deployment cell consists of a deployment manager and a location service daemon. (To run J2EE applications, you must add application server nodes. See below for details.) This article provides useful information for working with your new Network Deployment cell.

Before starting your server

Make sure that the WebSphere Application Server for z/OS product HFS and configuration HFS are mounted. If you chose to load the SBBOLPA (and possibly SBBOLOAD) into the system link pack area, make sure that these libraries are loaded into LPA before starting the server.

Starting the deployment manager

To start your deployment manager, issue the following MVS console command:

```
START server_proc,JOBNAME=dmgr_name,ENV=cell_name.node_name.dmgr_name
```

where:

- *server_proc* is the deployment manager controller cataloged procedure.
- *dmgr_name* is the deployment manager short name.
- *node_name* is the deployment manager node short name.
- *cell_name* is the cell short name.

If you chose default values and your system is named MVSA, for example, you would enter the following START command:

```
START BBO6DCR,JOBNAME=BBODMGR,ENV=MVSA.MVSA.BBODMGR
```

The START command brings up the deployment manager controller. The controller starts the location service daemon, then uses WLM to start the deployment manager servant. You should see a message like the following when the deployment manager is up and running:

```
BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR Z/OS CONTROL PROCESS BBODMGR
```

Accessing the server administrative console

Once the deployment manager is successfully started, access the administrative console by pointing a Web browser to the following URL:

```
http://hostname:http_port/ibm/console
```

where:

- *hostname* is the deployment manager HTTP transport host name you specified during customization.

Note: If you specified ″*″ for the deployment manager HTTP host name, this is actually the deployment manager node host name.

- *http_port* is the deployment manager HTTP port you specified during customization.

  Note: The default HTTP port for the deployment manager is 9060.

Until global security is enabled, you will see a signon screen that asks you for a user ID but no password.

The user ID can be anything and is used only to provide basic tracking of changes. Be aware that, until you enable global security, anyone with a Web browser and access to the HTTP port can modify your application serving environment.

You can use the administrative console, scripting, or both to manage the Network Deployment cell and deploy and manage J2EE applications. See the *Using the administrative clients* PDF for more information. Before you can deploy applications, however, you need to add application server nodes to your Network Deployment cell.

Adding application server nodes

Application server nodes (also called managed nodes) in a Network Deployment cell consist of a node agent and any number of application servers per node.

Note: Each z/OS system also needs one location service daemon for each stand-alone or Network Deployment cell hosted on the system.

Add an application server node to a Network Deployment cell using one of two methods:

- Create an (empty) managed node using the Customization Dialog. The new node can reside on the same or a different z/OS system as the deployment manager. The new managed node, consisting of just a node agent and perhaps a location service daemon, is federated into the Network Deployment cell. Once this is done, you can use the administrative console or scripting to add application servers and deploy and manage J2EE applications in the node.

  See the section ″Planning for a new managed node in a Network Deployment cell″ in the *Installing your application serving environment* PDF for more information.
- Federate an existing stand-alone application server into the Network Deployment cell. The stand-alone server node becomes a managed node in the Network Deployment cell, along with any J2EE applications that have been deployed on it.

  See the section ″Planning to federate a stand-alone server into a Network Deployment cell″ in the *Installing your application serving environment* PDF for more information.

Stopping your deployment manager

Use one of the following two methods to stop your deployment manager:

- Stop the location service daemon, which also stops the deployment manager and any of the cell’s managed nodes on the same z/OS system. The location service daemon holds pointers to modules in common storage, and stopping it forces the cell’s nodes on the same z/OS system as the location service daemon to shut down. To stop the location service daemon, enter the following MVS console command:

  ```
  STOP daemon_jobname
  ```

  where *daemon_jobname* is the location service daemon jobname. The default location service daemon jobname for a Network Deployment cell is BBODMNC.

  Note: This is the easiest way to stop the deployment manager.

- Stop just the deployment manager, leaving the location service daemon and any managed nodes on the z/OS system still running. This works because the deployment manager is used to administer only the cell--it does not need to be up for J2EE applications in the cell to run. To stop the deployment manager, enter the following MVS console command:

  ```
  STOP dmgr_name
  ```

  where *dmgr_name* is the deployment manager short name. The default deployment manager short name is BBODMGR.

Creating a managed server node

This article leads you through the tasks involved in creating a WebSphere Application Server for z/OS managed server node.

Perform this task to create a new WebSphere Application Server for z/OS managed node.

1. Log on to TSO on the z/OS system on which you intend to configure the managed node. Use a user ID that has READ access to the WebSphere Application Server for z/OS product data sets. You will also need access to a user ID with authority to make security system updates and a user ID with UID 0. (These can all be the same user ID.)
2. Start the Customization Dialog. See “Starting the Customization Dialog” on page 133 for details.
3. Choose the configuration data sets in which you will store your customization jobs and data. See “Choosing configuration data sets” on page 199 for details.
4. Load the security domain variables saved from the security domain you intend to use for this cell. See “Loading the common MVS groups and users variables” on page 198 for details. (Optionally, you can use the L command to load the Network Deployment cell variables if you saved them during Network Deployment cell setup. Network Deployment cell variables include the security domain variables--you do not need to load both.)
5. Set the customization variables according to the values recorded on your managed node worksheet. See “Setting the customization variables: Managed node” on page 199 for details.
6. (Optional but recommended.) Save the managed node customization variables in a data set. See “Saving the cell variables” on page 183 for details.
7. Create the customization jobs and files, based on the customization variable values you entered. See “Creating the customization jobs and files” on page 202 for details.
8. Follow the generated customization instructions. See “Following the generated customization instructions: Managed node” on page 204 for details, and a sample set of customization instructions.

You are done when you have successfully completed the steps in the generated instructions. The new managed node is up and running on the chosen z/OS system. See “Working with your new managed server node” on page 213 for more information.

Loading the security domain variables:

This article describes how to complete the ″Load security domain variables″ option for a WebSphere Application Server for z/OS managed node.

Create the security domain you will use for the new managed node and know the name of the saved security domain configuration variable file that you recorded on the security domain worksheet.

The security domain settings are used in the customization of every WebSphere Application Server for z/OS cell. By loading the security domain variables at the start of node or cell creation, you ensure that the security domain configuration you use is consistent and matches the RACF definitions that have already been set as part of security domain configuration.

Complete this task as the first step in configuring a new managed node. If you encounter problems during customization and change the security domain variable values, be sure to re-save them.

1. On the ″Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″Load security domain variables″ and press **Enter**. You will see a panel that looks similar to the following:

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   Load Security Domain Variables

    Specify the name of a data set containing the security domain variables,
    then press Enter to continue.

    IBM-supplied defaults are in ''


    Data set name:


    If this data set is not cataloged, specify the volume.

    Volume:
   ```
2. Fill in the name of the sequential data set you used to hold the security domain variable values and press **Enter**. The security domain variables will load.

The security domain settings are loaded. You can display these variables, but not change them.

Loading the common MVS groups and users variables:

This article describes how to complete the ″Load common MVS groups and users variables″ option for a WebSphere Application Server for z/OS managed node.

Create the common MVS groups and users you will use for the new managed node and know the name of the saved common MVS groups and users configuration variable file that you recorded on the common MVS group and user worksheet.

The common MVS groups and users settings are used in the customization of every WebSphere Application Server for z/OS cell. By loading the common MVS groups and users variables at the start of node or cell creation, you ensure that the configuration you use is consistent and matches the RACF definitions that have already been set as part of configuration.

Complete this task as the first step in configuring a new managed node. If you encounter problems during customization and change the common MVS groups and users values, be sure to re-save them.

1. On the ″Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″Common MVS groups and users variables″ and press **Enter**. You will see a panel that looks similar to the following:

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   Load Common MVS groups and users

    Specify the name of a data set containing the common MVS groups and users
    variables, then press Enter to continue.

    IBM-supplied defaults are in ''


    Data set name:


    If this data set is not cataloged, specify the volume.

    Volume:
   ```

2. Fill in the name of the sequential data set you used to hold the common MVS groups and users variable values and press **Enter**. The variables will load.

The common MVS groups and users settings are loaded. You can display these variables, but not change them.

Choosing configuration data sets:

This article leads you through the ″Allocate target data sets″ option in the Customization Dialog.

You must start the Customization Dialog and select the ″Create an empty managed node and add it to an existing Network Deployment cell″ option.

Each option in the Customization Dialog saves customization jobs and files in a pair of customization data sets. While is it possible to reuse these data sets, it is safest to create separate data sets for each WebSphere Application Server for z/OS configuration. The best practice is to use the customization data set name prefix (sometimes referred to as ″*config_hlq*″) to indicate the version and release of WebSphere Application Server for z/OS, the task you are performing, and the cell (and, in some cases, the node name) you are configuring. For example, you might use the following data set name prefix for configuring a WebSphere Application Server for z/OS managed node named MA6N01 for cell MAINT1:

JULIA.WASV6R2.MAINT1.MA6N01.MANAGED

Complete this task before generating the customization jobs and files.

1. On the main dialog panel, type the appropriate number in the *Option* field to select ″Allocate target data sets″.
2. Press Enter. **Result:** You see a panel that looks similar to the following:

   ```
   -----------------     WebSphere Application Server for z/OS Customization      ------------------
   Option  ===>

    Allocate Target Data Sets

    Specify a high level qualifier (HLQ) and press Enter to allocate the
    data sets to contain the generated jobs and instructions. You can
    specify multiple qualifiers (up to 39 characters).

    High level qualifier:                                         .CNTL
                                                                  .DATA

    The Dialog will display data set allocation panels. You can make
    changes to the default allocations, however you should not change
    the DCB characteristics of the data sets.

       .CNTL  - a PDS with fixed block 80-byte records to
                contain customization jobs.

       .DATA  - a PDS with variable length data to contain
                other data produced by the Customization Dialog.
   ```
3. Fill in your chosen configuration data set name prefix value (*config_hlq*). If the data sets *config_hlq*.CNTL and *config_hlq*.DATA do not exist, you will be prompted for data set allocation information. If the data sets already exist, a message will inform you that they will be reused.

The data sets *config_hlq*.CNTL and *config_hlq*.DATA are allocated and will store customization jobs and files. These data set names will also be saved along with the customization variables.

Setting the customization variables: Managed node:

This article describes how to complete the ″Define variables″ option for a WebSphere Application Server for z/OS managed node.

You must start the Customization Dialog and select the ″Create stand-alone Application Server nodes″ option. Have the Managed node Customization Dialog worksheet completed and at hand. A copy of this worksheet is available in the *Installing your application serving environment* PDF.

1. On the ″Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″Define variables″ and press **Enter**.
2. On the ″Define Variables to Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″System Locations (directories, HLQs, etc.)″ and press **Enter**.
3. Fill in the System Locations panels using the following screen shots as your guides. When you are done with each panel, press **Enter**.

   System Locations

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   System Locations (1 of 2)

      Specify the following for the system on which you are installing
      WebSphere Application Server for z/OS, then press Enter to continue.
      For some data sets, specify "Y" if they are in STEPLIB.

      System name.:  AQTS      Sysplex name :  MCLXCF01

    Full Names of Data Sets

      PROCLIB:  SYS1.PROCLIB
      PARMLIB:  SYS1.PARMLIB
      SYSEXEC:

      Run WebSphere Application Server from STEPLIB (Y/N)?  Y
      SBBOLPA.:  BOSS.VICOM.W000170.SBBOLPA
      SBBOLOAD:  BOSS.VICOM.W000170.SBBOLOAD
      SBBOLD2.:  BOSS.VICOM.W000170.SBBOLD2
      SBBOEXEC:  BOSS.VICOM.W000170.SBBOEXEC
      SBBOMSG.:  BOSS.VICOM.W000170.SBBOMSG

                                                          Use STEPLIB?
      SCEERUN.:  CEE.SCEERUN                                    N
      SCEERUN2:  CEE.SCEERUN2                                   N
      System SSL: GSK.SGSKLOAD                                  N
   ```

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   System Locations (2 of 2)

      Specify the following for your customization, then press Enter
      to continue.

    Locations of HFS Resident Components

      WebSphere Application Server product directory:
        /usr/lpp/zWebSphere/V6R0
   ```

4. On the ″Define Variables to Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″System Environment Customization″ and press **Enter**.
5. Fill in the ″System Environment Customization″ panel using the following screen shot as your guide. When you are done, press **Enter**.

   System Environment Customization

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   System Environment Customization (1 of 1)

      Specify the following to customize your system environment, then
   ```

```
   press Enter to continue.

 WebSphere Application Server for z/OS Configuration HFS Information

   Mount point....:  /WebSphere/V6R0
   Name..........:  OMVS.WAS.CONFIG.HFS
   Volume, or '*' for SMS.:  *
   Primary allocation in cylinders...:  250
   Secondary allocation in cylinders.:  100
```

6. On the ″Define Variables to Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″Server Customization″ and press **Enter**.
7. Fill in the ″Server Customization″ panels using the following screen shots as your guides. When you are done with each panel, press **Enter**.

   Server Customization

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (1 of 4)

   Specify the following to customize your node, then press Enter
   to continue.

 Cell and Node Definitions

   WebSphere Application Server home directory:
     /WebSphere/V6R0
         / AppServer

   Node Host Name:


   Cell name (short)......:  AQTS
   Cell name (long).......:  AQTS

   Node name (short)......:  AQTS
   Node name (long).......:  AQTS

   Admin asynch operations procedure name:  BBOW6SH
```

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (2 of 4)

   Specify the following to customize your node, then press Enter
   to continue.

 Procedure Name Definitions

   Controller Information

     Procedure name:  BBO6ACR
     User ID.......:  ASCR1
     UID...........:  2431

   Servant Information

     Procedure name:  BBO6ASR
     User ID.......:  ASSR1
     UID...........:  2432

   Control Region Adjunct
```

```
     Procedure name:  BBO6CRA
     User ID.......:  ASCRA1
     UID...........:  2433
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (3 of 4)

   Specify the following to customize daemon definitions for your
   node, then press Enter to continue

 Location Service Daemon Definitions

   Daemon home directory:
     /WebSphere/V6R0/Daemon

   Daemon jobname:  BBODMNB

   Procedure name.:  BBO6DMN
   User ID........:  WSDMNCR1
   UID............:  2411
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Server Customization (4 of 4)
   Specify the following, which will be used in a job to
   federate your node into the specified Deployment Manager cell.

 WebSphere Application Server home directory:
     /WebSphere/V6R0
         / AppServer
 Deployment Manager Access
   Node host name...........:
   JMX SOAP port............:  8879
   Deployment manager security is enabled:  N
      User ID.............:  WSADMIN

 Node group name.......:  DefaultNodeGroup

 Node Agent Definitions
   Server name (short)...:  BBON001
   Server name (long)....:  nodeagent
   JMX SOAP connector port........:  9360
   Node Discovery port............:  7272
   Node Multicast Discovery port..:  5000

 The High Availability Manager Host must not be multihomed
   High Availability Manager Host:
   High availability manager communication port:  9354

 ORB listener host name...:  *
   ORB port........................:  2809
   ORB SSL port...................:  0
```

Note: The ″High Availability Manager Host″ field was removed from the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.

8. On the ″Define Variables to Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″View Security Domain Configuration Panels″ and press **Enter**. These panels display values you previously set in the ″Configure security domain″ option--you cannot change any of the values here. If you do want to make changes, you must go back to the main dialog panel and run through the ″Configure security domain″ option again.

Creating the customization jobs and files:

You must select configuration data sets to use and complete the process of defining variables for this task. See “Choosing configuration data sets” on page 199 and “Setting the customization variables: Managed node” on page 199 for more information.

The Customization Dialog creates customization batch jobs and data files, based on the variable values you specified in the dialog. The batch jobs and data sets will be written to the *config_hlq*.CNTL and *config_hlq*.DATA configuration data sets that you created with the ″Allocate target data sets″ option.

1. Ensure that the configuration data sets are allocated and not in use.

   Note: Editing a member in *config_hlq*.CNTL or *config_hlq*.DATA will cause this task to fail.
2. On the ″Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″Generate customization jobs″ and press **Enter**. You will have one of two results:
   - **Result A:** If all variables are defined correctly, you see the ″Specify Job Cards″ panel, which looks similar to this:

     ```
     ------------  WebSphere Application Server for z/OS Customization     --------
     Option  ===>

     Generate Customization Jobs

      This portion of the Customization Dialog generates the jobs you must
      run after you complete this Dialog process. You must complete the
      customization process before you generate the jobs with this step.
      If you have not done this, please return to that step.

      Jobs and data files will get generated into data sets:
        'hlq.CNTL'
        'hlq.DATA'
      If you wish to generate customization jobs using other data sets, then
      exit from this panel and select the "Allocate target data sets" option.

      All the jobs that will be tailored for you will need a job card.
      Please enter a valid job card for your installation below. The
      file tailoring process will update the jobname for you in all the
      generated jobs, so you need not be concerned with that portion of
      the job cards below. If continuations are needed, replace the
      comment cards with continuations.

      Specify the job cards, then press Enter to continue.

      //jobname  JOB (ACCTNO,ROOM),'userid',CLASS=A,REGION=0M
      //*
      //*
      //*
     ```

     Note: Pay particular attention to the displayed target data sets. Make sure that they are the ones you intend to use.
   - **Result B:** If the variables are not defined correctly, you will see the ″Verification″ panel. Decide whether the warnings or errors are serious enough to warrant returning to the ″Define variables″ option.

     Note: If the return code is 8 or greater, return to the ″Define variables″ option and fix the uncovered problems. If you saved the variables previously, be sure to re-save them after making any updates.
3. Fill in the job card information, according to your installation requirements. For each job, the dialog generates a jobname and the ″JOB″ keyword to match the member name of the PDS, but you specify the rest.

Note: If you need to run these jobs on a particular system in the sysplex (for example, JES2 MAS or JES3 complex), you should specify the necessary Scheduling Environment (SCHENV), JES2 JOBPARM, or JES3 //*MAIN statement at this time.

Example of a job card entry:

```
//jobname JOB 1234,USER1,NOTIFY=????,MSGCLASS=O,REGION=0M
//*            USER=SYSADM1,PASSWORD=SYSADM1
/*JOBPARM SYSAFF=SYSB
```

Note: This example is useful for jobs that require a user ID other than that of the logged-on TSO user. (This is typically a user ID with UID=0.) In that case, you can just put a comma at the end of the first line, put in the correct user ID on the second line, then uncomment that second line.

You might want to use RACF SUBMIT authority to avoid having to keep passwords in your configuration data sets.

4. Fix any errors. If there are errors anywhere, you will see the ″Error″ panel. Press PF3 to exit the error panel, then enter the correct panel to fix the errors. Then return to the ″Generate Customization Jobs″ option and pick up where you left off. If necessary, you can update the variables and rerun this option. The generation process will delete and re-tailor all the members.

   Note: Compress the configuration data sets before you rerun this option.

You are done when all the jobs are generated. You can move ahead to viewing the generated jobs. See “Following the generated customization instructions: Managed node” for more information.

Following the generated customization instructions: Managed node:

You must generate the customization jobs and files for this task.

The Customization Dialog creates a set of instructions for each customization task. Follow these instructions to tailor and customize a managed node on your system.

Note: Do not attempt to fix a typographical error or make a change by modifying the generated output. Many of the variables are used in multiple members of the target data sets; if you do not change them all, therefore, you will run into problems that are very difficult to diagnose.

1. On the ″Configure Managed Node″ panel, type the appropriate number in the *Option* field to select ″View instructions″ and press **Enter**. ISPF Browse will open and you will see the BBOMNINS member of *config_hlq*.CNTL.
2. Read the instructions carefully, both to preview the customization process and to find any typographical or other errors you might have made while entering the customization variable values.
3. Follow the instructions as given. There are two ways to follow the directions:
   - Follow the instructions while remaining in ISPF Browse.
   - Record the data set name and member at the top of the screen and either print the instructions or use ISPF split screen and browse or edit the instructions while you follow them.
4. Fix any problems. If you encounter problems caused by your Customization Dialog values, modify your variables using the dialog, regenerate the instructions, and restart the customization process.

   Note: Remember that you cannot generate new customization jobs while either configuration data set is open!

You are done with this customization task when you have successfully followed the generated instructions.

Sample generated instructions: Managed node:

This article presents a sample of what the Customization Dialog’s generated instructions might look like. This is a sample only--you must use the instructions generated from your own variables when configuring your system.

Note: This sample is based on the installation of WebSphere Application Server for z/OS Version 6.0.2. Generated instructions from different releases of Version 6 would have different content.

```
-----------------------------------------------
Instructions for customizing WebSphere for z/OS
for a Managed node.

The customization dialog has created jobs based on the information you
provided. These instructions tell you how to modify the operating
system and run the jobs to customize WebSphere for z/OS.

RULES:

1.  If you created the target data sets (*.CNTL and *.DATA) on another
    (driving) system, you must copy them to the target system and give
    them the same data set names.

2.  You must perform these instructions on your target system.

Doing manual configuration updates
----------------------------------

The customization dialog for WebSphere for z/OS does not attempt to
update configuration data for your base operating system or existing
subsystems. You must do the following manual steps prior to running
the WebSphere for z/OS configuration jobs.

Perform these steps to do manual configuration updates:

1.  Update BLSCUSER. Refer to member BBOIPCSP in

    'USERID.TODAY.CNTL'

    In order to use the IPCS support provided by the product, append
    the contents of this member to the BLSCUSER member in your IPCSPARM
    or system PARMLIB datasets.

    -------------------------------------------------------------------

2.  Update SCHEDxx. Refer to member BBOSCHED in

    'USERID.TODAY.CNTL'

    In order to set the correct program properties for the WebSphere
    for z/OS run-time executables, append the contents of this member
    to the SCHEDxx member in your system PARMLIB concatenation.

    Note: When you are finished, issue the command SET SCH=(xx,xx)
    to activate SCHEDxx and load a new program properties table.

    -------------------------------------------------------------------

3.  Make sure the following data sets are APF-authorized:

        DBEL.WASX.SBBOLPA
        DBEL.WASX.SBBOLOAD
        DBEL.WASX.SBBGLOAD
        DBEL.WASX.SBBOLD2

    Add these datasets to your PROGxx or IEAAPFxx parmlib members, as
    appropriate, ensuring you specify the correct volsers.

    -------------------------------------------------------------------

4.  If you want to collect the SMF120 records created by the run-time
    servers, update SMFPRMxx via the following:
```

```
    EXAMPLE:

       SUBSYS(STC,EXITS(IEFU29,IEFACTRT),INTERVAL(SMF,SYNC),
                          TYPE(0,30,70:79,88,89,120,245))
                                                ---

    For details on the SMF records, see related topics in the
    WebSphere for z/OS Information Center at
    http://www.ibm.com/software/webservers/appserv/zos_os390/library/

    ------------------------------------------------------------------

5.  Update your active BPXPRMxx member to have the following WebSphere
    for z/OS configuration file system:

    OMVS.WAS.CONFIG.HFS

    mounted at:

    /WebSphere/V6R1

    in read/write mode.

    EXAMPLE:

       MOUNT FILESYSTEM('OMVS.WAS.CONFIG.HFS')
         MOUNTPOINT('/WebSphere/V6R1')
          TYPE(HFS)
          MODE(RDWR)

    ------------------------------------------------------------------

6.  Update TCP/IP by reserving the following ports for WebSphere for
    z/OS.  These will be used during the federation process of your
    managed node.

       SOAP JMX Connector port                     - 9360
       Node Discovery port                         - 7272
       Node Multicast Discovery Port               - 5000
       Node Agent's ORB port                       - 2809
       High Availability Manager Communication port - 9354

    View member BBOTCPIM in

    'USERID.TODAY.CNTL'

    Add the contents of this member to the PORT section of the file
    referenced by the DD statement for the TCP/IP profile in the
    TCP/IP start procedure. Cut and paste from this member into the
    data set used by your installation.

    ATTENTION: If another application has already reserved any of these
    ports for its own use, you must resolve the resulting conflict
    before you continue. If you update the WebSphere for z/OS
    customization dialog with new port specifications, be sure to
    regenerate the customization jobs, data, and instructions.


Note: The addNode process introduces a special utility server to
      the node.  This utility server is called a nodeagent and
      exists to support administrative functions on the node. By
      default the nodeagent takes over ORB port 2809.   Note on
      WebSphere z/OS the ORB port doubles as the INS CosNaming
      bootstrap port.  By default, this port (2809) was assigned
      to the Application Server.  Normally you want the nodeagent
      to be the INS CosNaming bootstrap point for the entire node,
      so that RMI/IIOP clients that do not override the INS
      CosNaming bootstrap defaults can locate within the namespace,
      EJBs installed on any server on that node.  In order for the
      nodeagent to take over port 2809,  the Application Server must
      be assigned a new ORB port.  The default new ORB port for the
      Application Server is 9810.  The nodeagent will take over a
      Application Server's ORB port if and only if the nodeagent's ORB
      port is equal to an Application Server's ORB port.  You can
      specify the nodeagent's ORB port in the 'ORB port' field.
      You can specify the new ORB port for the Application Server in
      the 'Appplication Server's ORB Port' field.

    ATTENTION: Skip this step if the ports are already defined in the
    TCP/IP profile.

    ------------------------------------------------------------------

7.  The WebSphere product libraries will be placed in STEPLIB as
    needed, rather than in the system link pack area and system link
    list. Make sure that the target MVS system has at least 8MB
    of free storage in extended CSA for the daemon and for EACH
    node (deployment manager node or application server node).

    SBBOLOAD and SBBOLD2:
    ====================
    The following data sets will be placed in the STEPLIB concatenation
    for the location service daemon, controller and servant regions,
    and in the setupCmdLine.sh script in the WebSphere Configuration
    file system. You must not remove these STEPLIB statements.

    DBEL.WASX.SBBOLOAD
    DBEL.WASX.SBBOLD2


    BBORTS61:
    =========

    The BBORTS61 module is used by WebSphere Application Server for
    component trace support. A copy of this module (any maintenance
    level) must be in the system link pack area in order for CTRACE
    to work correctly.

    If a copy of BBORTS61 is currently loaded into LPA, you need take
    no further action.

    Otherwise, issue the following MVS console command to load BBORTS61
    into dynamic LPA:

      SETPROG LPA,ADD,MODNAME=BBORTS61,
              DSNAME=DBEL.WASX.SBBOLPA

    Alternatively, you can place the following statement in a parmlib
    PROGxx member which is activated with the SET PROG= command after
    system IPL is complete:

    LPA ADD MODNAME(BBORTS61)
      DSNAME(DBEL.WASX.SBBOLPA)

    Make sure that the BBORTS61 module is loaded into LPA after each
    system IPL.

    ------------------------------------------------------------------

8.  WebSphere for z/OS customization assumes that the following system
    data sets are in the system link list or link pack area:

    Language Environment     SCEERUN
                             SCEERUN2
    System SSL               SGSKLOAD (z/OS 1.5 and below)
                             SIEALNKE (z/OS 1.6 and above)

    Placing these data sets in the link list or link pack area improves
    performance and insulates your WebSphere for z/OS configuration
    from changes in data set names (for example, when migrating to z/OS
    1.6).

    If the Language Environment or System SSL load module libraries are
    not in your system link list or link pack area, you must perform
    the following steps before starting any WebSphere Application
    Server for z/OS servers:

    - Make sure the data sets are APF-authorized
    - Complete the optional step below to add the data sets to STEPLIB
      in the server JCL and setupCmdLine.sh script(s).

    If you regenerate server cataloged procedures at any point, make
    sure the data sets are added to the new cataloged procedures.

    -------------------------------------------------------------------

9.  If the error logstream
    WAS.ERROR.LOG
    does not already exist on your target system, make a copy of the
    appropriate job in the SBBOJCL data set, customize it according
    to the comments in the job, and run it:

       BBOERRLC    Create an error logstream in a coupling facility
       BBOERRLD    Create a DASD-only error logstream

    -------------------------------------------------------------------

10. WebSphere for z/OS regions open a large number of files (more than
    1024). Make sure your BPXPRMxx parmlib member(s) specify a value of
    MAXFILEPROC that is greater than or equal to 2000. Use the
    following MVS console command to see your current MAXFILEPROC
    setting:

      D OMVS,OPTIONS

Running the customized jobs
---------------------------

The customization dialog built a number of batch jobs with the
variables you supplied. You must run the jobs in the order listed
below using user IDs with the appropriate authority.

BEFORE YOU BEGIN: Complete the section above entitled "Doing manual
configuration updates."

Follow the table below, which lists in order the jobs you must submit
and the commands you must enter. Special handling notes are included
in the table. All jobs are members of

USERID.TODAY.CNTL

Attention: After submitting each job, carefully check the output.
Errors may exist even when all return codes are zero.

+-----------+-------------------------------------------------------+
| BBOMSGC   | User ID requirement: Update authority for data set    |
+-----------+ SYS1.MSGENU and/or SYS1.MSGJPN.                       |
| Done:     |                                                       |
|           | ATTENTION: This is optional unless you require message |
|           | translation.                                          |
| By:       |                                                          |
|           | This job sets up MMS to translate messages for WebSphere |
|           | for z/OS.                                                |
|           |                                                          |
|           | There are two steps to update SYS1.MSGENU and            |
|           | SYS1.MSGJPN. Remove the unneeded step and change the     |
|           | target libraries, if necessary.                          |
+-----------+----------------------------------------------------------+
|           | Check WebSphere Application Server home directories.     |
+-----------+                                                          |
| Done:     | Verify that the following directories exist on your      |
|           | target z/OS system and that the ownership and permission |
|           | bits are correct:                                        |
| By:       |                                                          |
|           | /var/WebSphere/home                                      |
|           | ownership: (any)                                         |
|           | permission bits: 755                                     |
|           |                                                          |
|           | /var/WebSphere/home/WSCFG1                               |
|           | ownership: WSOWNER:WSCFG1                                |
|           | permission bits: 770                                     |
|           |                                                          |
|           |                                                          |
|           | /var/WebSphere/home/WSSR1                                |
|           | ownership: WSOWNER:WSSR1                                 |
|           | permission bits: 770                                     |
|           |                                                          |
|           |                                                          |
|           | /var/WebSphere/home/WSCLGP                               |
|           | ownership: WSOWNER:WSCLGP                                |
|           | permission bits: 770                                     |
|           |                                                          |
|           | If the these directories do not exist, create them with  |
|           | the above ownership and permission bits.                 |
|           |                                                          |
|           | The security domain configuration job BBOSBRAM can be    |
|           | used to create these directories if necessary.           |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOMBRAJ  | User ID requirement: Authority to update data set        |
+-----------+                                                          |
| Done:     | USERID.TODAY.DATA                                        |
|           |                                                          |
|           | This job builds (but does not execute) the RACF commands |
| By:       | for the WebSphere for z/OS run-time clusters and places  |
|           | them into member BBOMBRAK of data set                    |
|           |                                                          |
|           | USERID.TODAY.DATA                                        |
|           |                                                          |
|           | Carefully review these definitions with your security    |
|           | administrator.                                           |
+-----------+----------------------------------------------------------+
| BBOMBRAK  | User ID requirement: RACF special authority.             |
+-----------+                                                          |
| Done:     | This job executes the RACF commands set up in the        |
|           | previous job.                                            |
|           |                                                          |
|           | RESULT: You may receive errors, such as INVALID USER     |
| By:       | messages, from this job because a user ID, group  or     |
|           | profile is already defined.  Make sure the existing      |
|           | user ID, group or profile has the same characteristics   |
|           | as the user ID, group or profile being created by        |
|           | BBOMBRAK.  If not, then change the values in the         |
|           | customization dialog which are causing the conflict,     |
|           | regenerate the customization jobs, and restart the       |
|           | process.                                                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | Check user ID authorizations.                            |
+-----------+                                                          |
| Done:     | Make sure the WSCFG1 group has read access to all        |
|           | WebSphere product data sets, as well as to any other     |
|           | data sets which will be placed in WebSphere for z/OS     |
|           | cataloged procedure STEPLIB concatenations.              |
|           |                                                          |
| By:       | the resolver configuration file in use on your system.   |
|           | Depending on your IP setup, this file may be             |
|           | /etc/resolv.conf, SYS1.TCPPARMS(TCPDATA), or another     |
|           | data set.                                                |
|           |                                                          |
|           | ASCR1                                                    |
|           | ASSR1                                                    |
|           |                                                          |
|           | See the z/OS eNetwork Communication Server IP            |
|           | Configuration manual for the resolver search order.      |
|           |                                                          |
|           | Ensure the following user ID has read access to the data |
|           | sets in your system parmlib concatenation:               |
|           |                                                          |
|           | WSDMNCR1                                                 |
|           |                                                          |
|           | ATTENTION:                                               |
|           |                                                          |
|           |  If operator commands are protected by the z/OS security |
|           |  server at your installation, you must ensure that       |
|           |  sufficient authority is given to WebSphere tasks to     |
|           |  control operations.                                     |
|           |                                                          |
|           |  The Application Server Controller user ID (ASCR1)       |
|           |  needs the ability to perform operations on started      |
|           |  tasks belonging to WebSphere Application Server for     |
|           |  z/OS.                                                   |
|           |                                                          |
|           |  The asynchronous administrator user ID, and any user    |
|           |  ID used to run the federation job when the node agent   |
|           |  is started automatically, need the authority to issue   |
|           |  the MVS START command.                                  |
|           |                                                          |
|           |  If you are currently controlling MVS console command    |
|           |  authority with SAF OPERCMDS profiles, grant the         |
|           |  following authorities as indicated, substituting your   |
|           |  own profile names:                                      |
|           |                                                          |
|           |  PERMIT  START_profile_name  CLASS(OPERCMDS)             |
|           |          ID (ASCR1  WSADMSH)  ACCESS(UPDATE)             |
|           |                                                          |
|           |  PERMIT  STOP_profile_name  CLASS(OPERCMDS)              |
|           |          ID (ASCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  MODIFY_profile_name  CLASS(OPERCMDS)            |
|           |          ID (ASCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  CANCEL_profile_name  CLASS(OPERCMDS)            |
|           |          ID (ASCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  PERMIT  FORCE_profile_name  CLASS(OPERCMDS)             |
|           |          ID (ASCR1 )  ACCESS(UPDATE)                     |
|           |                                                          |
|           |  You must also grant the appropriate console command     |
|           |  authority to any user ID that executes the              |
|           |  startServer.sh or stopServer.sh script.                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOMCHFS  | User ID requirement: UID=0 and authority to allocate     |
+-----------+                                                          |
| Done:     | OMVS.WAS.CONFIG.HFS                                      |
|           |                                                          |
|           | This job:                                                |
|           |                                                          |
| By:       | o   Creates a mount point directory                      |
|           |                                                          |
|           |     /WebSphere/V6R1                                      |
|           |                                                          |
|           | o   Allocates the configuration file system using        |
|           |     the Hierarchical File System (HFS)                   |
|           |                                                          |
|           |     OMVS.WAS.CONFIG.HFS                                  |
|           |                                                          |
|           |     and mounts it at the above mount point.              |
|           |                                                          |
|           | DO NOT RUN THIS JOB IF:                                  |
|           |   1. The configuration file system already exists and is |
|           |      mounted at the desired mountpoint, or if            |
|           |                                                          |
|           |   2. The mount point directory is controlled by          |
|           |      automount.  Either disable the automount rule for   |
|           |      the configuration mount point while running this    |
|           |      job, or perform the following steps manually:       |
|           |                                                          |
|           |      a. Allocate the configuration file system data set. |
|           |      b. Issue the following shell commands, which will  |
|           |         also cause automount to mount the file system    |
|           |                                                          |
|           |      chmod 775 /WebSphere/V6R1                           |
|           |      chown WSOWNER:WSCFG1 /WebSphere/V6R1                |
|           |                                                          |
|           |                                                          |
|           | BEFORE YOU BEGIN: The BBOMCHFS job assumes your root     |
|           | file system is mounted in read/write mode.  If the root  |
|           | file system is not mounted in read/write mode, manually  |
|           | create the directory                                     |
|           |                                                          |
|           | /WebSphere/V6R1                                          |
|           |                                                          |
|           | and any needed higher directories, set file permissions  |
|           | to 775, and set the owning user ID and group to WSOWNER  |
|           | and WSCFG1 before running BBOMCHFS.                      |
|           |                                                          |
|           | EXAMPLE: If you plan to use /WebSphere/V6R0M0 as your    |
|           | directory, issue the following commands from within the  |
|           | OMVS shell:                                              |
|           |                                                          |
|           |   mkdir -p -m 775 /WebSphere/V6R0M0                      |
|           |   chown -R WSOWNER:WSCFG1 /WebSphere                     |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOMHFSA  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job populates the previously-created file system.   |
|           |                                                          |
|           | Upon completion, examine the job output. Success is      |
| By:       | indicated with a RC=0 in the job output.                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOWCPYM  | User ID requirement: update authority for:               |
+-----------+                                                          |
| Done:     |     SYS1.PROCLIB                                         |
|           |                                                          |
| By:       |                                                          |
|           |                                                          |
|           |                                                          |
|           | This job copies the tailored start procedures,           |
|           | parameters, and EXECs to the run-time libraries.         |
|           |                                                          |
|           | ATTENTION: Be aware that you may overlay existing        |
|           | members in the above data sets.                          |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOWWPFM  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job sets up the runtime file system.                |
|           |                                                          |
|           |                                                          |
| By:       | Upon completion, examine the job output. Success is      |
|           | indicated by rc=0.                                       |
|           |                                                          |
|           | Note: If the BBOWWPFM (profile creation) job fails,      |
|           | delete the WAS_HOME/profiles/default directory and all   |
|           | its contents before rerunning BBOWWPFM.                  |
|           |                                                          |
+-----------+----------------------------------------------------------+
| BBOMHFSB  | User ID requirement: UID=0.                              |
+-----------+                                                          |
| Done:     | This job will complete the file system initialization.   |
|           |                                                          |
|           | Upon completion, examine the job output. Success is      |
| By:       | indicated with a RC=0 in the job output.                 |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | All WebSphere Application Server processes require       |
+-----------+ access to the Language Environment and System SSL load   |
| Done:     | modules.                                                 |
|           |                                                          |
|           | If the SCEERUN, SCEERUN2 and System SSL load module      |
|           | libraries are not in the system link list or link pack   |
|           | area, add them to the STEPLIB DD concatenation in each   |
|           | of the following cataloged procedures in                |
|           | SYS1.PROCLIB:                                            |
|           |                                                          |
|           |     BBO6ACRZ                                             |
|           |     BBO6ASRZ                                             |
|           |     BBO6CRAZ                                             |
|           |     BBO6DMNZ                                             |
|           |                                                          |
|           | and also add the full data set names, separated by       |
|           | colons (:), to the STEPLIB variable in the shell script  |
|           |                                                          |
|           |     /WebSphere/V6R1/                                     |
|           |      AppServer/                                          |
|           |       profiles/default/bin/setupCmdLine.sh               |
| By:       |                                                          |
|           | When modifying the setupCmdLine.sh script, do not        |
|           | remove lines or comment them out, as this may cause      |
|           | problems with automated updates to the script.           |
|           |                                                          |
|           | Add only those data sets which are NOT in the link list  |
|           | or link pack area.                                       |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | Make sure Resource Recovery Services (RRS) is active.    |
+-----------+ (See the InfoCenter for setup instructions if necessary.)|
| Done:     | Look for the following console message to verify that    |
|           | RRS was successfully started:                            |
|           |                                                          |
| By:       |                                                          |
|           |   ASA2011I RRS INITIALIZATION COMPLETE. COMPONENT        |
|           |     ID=SCRRS                                             |
|           |                                                          |
+-----------+-----------------------------------------------------------+
| BBOWMNAN  | User ID requirement: WSADMIN                              |
+-----------+                                                           |
| Done:     | If you choose the start the node agent automatically,     |
|           | the user ID used to run BBOWMNAN will also need the       |
|           | authority to issue the MVS START command.                 |
|           |                                                           |
|           | This job will federate your node into the specified       |
|           | Deployment Manager cell.  Ensure that your Deployment     |
|           | Manager is running before submitting this job             |
|           |                                                           |
|           | Upon completion, examine the job output. Success is       |
| By:       | indicated with a RC=0 in the job output.                  |
|           |                                                           |
+-----------+-----------------------------------------------------------+
| --------  | If your system is busy, you may want to include a rule    |
+-----------+ in your WLM policy that OMVS work for job BBON001         |
| Done:     | (such as the postinstaller step) is to run in a service   |
|           | class with a high service objective.                      |
| By:       |                                                           |
+-----------------------------------------------------------------------+
| --------  | Start the node agent server                               |
+-----------+                                                           |
|           | Note: The node agent is automatically started by the node|
|           | federation process.  This step is information, for        |
|           | starting the node agent at other times.                   |
|           |                                                           |
|           | Issue the following MVS command to start your node agent  |
|           | server, replacing dmgr_cell_short_name with the     |
| Done:     | cell short name of the target Deployment Manager cell     |
|           |                                                           |
| By:       |   START BBO6ACR,JOBNAME=BBON001,                          |
|           |         ENV=dmg_cell_short_name.AQFT.BBON001        |
|           |                                                           |
|           | RESULT: The following message appears on the console and  |
|           | in the job log of BBON001.                                |
|           |   BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR     |
|           |   z/OS CONTROL PROCESS BBON001                            |
|           |                                                           |
+-----------+-----------------------------------------------------------+
+-----------+-----------------------------------------------------------+
| The product is now configured.  You may create and manage             |
| application servers in the node using the administrative console or   |
| scripting.                                                            |
|                                                                       |
+-----------------------------------------------------------------------+
```

Working with your new managed server node:

Once you complete the customization instructions, you will have a WebSphere Application Server for z/OS application server node (managed node) in your Network Deployment cell.

Before starting your server

Make sure that the WebSphere Application Server for z/OS product HFS and configuration HFS are mounted. If you chose to load the SBBOLPA (and possibly SBBOLOAD) into the system link pack area, make sure that these libraries are loaded into LPA before starting the server.

Starting the node agent

To start your node agent, issue the following MVS console command:

```
START server_proc,JOBNAME=nodeagent_name,ENV=cell_name.node_name.nodeagent_name
```

where

- *server_proc* is the node agent cataloged procedure.
- *nodeagent_name* is the node agent short name.
- *node_name* is the node short name.
- *cell_name* is the cell short name.

If you chose default values for example (your sysplex is named CELL1 and your system is named MVSA), you would enter the following START command:

```
START BBO6ACR,JOBNAME=BBON001,ENV=CELL1.MVSA.BBON001
```

The START command brings up the node agent. The node agent starts the location service daemon (if one is not already running). You should see a message like the following when the node is up and running:

```
BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR z/OS CONTROL PROCESS BBON001
```

The node agent must be running in order for the deployment manager to administer the node.

Administering the node through the administrative console

If the deployment manager for the cell is up and running, access the administrative console by pointing a Web browser to the following URL:

```
http://hostname:http_port/ibm/console
```

where

- *hostname* is the deployment manager HTTP transport host name you specified during customization.

  Note: If you specified ″*″ for the deployment manager HTTP host name, this is actually the deployment manager node host name.
- *http_port* is the deployment manager HTTP port you specified during customization.

  Note: The default HTTP port for the deployment manager is 9060.

Until administrative security is enabled, you will see a signon screen that asks you for a user ID but no password.

The user ID can be anything and is used only to provide basic tracking of changes. Be aware that until you enable administrative security, anyone with a Web browser and access to the HTTP port can modify your application serving environment.

You can use the administrative console, scripting, or both to manage the node and deploy and manage J2EE applications. See Using the administrative clients for more information. Before you can deploy applications, however, you need to add application servers to your managed node.

Adding application servers

Application servers can be added to the managed server node using the administrative console or scripting. Either of two methods can be used:

- Create a new application server directly using the administrative console or scripting. See Creating application servers for more information. You can use the controller, servant and CRA cataloged procedures and user IDs created during the managed node setup process for any application servers you create in the managed node.
- Cluster an existing application server in another node, using this managed node as a target. This will create a ″cloned″ copy of the application server being clustered in your new managed node. See Creating clusters for more information.

Starting application servers

To start one of your managed node's application servers, issue the following MVS console command:

```
START server_proc,JOBNAME=server_name,ENV=cell_name.node_name.server_name
```

where

- *server_proc* is the application server agent cataloged procedure (can be the same as the node agent cataloged procedure).
- *server_name* is the application server short name.
- *node_name* is the node short name.
- *cell_name* is the cell short name.

If you chose the default procedure name for example (your sysplex is named CELL1, your node is named MVSA, and your server is named AZSR01A), you would enter the following START command:

```
START BBO6ACR,JOBNAME=AZSR01A,ENV=CELL1.MVSA.AZSR01A
```

The START command brings up the application server controller. The controller starts the location service daemon (if one is not already running) and then uses WLM to start the control region adjunct and the servants. You should see a message similar to the following when the node is up and running:

```
BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR Z/OS CONTROL PROCESS AZSR01A
```

Stopping your managed node

Use one of the following two methods to stop your deployment manager:

- Stop the location service daemon, which also stops any of the cell's nodes on the same z/OS system. The location service daemon holds pointers to modules in common storage, and stopping it forces all cell members on the same z/OS system as the daemon to shut down. To stop the location service daemon, enter the following MVS console command:

  ```
  STOP daemon_jobname
  ```

  where *daemon_jobname* is the location service daemon jobname. The default location service daemon jobname for a Network Deployment cell is BBODMNC.
- Stop just the node agent and its application servers while leaving the location service daemon, deployment manager (if present), and any other managed nodes on the z/OS system still running. To stop the node agent, enter the following MVS console command:

  ```
  STOP nodeagent_name
  ```

  where *nodeagent_name* is the node agent short name. The default node agent short name is BBON001.

Federating a stand-alone application server into a Network Deployment cell

This article leads you through the tasks involved in federating a WebSphere Application Server for z/OS stand-alone application server into a Network Deployment cell.

Perform this task to federate an existing WebSphere Application Server for z/OS stand-alone application server into a Network Deployment cell.

1. Log on to TSO on the z/OS system on which you intend to federate the server. Use a user ID that has READ access to the WebSphere Application Server for z/OS product data sets. You will also need access to a user ID with authority to make security system updates and a user ID with UID 0. (These can all be the same user ID.)
2. Start the Customization Dialog. See "Starting the Customization Dialog" on page 133 for details.
3. Choose the configuration data sets in which you will store your customization jobs and data. See "Choosing configuration data sets" on page 217 for details.

4. Load the security domain variables saved from the security domain you intend to use for this cell. See “Loading the common MVS groups and users variables” on page 217 for details.
5. Set the customization variables according to the values recorded on your federated application server node worksheet. See “Setting the customization variables: Federated application server node” on page 218 for details.
6. (Optional but recommended.) Save the federated application server node customization variables in a data set. See “Saving the cell variables” on page 183 for details.
7. Create the customization jobs and files, based on the customization variable values you entered. See “Creating the customization jobs and files” on page 219 for details.
8. Follow the generated customization instructions. See “Following the generated customization instructions: Federated application server node” on page 220 for details, and a sample set of customization instructions.

You are done when you have successfully completed the steps in the generated instructions. The new federated application server node is up and running on the chosen z/OS system. See “Working with your new federated server node” on page 223 for more information.

Loading the security domain variables:

This article describes how to complete the ″Load security domain variables″ option for a WebSphere Application Server for z/OS federated application server node.

Create the security domain you will use for the new federated application server node and know the name of the saved security domain configuration variable file that you recorded on the security domain worksheet.

The security domain settings are used in the customization of every WebSphere Application Server for z/OS cell. By loading the security domain variables at the start of node or cell creation, you ensure that the security domain configuration you use is consistent and matches the RACF definitions that have already been set as part of security domain configuration.

Complete this task as the first step in configuring a new federated application server node. If you encounter problems during customization and change the security domain variable values, be sure to re-save them.

1. On the ″Federate stand-alone application server node″ panel, type the appropriate number in the *Option* field to select ″Load security domain variables″ and press **Enter**. You will see a panel that looks similar to the following:

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Load Security Domain Variables

 Specify the name of a data set containing the security domain variables,
 then press Enter to continue.

 IBM-supplied defaults are in ''


 Data set name:


 If this data set is not cataloged, specify the volume.

 Volume:
```

2. Fill in the name of the sequential data set you used to hold the security domain variable values and press **Enter**. The security domain variables will load.

The security domain settings are loaded. You can display these variables, but not change them.

Loading the common MVS groups and users variables:

This article describes how to complete the ″Load common MVS groups and users variables″ option for a WebSphere Application Server for z/OS federated application server node.

Create the common MVS groups and users you will use for the new federated application server node and know the name of the saved common MVS groups and users configuration variable file that you recorded on the common MVS groups and users worksheet.

The common MVS groups and users settings are used in the customization of every WebSphere Application Server for z/OS cell. By loading the common MVS groups and users variables at the start of node or cell creation, you ensure that the common MVS groups and users configuration you use is consistent and matches the RACF definitions that have already been set as part of common MVS groups and users configuration.

Complete this task as the first step in configuring a new federated application server node. If you encounter problems during customization and change the common MVS groups and users variable values, be sure to re-save them.

1. On the ″Federate stand-alone application server node″ panel, type the appropriate number in the *Option* field to select ″Load common MVS groups and users variables″ and press **Enter**. You will see a panel that looks similar to the following:

   ```
   ------------  WebSphere Application Server for z/OS Customization     --------
   Option  ===>

   Load common MVS groups and users Variables

    Specify the name of a data set containing the common MVS groups and users variables,
    then press Enter to continue.

    IBM-supplied defaults are in ''


    Data set name:


    If this data set is not cataloged, specify the volume.

    Volume:
   ```

2. Fill in the name of the sequential data set you used to hold the common MVS groups and users variable values and press **Enter**. The common MVS groups and users variables will load.

The common MVS groups and users settings are loaded. You can display these variables, but not change them.

Choosing configuration data sets:

This article leads you through the ″Allocate target data sets″ option in the Customization Dialog.

You must start the Customization Dialog and select the ″Federate an existing stand-alone application server node into an existing Network Deployment cell″ option.

Each option in the Customization Dialog saves customization jobs and files in a pair of customization data sets. While is it possible to reuse these data sets, it is safest to create separate data sets for each WebSphere Application Server for z/OS configuration. The best practice is to use the customization data set name prefix (sometimes referred to as ″*config_hlq*″) to indicate the version and release of WebSphere Application Server for z/OS, the task you are performing, and the cell (and, in some cases, the node name) you are configuring.

Complete this task before generating the customization jobs and files.

1. On the main dialog panel, type the appropriate number in the *Option* field to select ″Allocate target data sets″.
2. Press Enter. **Result:** You see a panel that looks similar to the following:

```
-----------------     WebSphere Application Server for z/OS Customization      ------------------
Option  ===>

 Allocate Target Data Sets

 Specify a high level qualifier (HLQ) and press Enter to allocate the
 data sets to contain the generated jobs and instructions. You can
 specify multiple qualifiers (up to 39 characters).

 High level qualifier:                                        .CNTL
                                                              .DATA

 The Dialog will display data set allocation panels. You can make
 changes to the default allocations, however you should not change
 the DCB characteristics of the data sets.

    .CNTL  - a PDS with fixed block 80-byte records to
             contain customization jobs.

    .DATA  - a PDS with variable length data to contain
             other data produced by the Customization Dialog.
```

3. Fill in your chosen configuration data set name prefix value (*config_hlq*). If the data sets *config_hlq*.CNTL and *config_hlq*.DATA do not exist, you will be prompted for data set allocation information. If the data sets already exist, a message will inform you that they will be reused.

The data sets *config_hlq*.CNTL and *config_hlq*.DATA are allocated and will store customization jobs and files. These data set names will also be saved along with the customization variables.

Setting the customization variables: Federated application server node:

This article describes how to complete the ″Define variables″ option for a WebSphere Application Server for z/OS federated node.

You must start the Customization Dialog and select the ″Create stand-alone Application Server nodes″ option. Have the Federating an application server node Customization Dialog worksheet completed and at hand. A copy of this worksheet is available in the *Installing your application serving environment* PDF.

1. On the ″Federate stand-alone application server Node″ panel, type the appropriate number in the *Option* field to select ″Define variables″ and press **Enter**.
2. On the ″Define Variables for Federate stand-alone Application Server Node″ panel, type the appropriate number in the *Option* field to select ″Define Variables for Node Federation″ and press **Enter**.
3. Fill in the ″Define Variables for Federate stand-alone Application Server Node″ panels using the following screen shots as your guides. When you are done with each, press **Enter**.

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Define Variables for Federate stand-alone application server Node (1 of 2)
   Specify the following to customize your server, then press Enter
   to continue.

 WebSphere Application Server home directory:
    /WebSphere/V6R0
        / AppServer
 Deployment Manager Access
   Node host name...........:
   JMX SOAP port............:  8879
   Deployment manager security is enabled:  N
      User ID.............:  WSADMIN

 Include Apps..........:  Y
 Launch the node agent after node federation:  Y
 Application server's ORB port..:  9810
 Node group name.......:  DefaultNodeGroup

 Node Agent Definitions
   Server name (short)...:  BBON001
   Server name (long)....:  nodeagent
   JMX SOAP connector port........:  9360
   Node Discovery port............:  7272
   Node Multicast Discovery port..:  5000
   High availability manager communication port:  9354

 ORB listener host name...:  *
   ORB port.......................:  2809
   ORB SSL port...................:  0
```

Note: The ″Launch the node agent after node federation″ field was added to the Customization Dialog in WebSphere Application Server for z/OS Version 6.0.2.

```
------------  WebSphere Application Server for z/OS Customization     --------
Option  ===>

Define Variables for Federate stand-alone application server Node (2 of 2)
   Specify the following to customize your server, then press Enter
   to continue.

 Do you wish to federate service integration busses
   that exist on this node?" (Y/N):  N
```

Creating the customization jobs and files:

You must select configuration data sets to use and complete the process of defining variables for this task. See “Choosing configuration data sets” on page 217 and “Setting the customization variables: Federated application server node” on page 218 for more information.

The Customization Dialog creates customization batch jobs and data files, based on the variable values you specified in the dialog. The batch jobs and data sets will be written to the *config_hlq*.CNTL and *config_hlq*.DATA configuration data sets that you created with the ″Allocate target data sets″ option.

1. Ensure that the configuration data sets are allocated and not in use.

   Note: Editing a member in *config_hlq*.CNTL or *config_hlq*.DATA will cause this task to fail.
2. On the ″Federate stand-alone application server Node″ panel, type the appropriate number in the *Option* field to select ″Generate customization jobs″ and press **Enter**. You will have one of two results:
   - **Result A:** If all variables are defined correctly, you see the ″Specify Job Cards″ panel, which looks similar to this:

     ```
     ------------  WebSphere Application Server for z/OS Customization     --------
     Option  ===>

     Generate Customization Jobs

      This portion of the Customization Dialog generates the jobs you must
      run after you complete this Dialog process. You must complete the
      customization process before you generate the jobs with this step.
      If you have not done this, please return to that step.

      Jobs and data files will get generated into data sets:
     ```

```
  'hlq.CNTL'
  'hlq.DATA'
If you wish to generate customization jobs using other data sets, then
exit from this panel and select the "Allocate target data sets" option.

All the jobs that will be tailored for you will need a job card.
Please enter a valid job card for your installation below. The
file tailoring process will update the jobname for you in all the
generated jobs, so you need not be concerned with that portion of
the job cards below. If continuations are needed, replace the
comment cards with continuations.

Specify the job cards, then press Enter to continue.

//jobname  JOB (ACCTNO,ROOM),'userid',CLASS=A,REGION=0M
//*
//*
//*
```

Note: Pay particular attention to the displayed target data sets. Make sure that they are the ones you intend to use.

- **Result B:** If the variables are not defined correctly, you will see the ″Verification″ panel. Decide whether the warnings or errors are serious enough to warrant returning to the ″Define variables″ option.

  Note: If the return code is 8 or greater, return to the ″Define variables″ option and fix the uncovered problems. If you saved the variables previously, be sure to re-save them after making any updates.

3. Fill in the job card information, according to your installation requirements. For each job, the dialog generates a jobname and the ″JOB″ keyword to match the member name of the PDS, but you specify the rest.

   Note: If you need to run these jobs on a particular system in the sysplex (for example, JES2 MAS or JES3 complex), you should specify the necessary Scheduling Environment (SCHENV), JES2 JOBPARM, or JES3 //*MAIN statement at this time.

   Example of a job card entry:

   ```
   //jobname JOB 1234,USER1,NOTIFY=????,MSGCLASS=O,REGION=0M
   //*            USER=SYSADM1,PASSWORD=SYSADM1
   /*JOBPARM SYSAFF=SYSB
   ```

   Note: This example is useful for jobs that require a user ID other than that of the logged-on TSO user. (This is typically a user ID with UID=0.) In that case, you can just put a comma at the end of the first line, put in the correct user ID on the second line, then uncomment that second line.

   You might want to use RACF SUBMIT authority to avoid having to keep passwords in your configuration data sets.

4. Fix any errors. If there are errors anywhere, you will see the ″Error″ panel. Press PF3 to exit the error panel, then enter the correct panel to fix the errors. Then return to the ″Generate Customization Jobs″ option and pick up where you left off. If necessary, you can update the variables and rerun this option. The generation process will delete and re-tailor all the members.

   Note: Compress the configuration data sets before you rerun this option.

You are done when all the jobs are generated. You can move ahead to viewing the generated jobs. See “Following the generated customization instructions: Federated application server node” for more information.

Following the generated customization instructions: Federated application server node:
You must generate the customization jobs and files for this task.

The Customization Dialog creates a set of instructions for each customization task. Follow these instructions to tailor and customize a federated node on your system.

Note: Do not attempt to fix a typographical error or make a change by modifying the generated output. Many of the variables are used in multiple members of the target data sets; if you do not change them all, therefore, you will run into problems that are very difficult to diagnose.

1. On the ″Federate stand-alone application server Node″ panel, type the appropriate number in the *Option* field to select ″View instructions″ and press **Enter**. ISPF Browse will open and you will see the BBOANINS member of *config_hlq*.CNTL.
2. Read the instructions carefully, both to preview the customization process and to find any typographical or other errors you might have made while entering the customization variable values.
3. Follow the instructions as given. There are two ways to follow the directions:
   - Follow the instructions while remaining in ISPF Browse.
   - Record the data set name and member at the top of the screen and either print the instructions or use ISPF split screen and browse or edit the instructions while you follow them.
4. Fix any problems. If you encounter problems caused by your Customization Dialog values, modify your variables using the dialog, regenerate the instructions, and restart the customization process.

   Note: Remember that you cannot generate new customization jobs while either configuration data set is open!

You are done with this customization task when you have successfully followed the generated instructions.

Sample generated instructions: Federated application server node:

This article presents a sample of what the Customization Dialog's generated instructions might look like. This is a sample only--you must use the instructions generated from your own variables when configuring your system.

Note: This sample is based on the installation of WebSphere Application Server for z/OS Version 6.0.2. Generated instructions from different releases of Version 6 would have different content.

```
------------------------------------------------
Instructions for customizing WebSphere for z/OS
for Federate stand-alone Application Server node.

The customization dialog has created jobs based on the information you
provided. These instructions tell you how to modify the operating
system and run the jobs to customize WebSphere for z/OS.

RULES:

1.  If you created the target data sets (*.CNTL and *.DATA) on another
    (driving) system, you must copy them to the target system and give
    them the same data set names.

2.  You must perform these instructions on your target system.

3.  Update TCP/IP by reserving the following ports for WebSphere for
    z/OS:

       SOAP JMX Connector port                    - 9360
       Node Discovery port                        - 7272
       Node Multicast Discovery Port              - 5000
       Node Agent's ORB port                      - 2809
       High Availability Manager Communication port - 9354
       Application Server's ORB Port              - 9810

    View member BBOTCPIA in
```

'USERID.TODAY.CNTL'.

Add the contents of this member to the PORT section of the file referenced by the DD statement for the TCP/IP profile in the TCP/IP start procedure. Cut and paste from this member into the data set used by your installation.

Note: The addNode process introduces a special utility server to the node. This utility server is called a nodeagent and exists to support administrative functions on the node. By default the nodeagent takes over ORB port 2809. Note on WebSphere z/OS the ORB port doubles as the INS CosNaming bootstrap port. By default, this port (2809) was assigned to the Application Server. Normally you want the nodeagent to be the INS CosNaming bootstrap point for the entire node, so that RMI/IIOP clients that do not override the INS CosNaming bootstrap defaults can locate within the namespace, EJBs installed on any server on that node. In order for the nodeagent to take over port 2809, the Application Server must be assigned a new ORB port. The default new ORB port for the Application Server is 9810. The nodeagent will take over an Application Server's ORB port if and only if the nodeagent's ORB port is equal to an Application Server's ORB port. You can specify the nodeagent's ORB port in the 'ORB port' field. You can specify the new ORB port for the Application Server in the 'Appplication Server's ORB Port' field.

ATTENTION: Skip this step if the ports are already defined in the TCP/IP profile.

4. You must first complete the customization of a stand-alone application server and the customization of a deployment manager server before starting these instructions. Also, ensure that the deployment manager server has been started before starting these instructions.

```
+-----------+-------------------------------------------------------+
| --------  | If your system is busy, you may want to include a rule |
+-----------+ in your WLM policy that OMVS work for job BBON001      |
| Done:     | (such as the postinstaller step) is to run in a service |
|           | class with a high service objective.                   |
| By:       |                                                        |
+-----------+-------------------------------------------------------+
| BBOWADDN  | User ID requirement: WSADMIN                           |
+-----------+                                                        |
|           | If you choose the start the node agent automatically,  |
| Done:     | the user ID used to run BBOWADDN will also need the    |
|           | authority to issue the MVS START command.              |
|           |                                                        |
|           |                                                        |
|           | Add the application server(s) associated with the stand-|
|           | alone application server node to the deployment manager's|
|           | cell.                                                  |
| By:       |                                                        |
|           | ATTENTION: Before you run this job (BBOWADDN) to       |
|           | federate the stand-alone Application Server node, you  |
|           | must have started the stand-alone Application server   |
|           | that you are federating at least once so the applyPTF  |
|           | step runs.  Otherwise, the BBOWADDN job will fail with |
|           | an error such as:                                      |
|           |   java.lang.NullPointerException at                    |
|           |   com.ibm.ws.management.tools.NodeFederationUtility    |
|           |     .createNDProductFile(NodeFederationUtility.java:1929)|
|           |                                                        |
|           | Note: Run this job only once for each node that you    |
|           | federate, regardless of how many application servers the|
|           | node contains.                                         |
|           |                                                          |
|           | RESULT: Upon successful completion of this job, you will |
|           | see the following message in SYSPRINT:                   |
|           |                                                          |
|           |   ADMU0003I: Node nodename has been successfully    |
|           |     federated.                                           |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | Start the node agent server                              |
+-----------+                                                          |
| Done:     | Note: The node agent is automatically started by the node|
|           | federation process.  This step is information, for       |
|           | starting the node agent at other times.                  |
| By:       |                                                          |
|           | If your Application Server proc name is XXXXXXXX         |
|           | your Application Server node name (short)                |
|           | is YYYYYYYY and your Deployment Manager cell             |
|           | name (short) is ZZZZZZZZ, issue the MVS command:         |
|           |                                                          |
|           |   START XXXXXXXX,JOBNAME=BBON001,                        |
|           |         ENV=ZZZZZZZZ.YYYYYYYY.BBON001                    |
|           |                                                          |
|           | This command starts the node agent server.               |
|           |                                                          |
|           | RESULT: The following message appears on the console and |
|           | in the job log of BBON001.                               |
|           |                                                          |
|           |   BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR    |
|           |   z/OS CONTROL PROCESS BBON001.                          |
|           |                                                          |
+-----------+----------------------------------------------------------+
| --------  | Start the Application Server (optional)                  |
+-----------+                                                          |
| Done:     | Note: You must start the node agent before you start the |
|           | Application Server, if the node agent has not already    |
|           | been started.                                            |
|           |                                                          |
| By:       | Example: If your Application Server proc name is         |
|           | XXXXXXXX, your Application Server node name (short) is   |
|           | YYYYYYYY your server short name is BBOS001, and your     |
|           | Deployment Manager cell name (short) is ZZZZZZZZ, issue  |
|           | the MVS command                                          |
|           |                                                          |
|           |   START XXXXXXXX,JOBNAME=BBOS001,                        |
|           |         ENV=ZZZZZZZZ.YYYYYYYY.BBOS001                    |
|           |                                                          |
|           | This command starts the Application Server.              |
|           |                                                          |
|           | RESULT: The following message appears on the console and |
|           | in the job log of BBOS001.                               |
|           |                                                          |
|           |   BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR    |
|           |   z/OS CONTROL PROCESS BBOS001                           |
|           |                                                          |
+-----------+----------------------------------------------------------+
```

Working with your new federated server node:

Once you have successfully federated an application server node, perform the following tasks:

- Check the default host alias list and all other cell-level documents to see if any need to be added in support of the applications and application servers on the newly federated node. Cell-level documents are **NOT** automatically updated by the federation process.
- Remove the location service daemon port definitions for the stand-alone application server cell because these are not used after federation.

Note that Web server configurations in an unmanaged node in a stand-alone application server cell are not migrated as part of federation. Use the administrative console or scripting to add new Web Server definitions to a Network Deployment cell.

Once these tasks are accomplished, a federated application server node is just like any other application server node. The primary difference is that it already has an application server and applications if they were federated as well. See “Working with your new managed server node” on page 213 for further information.

Configuring with the Profile Management tool

Use this task to configure WebSphere Application Server for z/OS application serving environments for your z/OS target systems using the Profile Management tool.

- Choose a z/OS target system and complete the steps in “Installing the product and additional software” on page 40 and “Preparing the base operating system” on page 41.
- Check that an FTP server is up and running on the z/OS target system.
- Choose a WebSphere Application Server for z/OS configuration (practice, stand-alone or Network Deployment cell) and complete the steps in “Planning for product configuration” on page 51.

Configuring a WebSphere Application Server for z/OS application serving environment consists of setting up the WebSphere Application Server for z/OS configuration directory for the environment, making any required changes to the z/OS target system that pertain to the particular application serving environment, and starting the new environment to verify the configuration. Configuring these application serving environments after product installation requires a fair amount of planning and coordination. If you have not previously configured WebSphere Application Server for z/OS, you should configure a ″practice″ stand-alone application server using the default options then proceed to configure the actual product configuration that you want. See “Using the Profile Management tool” on page 225 and “Building a practice WebSphere Application Server for z/OS cell” on page 84 for more information.

If you have already created a Network Deployment cell, follow the instructions in this section to expand the cell by creating a new managed node or federating an existing stand-alone application server node into the Network Deployment cell.

WebSphere Application Server for z/OS application serving environment nodes are created using the Profile Management Tool. Once a node is configured and running, make further changes using the Web-based administrative console or scripting.

After you have installed the WebSphere Application Server for z/OS product, prepared your z/OS target systems, and planned your new application server environment, perform the tasks in this section to configure and start the application server environment.

1. Install the Application Server Toolkit (AST) by following the instructions in “Installing the Application Server Toolkit” on page 225.
2. Review the procedures for the tool in “Using the Profile Management tool” on page 225.
3. Follow the directions for the type of application server environment you want to configure:
   - “Creating a stand-alone application server cell” on page 227
   - “Creating a network deployment cell” on page 228
   - “Creating a Network Deployment cell with an application server” on page 228
   - “Creating a managed node” on page 242
   - “Federating a stand-alone application server into a network deployment cell” on page 242

Once your application serving environment is up and running, you can install and test the applications. You might also want to configure your Web servers to interact with WebSphere Application Server for z/OS.

Using the Profile Management tool

This article provides general information on starting and using the Profile Management tool. See the instructions for each customization task for directions on using the Profile Management tool to perform a particular task.

The WebSphere Application Server for z/OS Profile Management tool (PMT) is a dialog tool that runs on the WebSphere Application Server Toolkit.

The WebSphere Application Server for z/OS Profile Management Tool (PMT) is an Eclipse plug-in, running under the Application Server Toolkit, that you use for the initial setup of WebSphere Application Server for z/OS cells and nodes. The Profile Management Tool itself does not create the cells and nodes; instead, it creates batch jobs, scripts, and data files that you can use to perform WebSphere Application Server for z/OS customization tasks. These jobs, scripts, and data files form a **customization definition** on your workstation, which is then uploaded to z/OS and used for customization.

Note: In WebSphere Application Server for z/OS, you use the ISPF-based Customization Dialog or the Eclipse-based Profile Management Tool (and the jobs they generate) to create new cells and nodes. Once you have created a stand-alone application server or Network Deployment cell, however, you use the WebSphere Application Server for z/OS administrative console or scripting to administer it.

The Profile Management tool is intended for use by a systems programmer or WebSphere Application Server for z/OS administrator who is familiar with the z/OS target system on which the resulting WebSphere Application Server for z/OS cells and nodes will run.

The PMT uses response files to hold the various values used to create WebSphere Application Server for z/OS customization jobs, scripts and files. These response files remain on the workstation where the Profile Management tool is run.

Follow these steps to use the Profile Management tool:

1. To install the Application Server Toolkit (AST), which includes the Profile Management tool, refer to "Installing the Application Server Toolkit" for more information.
2. To start the Profile Management tool, see the instructions in "Starting the Profile Management tool" on page 226 in order to use the tool.
3. To create a customization definition, follow the steps in "Creating the customization definition" on page 226.
4. To review a customization definition, follow the steps in "Reviewing the customization definition" on page 226.
5. To upload customization jobs to the target z/OS system, follow the instructions in "Uploading the customization jobs to z/OS" on page 227.
6. To delete a customization definition, select the customization definition and click **Delete**.

Installing the Application Server Toolkit:

This article leads you through the tasks involved in installing the WebSphere Application Server Toolkit.

The Application Server Toolkit is available on CD-ROM in the WebSphere Application Server CD-ROM package.

Follow the steps below to install the WebSphere Application Server toolkit.

1. Install the Application Server Toolkit. Before you can work with the Profile Management tool, install the Application Server Toolkit. The Application Server Toolkit is available on CD-ROM in the WebSphere Application Server CD-ROM package. To install Application Server Toolkit, follow the installation instructions that are on its CD-ROM. Install the Application Server Toolkit on the Windows® platform.

   For more information refer to the section ″Installing the Application Server Toolkit″ in the *Installing your application serving environment* PDF.

2. Start working with the Profile Management tool. After the Application Server Toolkit is installed, you are ready to start working with the Profile Management tool.

Starting the Profile Management tool:

This article leads you through the tasks involved in starting the Application Server Toolkit.

Install the Application Server Toolkit (AST) on your workstation. Refer to *Installing the Application Server Toolkit* for instructions.

Follow the steps below:

1. Launch the Application Server Toolkit: On Windows: From the **Start** button, select **Programs** or **All Programs**. Select **IBM WebSphere** and then select **Application Server Toolkit Version 6.1**. Finally, select **Application Server Toolkit**.
2. Select a workspace, unless you have previously selected a default workspace.
3. To start the Profile Management tool, click on **Window** and release the mouse button on **Preferences**. From the navigation bar on the left, find **WebSphere for z/OS**, located under the Server section. Select **WebSphere for z/OS**
4. The table titled ″WebSphere for z/OS customization definition″ shows the customization definitions in the current workspace. You are now ready to select and work with a customization definition.

Creating the customization definition:

This article leads you through the tasks involved in creating the customization profile.

Install the Application Server toolkit and perform any planning indicated for the customization task you have chosen.

A customization definition consists of a set of files on your workstation that is uploaded to z/OS and used to perform customization tasks.

1. Start the Profile Management tool. Refer to “Starting the Profile Management tool” to start the tool.
2. From the WebSphere for z/OS Customization Profile menu, select the **create** button. Read the Welcome page and select **Next**.
3. Select the type of customization definition you wish to create, and click **Next**.
4. The Profile Management tool will generate a name for this customization definition, and choose a directory to store the customization definition files. Change these values if desired. Also, if you need to use an existing response file to pre-load customization values, enter the path to the response file where indicated. When you have finished, click **Next**.
5. Enter the high level qualifiers for the pair of z/OS data sets that will contain the customization jobs and files for this customization task, when the definition is uploaded to z/OS. When you have finished, click **Next**.
6. Proceed through the panels as displayed, using the **Back** and **Next** buttons. Change the customization values where necessary. Each field has help information that can be displayed by hovering the cursor over the field. Use the **Cancel** button to leave the profile creation process without creating a definition.
7. When you have successfully filled in all the customization panels for this customization profile type, the Profile Management tool will display the definition type, location, and name. Click **Create** to build the customization definition jobs and files on your workstation.
8. Click **Finish** to return to the WebSphere for z/OS Customization Definition menu.

Reviewing the customization definition:

This article explains how to work with a customization definition that you have created in the Profile Management tool.

In order to review the customization definition, you will need to have the Application Server Toolkit installed.

Follow these steps to review profiles that you have created with the Profile Management tool.

1. Refer to “Starting the Profile Management tool” on page 226 for instructions.
2. From the WebSphere for z/OS Customization Definition menu, select one of the customization profiles you have created.
3. If you want to view the selected profile, click **View**. You will be presented with a Customization Profile Information window with three tabs:
   a. The **Summary** tab displays the definition name and type, the location on your workstation of various files related to the customization definition, and the z/OS data set names that will be used for the customization jobs and files when they are uploaded.
   b. The **Instructions** tab displays the generated task instructions you will follow after the jobs and files are uploaded to z/OS.
   c. The **Response File** tab displays the customization values stored in the response file.
4. If you want to update or change the selected profile, click **Regen**.
5. If you need to delete the selected profile, click **Delete**.
6. When you have finished reviewing the profile, click **Finish** or **Cancel** to return to the *WebSphere for z/OS customization definition* panel.

Uploading the customization jobs to z/OS:

This article explains how to upload jobs and files associated with a customization definition to z/OS.

In order to upload the customization definition, you will need to have the Application Server Toolkit installed on your workstation.

Follow the steps below to upload to z/OS.

1. Refer to “Starting the Profile Management tool” on page 226 for details in starting the tool.
2. From the WebSphere for z/OS Customization Definition menu, select the customized definition you want uploaded to z/OS by highlighting the line.
3. Press the **Upload** button to activate the step.
4. On the next dialog menu, you will be prompted to select the target z/OS system, along with your userid and password. The tool will list the designated partitioned datasets where the profiles will be stored. If you are uploading to a system different from the system where the jobs will be run, then you need to target a volume which is shared with that system.
5. Indicate whether the target datasets need to be allocated on z/OS, optionally specify the volume and unit type.
6. Press the **Upload** button to upload the files to z/OS. Wait until the upload step has completed. Once the upload has completed, you will need to press the **OK** button on the popup window.
7. Press the **Finish** button to end the Profile Management tool.

Creating a stand-alone application server cell

This article leads you through the tasks involved in setting up a WebSphere Application Server for z/OS stand-alone application server environment using the Profile Management tool.

Have available copies of the worksheets that you completed as a part of “Planning for a stand-alone application server cell” on page 86 and “Planning for product configuration” on page 51.

Follow the steps below to set up a new WebSphere Application Server for z/OS stand-alone application server cell.

1. Create a customization profile for the stand-alone application server.

    a. Follow the instructions in “Creating the customization definition” on page 226.
    b. Select ″z/OS Application Server″ for application environment type.
    c. Fill in the prompts using the values from your “Customization Dialog worksheet: Stand-alone application server cell” on page 96.
2. Review the stand-alone application server definition to make sure all the values are correct. Refer to the “Reviewing the customization definition” on page 226 for instructions.
3. Upload the customization jobs and scripts to the target z/OS system. Refer to “Uploading the customization jobs to z/OS” on page 227 for instructions.
4. Follow the instructions in the BBOSSINS member of the CNTL data set you uploaded. See “Following the generated customization instructions: Stand-alone application server cell” on page 163 for more details.

When you have successfully completed the steps in the generated instructions. The new stand-alone application server is up and running on the chosen z/OS system. See “Working with your new server” on page 174 for more information.

You can now deploy and test applications on your new stand-alone application server.

Creating a network deployment cell

This article leads you through the tasks involved in setting up a WebSphere Application Server for z/OS network deployment cell using the Profile Management tool.

Have available copies of the worksheets that you completed as a part of “Planning for a Network Deployment cell” on page 100 and “Planning for product configuration” on page 51.

Perform this task to set up a new WebSphere Application Server for z/OS Network Deployment cell.

1. Create a customization definition for the network deployment cell.
    a. Follow the instructions in “Creating the customization definition” on page 226.
    b. Select ″z/OS deployment manager″ for application environment type.
    c. Fill in the prompts using the values from your “Customization Dialog worksheet: Network Deployment cell” on page 108.
2. Review the network deployment definition to make sure all the values are correct. Refer to the “Reviewing the customization definition” on page 226 for instructions.
3. Upload the customization jobs and scripts to the target z/OS system. Refer to “Uploading the customization jobs to z/OS” on page 227 for instructions.
4. Follow the instructions in the BBOCCINS member of the CNTL data set you uploaded. Refer to “Following the generated customization instructions: Network Deployment cell” on page 185 for more details.

When you have successfully completed the steps in the generated instructions the task is finished. The new network deployment cell is up and running on the chosen z/OS system. See “Working with your new deployment manager” on page 195 for more information.

Add application server nodes to your cell using one of two methods:

- Create a new managed node using the Profile Management tool and add application servers to it using the administrative console or scripting.
- Federate existing stand-alone application servers into your network deployment cell to create managed nodes with application servers.

Creating a Network Deployment cell with an application server

This article leads you through the tasks involved in setting up a WebSphere Application Server for z/OS Network Deployment cell including an initial application server, using the Profile Management tool.

Have available copies of the worksheets that you completed as a part of “Planning for a Network Deployment cell” on page 100 and “Planning for product configuration” on page 51.

Perform this task to set up a new WebSphere Application Server for z/OS Network Deployment cell (deployment manager and federated application server). These steps will lead you through the process.

1. Create a customization definition for the network deployment manager.
   a. Follow the instructions in “Creating the customization definition” on page 226.
   b. Select ″z/OS cell (deployment manager and an application server)″ for application environment type.
   c. Fill in the prompts using the values from your “Customization Dialog worksheet: Network Deployment cell with an Application Server” on page 237 and “Customization Dialog variables: Network Deployment cell with an Application Server.”
2. Review the Network Deployment cell definition to make sure all the values are correct. Refer to the “Reviewing the customization definition” on page 226 for instructions.
3. Upload the customization jobs and scripts to the target z/OS system. Refer to “Uploading the customization jobs to z/OS” on page 227 for instructions.
4. Follow the instructions in the BBODMINS member of the CNTL data set you uploaded. See “Following the generated customization instructions: Network Deployment cell” on page 185 for more details.

When you have successfully completed the steps in the generated instructions the task is complete. The new Network Deployment cell is up and running on the chosen z/OS system. See “Working with your new deployment manager” on page 195 for more information.

Add additional application server nodes to your cell using one of two methods:

- Create a new managed node using the Profile Management tool and add application servers to it using the administrative console or scripting.
- Federate existing stand-alone application servers into your network deployment cell to create managed nodes with application servers.

Customization Dialog variables: Network Deployment cell with an Application Server: This article lists definitions for the terms you will come across in the WebSphere Application Server for z/OS Customization Dialog and the Profile Management tool..

The WebSphere Application Server for z/OS runtime requires four stand-alone cell servers: application server, deployment manager, node agent, and location service daemon. The panels corresponding to the following tables set up the names, network configuration, start procedures, and user IDs for a Network Deployment cell with an application server.

High Level Qualifier

Configure Common Groups and Users

System Locations

This section identifies the target z/OS system on which you will configure the deployment manager for the Network Deployment cell, along with system data set names.

System name
: The system name for the target z/OS system on which you will configure WebSphere Application Server for z/OS.

Sysplex name
: The sysplex name for the target z/OS system on which you will configure WebSphere Application Server for z/OS.

Tip: If you are not sure what the system name (&SYSNAME) and sysplex name (&SYSPLEX) are, use the console command `D SYMBOLS` on the target z/OS system to display them.

For the following, specify the fully qualified data set names without quotes.

Rule: You can specify up to 44 characters for the data set names.

PROCLIB
An existing procedure library where the WebSphere Application Server for z/OS cataloged procedures are added.

WebSphere product data sets high level qualifier

WebSphere Application Server for z/OS product data sets

Specify the following WebSphere Application Server for z/OS libraries so they can be accessed by the customized jobstreams the dialog produces. These data sets must be cataloged. See the section ″Product data sets″ in the *Installing your application serving environment* PDF for more information.

Run WebSphere Application Server from STEPLIB (Y/N)?
Specifies whether to load WebSphere Application Server for z/OS load modules from STEPLIB (″Y″) or from the link pack area and link list (″N″).

See the section ″Link pack area, link list, and STEPLIB″ in the *Installing your application serving environment* PDF for more information.

Note: Specify ″Y″ if you have another instance of WebSphere Application Server for z/OS (Version 4 or later) in the system link pack area or link list.

SBBOLPA
WebSphere Application Server for z/OS load module library. It has modules that should go into LPA or the location service daemon STEPLIB.

SBBOLOAD
WebSphere Application Server for z/OS 31-bit load module library. It has members that should go into the link list or LPA, or into STEPLIB.

SBBGLOAD
WebSphere Application Server for z/OS 64-bit load module library. It has members that should go into the link list or LPA, or into STEPLIB.

SBBOLD2
WebSphere Application Server for z/OS load module library that you installed through SMP/E. It has members that should go into the link list, or into STEPLIB. **DO NOT** place them in LPA.

SBBOEXEC
WebSphere Application Server for z/OS CLIST library.

SBBOMSG
SBBOMSG WebSphere Application Server for z/OS message skeletons for language translation.

Locations of File System Resident components

WebSphere Application Server product directory
The name of the directory where WebSphere Application Server for z/OS files reside after installation.

See the section ″Product file system″ in the *Installing your application serving environment* PDF for more information.

WebSphere Configuration File System

This section defines the WebSphere configuration HFS that you will use for the deployment manager.See the section ″Configuration file system″ in the *Installing your application serving environment* PDF for more information.

Mount point
Read/write HFS directory mount point where application data and environment files are written. The customization process creates this mount point if it does not already exist.

Name Hierarchical File System data set you will create and mount at the above mount point.

Rule: You can specify up to 44 characters for the data set name.

Volume, or '*' for SMS
Specify either the DASD volume serial number to contain the above data set or ″*″ to let SMS select a volume. Using ″*″ requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle data set allocation automatically, list the volume explicitly.

Primary allocation in cylinders
Initial size allocation in cylinders for the above data set.

Recommendation: The minimum suggested size is 420 cylinders (3390).

Secondary allocation in cylinders
Size of each secondary extent in cylinders.

Recommendation: The minimum suggested size is 100 cylinders.

Use zSeries file system or HFS

WebSphere error log stream

This section defines the System Logger log stream that you will use for WebSphere Application Server for z/OS error logging if TRACEBUFFLOC=BUFFER. Having the error log in the log stream is optional. By default, the error log is written to JES SYSOUT.

Name Name of the WebSphere error log stream you will create.

Rules:
- The name must be 26 or fewer characters.
- Do NOT put quotes around the name.

Ctrace Writer Definitions

WebSphere application Server for z/OS uses component trace (CTRACE) to capture and to display trace data in trace data sets. WebSphere Application Server for z/OS identifies itself to CTRACE with the *cell short name.*

Trace Parmlib member suffix
Value that is appended to CTIBBO to form the member name for the Trace parmlib member.

Server customization

During this customization task, you will create a cell configuration, a deployment manager node and server, and a location service daemon. The panels corresponding to the following tables set up the names, network configuration, start procedures, and user IDs for a deployment manager server.

Rule: In the following customization, names must be eight or fewer characters unless otherwise specified.

Cell name (short)
Name that identifies the cell to z/OS facilities such as SAF.

Rules:
- Name must be eight or fewer characters and all uppercase.
- Name must be unique among all other cells in the sysplex.

Cell name (long)
Primary external identification of this WebSphere Application Server for z/OS cell. This name identifies the cell as displayed through the administrative console.

Rules:

- Name must be 50 or fewer characters and can be of mixed case.
- Name must be unique among all other cells in the sysplex.

Deployment manager node name (short)

Name that identifies the deployment manager node to z/OS facilities such as SAF.

Rules:

- Name must be eight or fewer characters and all uppercase.
- Name must be unique within the cell.

Deployment manager node name (long)

Primary external identification of this WebSphere Application Server for z/OS deployment manager node. This name identifies the node as displayed through the administrative console.

Rules:

- Name must be 50 or fewer characters and can be of mixed case.
- Name must be unique within the cell.

Deployment Manager Server name (short)

This value identifies the deployment manager server to z/OS facilities such as SAF.

Note: The server short name is used as the server JOBNAME.

Rule: Name must usually contain seven or fewer all-uppercase characters. To change the jobname later to an eight-character value, you must follow the steps outlined in Converting a 7-character server short name to 8 characters.

Deployment Manager Server name (long)

Name of the application server and the primary external identification of this WebSphere Application Server for z/OS server. This name identifies the server as displayed through the administrative console. The server name has a fixed name (long) of ″dmgr″.

Deployment Manager Cluster transition name

WLM APPLENV name for this server.

Note: The deployment manager is not clusterable, so this value never actually becomes the cluster short name of this server's cluster. However, like an application server, the deployment manager still needs an APPLENV, so the cluster transition name is used for this purpose.

Rule: Name must be eight or fewer characters and all uppercase.

Application Server Cluster Transition name

Node Agent and Application Server node names (short)

Node Agent and Application Server node names (long)

Node Agent Server name (short)

Node Agent Server name (long)

Application Server name (short)

Application Server name (long)

Deployment manager definitions

Rule: In the following definitions, names must be eight or fewer characters unless otherwise specified.

Controller Information

Jobname

The jobname, specified in the MVS START command JOBNAME parameter, associated with the deployment manager controller. This is the same as the server short name and it cannot be changed through the Customization Dialog.

Procedure name

Name of member in your procedure library to start the deployment manager controller.

Rule: Name must be seven or fewer characters.

User ID

The user ID associated with the deployment manager controller.

Note: If you are using a non-IBM security system, the user ID might have to match the procedure name. Please refer to your security system's documentation.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Servant Information

Jobname

The jobname used by WLM to start the deployment manager servant. This is set to the server short name followed by the letter ″S″, and it cannot be changed through the Customization Dialog.

Procedure name

Name of member in your procedure library to start the deployment manager servant.

Rule: Name must be seven or fewer characters.

User ID

The user ID associated with the deployment manager servant.

Note: If you are using a non-IBM security system, the user ID might have to match the procedure name. Please refer to your security system's documentation.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

Deployment manager TCP/IP information

Note: Do not choose port values already in use.

Node host name

IP name or address of the system on which the server is configured. This value is used by other WebSphere Application Server for z/OS functions to connect to this server.

Note: The node host name must always resolve to an IP stack on the system where the application server runs. (You can either have multiple IP stacks on a given MVS image and have the deployment manager and stand-alone application server tied to separate host names, or you can associate them with the same node host name.) The node host name cannot be a DVIPA or a DNS name that, in any other way, causes the direction of requests to more than one system.

SOAP JMX Connector port

Port number for the JMX HTTP connection to this server based on the SOAP protocol. JMX is used for remote administrative functions, such as invoking scripts through wsadmin.sh.

Rule: Value cannot be 0.

Cell Discovery Address port

Port number used by node agents to connect to this deployment manager server.

ORB Listener host name

IP address on which the server's ORB listens for incoming IIOP requests. The default is ″*″, which instructs the ORB to listen on all available IP addresses.

ORB port

Port for IIOP requests that acts as the bootstrap port for this server and also as the port through which the ORB accepts IIOP requests.

Rule: Value cannot be 0.

ORB SSL port

Port for secure IIOP requests. The default is ″0″, which allows the system to choose this port.

HTTP transport host name

IP address on which the server's Web container should listen for incoming HTTP requests. The default is ″*″, which instructs the Web container to listen on all available IP addresses.

Note: The ″transport host name″ becomes the ″hostname″ in the virtualhosts.xml file, which makes setting a specific IP address here less than ideal because, if you do so, you are restricting yourself to that IP address until you go into the administrative console and add another virtual host.

Administrative console port

Port for HTTP requests to the administrative console.

Administrative console secure port

Port for secure HTTP requests to the administrative console.

High Availability Manager communication port

Port on which the High Availability Manager listens.

Location service daemon

The location service daemon is the initial point of client contact in WebSphere Application Server for z/OS. The server contains the CORBA-based location service agent, which places sessions in a cell. All RMI/IIOP IORs (for example, for enterprise beans) establish connections to the location service daemon first, then forward them to the target application server.

Daemon home directory

Directory in which the location service daemon resides. This is set to the configuration HFS mount point/Daemon and cannot be changed.

Daemon jobname

Specifies the jobname of the location service daemon, specified in the JOBNAME parameter of the MVS start command used to start the location service daemon.

Note: The same daemon jobname is used on all z/OS systems hosting the cell. Therefore, this name is generic and has no system indicator. When the deployment manager's controller started, its location service daemon was started automatically using this particular daemon jobname value.

Caution: When configuring a second cell, ensure you change the daemon jobname from the default or value you used for the first cell.

Note: A server automatically starts the location service daemon if it is not already running.

Procedure name

Name of the member in your procedure library to start the location service daemon.

Rule: Name must be seven or fewer characters.

User ID

The user ID associated with the location service daemon.

UID The user identifier associated with this user ID.

Rule: UIDs must be unique numbers, between 1 and 2,147,483,647, within the system.

IP Name

The fully qualified IP name, registered with the Domain Name Server (DNS), that the location service daemon uses.

Adhere to the following regarding your location service daemon IP name:

- It must be unique in the sysplex.
- It must not have the same value as any one system's node host name. (You can use the host name of the LPAR.)
- It should be a name that can be used in conjunction with a routing service that distributes requests among nodes in the cell (that is, systems in the sysplex).
- It should be a virtual IP address (VIPA) if you are operating in a sysplex.

Note:

- IBM recommends you use z/OS Sysplex Distributor by way of a ″Dynamic Virtual IP address″ (DVIPA). See related sections in the information center for more information.
- Select the IP name for the location service daemon carefully. Once chosen it is difficult to change, even in the middle of customization. This name must not be a numeric, such as, 3.7.25.43.

Daemon Listen IP
The default value is *.

Rule: The default is * or a numeric IP address.

Port The port number on which the location service daemon listens.

Note: Select the port number for the location service daemon carefully. You can choose any port you want; but, once chosen, it is difficult to change, even in the middle of customization.

SSL port
The port number on which the location service daemon listens for SSL connections.

Register daemon with WLM DNS
If you use the WLM DNS (connection optimization), you must select ″Y″ to register your location service daemon with it. Otherwise, select ″N.″

Note: Only one location service daemon per LPAR can register its domain name with WLM DNS. If you have multiple cells in the same LPAR and register one location service daemon and then a second, the second will fail to start.

SSL customization

If you plan to enable Administrative Security at some point, as is recommended, fill in the following SSL values:

Certificate authority keylabel
The name of the key label that identifies the certificate authority (CA) to be used in generating server certificates.

Generate authority certificate
Select ″Y″ to generate a new CA certificate. Select ″N″ to have an existing CA certificate generate server certificates.

Expiration date for certificates
The expiration date used for any X509 Certificate Authority certificates, as well as the expiration date for the personal certificates generated for WebSphere Application Server for z/OS servers. You must specify this even if you selected ″N″ for ″Generate Certificate Authority (CA) certificate.″

Default RACF keyring name
The default name given to the RACF keyring used by WebSphere Application Server for z/OS. The keyring names created for repertoires are all the same within a cell.

Enable SSL on location service daemon
Select ″Y″ if you want to support secure communications using Inter-ORB Request Protocol (IIOP) to the location service daemon using SSL. If you specify ″Y″, a RACF keyring will be generated for the location service daemon to use.

Security Customization

During the initial setup, you can choose one of the following three options for administrative security.

Option 1 - WebSphere for z/OS security
Use the z/OS system's SAF-compliant security database to define WebSphere Application Server users. The EJBROLE profile will be used to control role-based access to applications. An administrator user ID and an unauthenticated user ID will be created and defined in the security

database. Select this option if the WebSphere Application Server environment will run entirely on z/OS with a shared SAF-compliant (Local OS) user registry, or if you plan to implement an non-Local OS user registry (such as LDAP) with mapping to SAF user IDs.

Option 2 - WebSphere family security
Use a simple file-based registry to define WebSphere Application Server users. An administrator user ID will be created and defined in the file-based registry.

Option 3 - No security
Do not enable administrative security (not recommended). Your WebSphere Application Server environment will not be secured until you configure and enable security manually.

Depending on the security option you choose, there may be additional customization dialog values you need to set.

Security customization - WebSphere for z/OS security

For this security option, you must decide whether to set a security domain name, and choose an administrator user ID and an unauthenticated (guest) user ID.

Use security domain identifier in RACF profiles
Set this to Y if you wish to include a security domain name in certain SAF security checks (APPL, CBIND, EJBROLE). Enter a 1-8 character security domain name.

Administrator user ID
Administrator user ID Enter a valid SAF user ID which will become the initial cell administrator. If this user ID already exists, it must have the WebSphere Application Server configuration group for this cell as its default Unix System Services group. Also specify a valid UID for this user ID.

Unauthenticated User ID
Enter a valid SAF user ID which will associated with unauthenticated client requests. Also specify a valid UID for this user ID.

Security customization - WebSphere family security

For this security option, you must choose an administrator user ID and password.

Administrator user ID
Enter an alphanumeric user ID which you will use to log on to the administrative console and perform administrative tasks. This user ID and its password will initially be the only entry in the file-based user registry.

Administrator password
This password must not be blank.

Security Customization - no security

For this security option, there are no other choices to make. Your WebSphere Application Server environment will not be secured until you configure and enable security manually.

Web Server Configuration

If you want to create a web server definition at this point, fill in the following values.

Note: You can only have one Web server defined on a stand alone application server.

You will not be able to administer the Web server through the integrated solutions console on the stand alone server until it is federated to a managed node.

Web Server Type
Valid values: IHS, HTTPSERVER_ZOS, APACHE, IPLANET, DOMINO, IIS.

Web Server Operating System
The operating system where the Web server is located.

Web Server Name
The name used in defining the Web server in the admin console.

Web Server Host or IP Address
IP name or address of the z/OS on which the Web server is located.

Web Server Port
HTTP Port on which the Web server is listening.

Web Server Install Directory Path
/usr/lpp/internet/

Web Server Plugin Install Directory Path
(*This varies by Web server type*)

Customization Dialog worksheet: Network Deployment cell with an Application Server:

Date: ______________________

Purpose of this Network Deployment cell:
____

System name: __________________________ Sysplex name: __________________________

Print out this worksheet and use it when collecting information about the WebSphere Application Server for z/OS Customization Dialog variables. The variables and defaults are provided, along with spaces for you to fill in your own value for each.

z/OS Target Dataset High Level Qualifier

Configure Common Groups and Users

System Locations

Item	Default	Your value
System Name		
Sysplex Name		
PROCLIB	SYS1.PROCLIB	
WebSphere product data set high level qualifier		

WebSphere Application Server for z/OS product data sets

Run WebSphere Application Server from STEPLIB (Y/N)?	Y	
SBBOLPA	*was_hlq*.SBBOLPA	
SBBOLOAD	*was_hlq*.SBBOLOAD	
SBBGLOAD	*was_hlq*.SBBGLOAD	
SBBOLD2	*was_hlq*.SBBOLD2	

SBBOEXEC	*was_hlq*.SBBOEXEC	
SBBOMSG	*was_hlq*.SBBOMSG	

Locations of Resident File System Components

WebSphere Application Server product directory	/usr/lpp/zWebSphere/V6R1	

WebSphere Configuration File System

Mount point	/WebSphere/V6R1	
Name	OMVS.WAS.CONFIG.HFS	
Volume, or '*' for SMS	*	
Primary allocation in cylinders	420	
Secondary allocation in cylinders	100	
Use zSeries File Ssystem instead of HFS	Y	

WebSphere error log stream

Name	WAS.ERROR.LOG	

Component Trace

Trace Parmlib member suffix	60	

Server customization

Cell name (short)	*sysplexname*	
Cell name (long)	*sysplexname*Network	
Deployment manager node name (short)	*sysplexname*	
Deployment manager node name (long)	*sysplexname*Manager	
Deployment manager server name (short)	BBODMGR	
Deployment manager server name (long)	dmgr	(cannot change)
Node agent & application server node name (short)	*systemname*	
Node agent & application serve node name (long)	*systemname*	
Node agent server name (short)	BBON001	
Node agent server name (long)	nodeagent	
Application server name (short)	BBOS001	
Application server name (long)	server1	

Deployment manager cluster transition name	BBODMGR	
Application server cluster transition name	BBOC001	

Server customization

Deployment manager path related to mount point	DeploymentManager	
Application server path related to mount point	AppServer	
Admin asynch operations proc name	BBOW6SH	
Asynch ID	WSADMSH	
Asych UID	2504	
Install samples	Y	

Server address space information

Application Server Controller Information		
Jobname	*servershortname*	(cannot change)
Procedure name	BBO6ACR	
User ID	ASCR1	
UID	2431	
Application Server Control Region Adjunct		
Jobname	*servershortname*A	(cannot change)
Procedure name	BBO6CRA	
User ID	ASCRA1	
UID	2433	
Application Server Servant Information		
Jobname	*servershortname*S	(cannot change)
Procedure name	BBO6ASR	
User ID	ASSR1	
UID	2432	

Deployment manager definitions

Deployment Manager Controller Information		
Jobname	*deploymentmanagershortname*	(cannot change)
Procedure name	BBO6DCR	
User ID	DMCR1	
UID	2421	
Deployment Manager Servant Information		
Jobname	*deploymentmanagershortname*S	(cannot change)
Procedure name	BBO6DSR	

User ID	DMSR1	
UID	2422	

Deployment manager TCP/IP information

Node host name	(blank)	
SOAP JMX connector port	8879	
Cell Discovery Address port	7277	
ORB Listener host name	*	
ORB port	9809	
ORB SSL port	0	
HTTP transport host name	*	
Administrative console port	9060	
Administrative secure console port	9043	
High availability manager communication port	9352	

Node Agent TCP/IP information

JMX SOAP Connector port	9360	
ORB port	2810	
ORB SSL port	0	
High Availability Manager Communication Port	9354	
Node Discovery Port	7272	
Node Multicast Discovery Port	5000	

Application Server TCP/IP information

JMX SOAP Connector port	8880	
ORB port	2809	
ORB SSL port	0	
HTTP port	9080	
HTTP SSL port	9443	
High Availability Manager Communication Port	9353	
Service Integration port	7276	
Service Integration Secure port	7286	
Service Integration MQ Interoperability port	5558	
Service Integration MQ Interoperability Secure port	5578	
Session Initiation Protocol	5060	
Session initiation Secure Protocol	5061	

Location service daemon

Daemon home directory	*configmountpoint*/Daemon	(cannot change)
Daemon jobname	BBODMNC	
Procedure name	BBO6DMN	
User ID	WSDMNCR1	
UID	2411	
IP name	*nodehostname*	
Port	5755	
SSL port	5756	
Register daemon with WLM DNS	N	

SSL Customization

Certificate Authority Keylabel	WebSphereCA
Generate Certificate Authority (CA) certificate	Y
Expiration date for CA authority	2010/12/31
Default RACF keyring name	WASkeyring.*node name*
Enable SSL on location service daemon	Y

Security configuration

WebSphere for z/OS security

Use security domain identifier in RACF profiles:	N
Security domain identifier.	
WebSphere Application Server Administrator Information	
User ID	WSADMIN
UID	2403
WebSphere Application Server Unauthenticated User Information	
User ID	WSGUEST
UID	2402

WebSphere family security

User ID.: WSADMIN	WSADMIN
Password:	
Confirm Password:	Insert link to security section.

Web Server Configuration

Web Server Type	HTTPSERVER - Z/OS	
Web Server Operating System	os390	
Web Server Name	webserver1	

Web Server Host or IP Address	*nodehostname*	
Web Server Port	80	
Web Server Install Directory Path	(varies by type)	
Web Server Plugin Install Directory Path	(varies by type)	

Job statement definition

(ACCTNO,ROOM),'USERID',CLASS=A,REGION=OM
//*
//*
//*

Name of Network Deployment cell configuration variable data set: ____

Creating a managed node

This article leads you through the tasks involved in creating a customization definition for a managed server node using the Profile Management tool.

Perform this task to create a customization definition for the managed node.

1. Create a customization definition for the managed node.
   a. Follow the instructions in "Creating the customization definition" on page 226,
   b. Select ″z/OS managed (custom) node″ for application environment type.
   c. Fill in the prompts using the values from your "Customization Dialog worksheet: Managed node" on page 118.
2. Review the managed node definition to make sure all the values are correct. Refer to the "Reviewing the customization definition" on page 226 for instructions.
3. Upload the customization jobs and scripts to the target z/OS system. Refer to "Uploading the customization jobs to z/OS" on page 227 for instructions.
4. Follow the instructions in the BBOMNINS member of the CNTL data set you uploaded. See "Following the generated customization instructions: Managed node" on page 204 for more details.

When you have successfully completed the steps in the generated instructions this task is complete. The new managed node is up and running on the chosen z/OS system. See "Working with your new managed server node" on page 213 for more information.

Federating a stand-alone application server into a network deployment cell

This article leads you through the tasks involved in federating a WebSphere Application Server for z/OS stand-alone application server into a network deployment cell using the Profile Management tool.

Perform this task to federate an existing WebSphere Application Server for z/OS stand-alone application server into a Network Deployment cell.

1. Create a customization definition for federating the stand-alone application server into a network deployment cell.
   a. Follow the instructions in "Creating the customization definition" on page 226.
   b. Select ″z/OS federate an Application Server″ for application environment type.
   c. Fill in the prompts using the values from your "Customization Dialog worksheet: Federating an application server node" on page 124.

2. Review the definition to make sure all the values are correct. Refer to the "Reviewing the customization definition" on page 226 for instructions.
3. Upload the customization jobs and scripts to the target z/OS system. Refer to "Uploading the customization jobs to z/OS" on page 227 for instructions.
4. Follow the instructions in the BBOANINS member of the CNTL data set you uploaded. See "Following the generated customization instructions: Federated application server node" on page 220 for more details.

When you have successfully completed the steps in the generated instructions this task is complete. The new federated application server node is up and running on the chosen z/OS system. See "Working with your new federated server node" on page 223 for more information.

Using the installation verification test

You initially run the installation verification test (IVT), which verifies that WebSphere Application Server is configured correctly for your system, during ISPF customization of each of your systems. If you want to run the IVT at a time other than during initial customization, however, there are two methods from which you can choose.

Note: These options are now available when you are running a stand-alone application server configuration as well as after federating an application server.

Select either method to invoke the IVT:

- "Running the installation verification test with a job" on page 176
- "Running the installation verification test from a command line" on page 176

Running the installation verification test with a job

The application server must be running.

Follow these steps to run the installation verification test (IVT) using the BBOWIVT job.

1. Verify that the application server is running.
2. Confirm that the ivtApp application is installed and started.
3. Submit the job BBOWIVT.

The IVT runs a series of verification tests and reports pass or fail status for each in the messages generated by the BBOWIVT job. The output is written to standard output and to the `WAS_HOME/profiles/default/logs/ivtClient.log` file.

Running the installation verification test from a command line

The application server must be running.

Follow these steps to run the installation verification test (IVT) from a command line.

1. Verify that the application server is running.
2. Confirm that the ivtApp application is installed and started.
3. From a command line, navigate to the `WAS_HOME/bin` directory.
4. Issue the following command:

   ```
   ivt.sh server_name profile_name -p port_number [-host host_name]
   ```

   where

   - *server_name* is the short name of the server.
   - *profile_name* is the name of the profile.
   - -p *port_number* is an argument that specifies the port number.

- `-host` *`host_name`* is an optional argument that specifies the host name. If you do not specify a host name, the program will use the host-name value that is set in your TCP/IP hosts file.

  Example:

  ```
  /WebSphere/V6R0/AppServer/bin> ivt.sh serverj default -p 9080 -host myhost
  ```

The IVT runs a series of verification tests and reports pass or fail status for each in the messages generated by the BBOWIVT job. The output is written to standard output and to the `WAS_HOME/profiles/default/logs/ivtClient.log` file.

Applying product maintenance

Maintaining WebSphere Application Server requires an in-depth knowledge of MVS service techniques and the product service stream. This is especially important in a high yield production environment where availability is of the utmost importance. Therefore, planning service and knowing how to best apply service are essential skills.

Contact the IBM Software Support Center for information about preventive service planning (PSP) upgrades for WebSphere Application Server for z/OS. For more information about PSP upgrades, see the *WebSphere Application Server for z/OS: Program Directory*. Although the *Program Directory* contains a list of required program temporary fixes (PTFs), the most current information is available from the IBM Software Support Center.

Use the following procedure whenever you want to apply a new service release to your system.

1. Decide whether to run the post-installer automatically or manually. See “Applying a service level or restoring to the previous accepted service level” on page 245 for more information.
2. Make copies of your product data sets and file system structure.
3. Ensure your deployment manager node is at the same or a later service level than the application server nodes.
4. Apply the required maintenance to the copies of your product data set and file system structure.

   Note: Notice if you receive WTOR message BBOO0286A when applying your service. This is significant because it means the service you installed contains defects that are not backwards-compatible and you must therefore run the backoutPTF.sh script if you want to backout the service later.
5. Stop the application servers and the WebSphere Application Server for z/OS location service daemon.
6. Stop CTRACE.
7. Switch to the newly serviced WebSphere Application Server for z/OS product data sets. You can do this one of two ways:
   a. Rename the new data sets to replace the old ones.
   b. Recatalog the product data sets if the names are identical, or change WebSphere Application Server for z/OS cataloged procedures to refer explicitly to the new data sets.

      Note: If you use new data set names, you need to also handle program controls, so ensure you update the program control list.

   Verify that the MVS link list and APF list refer to the newly-serviced data sets.
8. If the WebSphere Application Server for z/OS runtime is loaded into the link pack area, delete the old modules and load the new ones, or IPL the system to load the new modules into the LPA.
9. Verify that the newly-serviced product file system is correctly mounted.
10. Perform any other migration actions as instructed in PTF or APAR cover letters.
11. Start the location service daemon and application servers.

12. Complete any necessary post-install tasks. In particular, make sure that the post-installer is run (either automatically or manually) against each node, in order to update the configuration file system to the new service level. For more information, refer to “Post-install tasks to complete after using SMP/E to apply a new service level.”

You can maintain service to clients when upgrading the host cluster of WebSphere Application Server for z/OS.

If you roll back to a previous maintenance level of WebSphere Application Server, be sure to run the post-installer against each node to back out configuration file system changes before switching to the older service level. See “Post-install tasks to complete before using SMP/E to restore to the previous accepted service level” on page 247 for more information.

Applying a service level or restoring to the previous accepted service level

Service that is applied to the WebSphere Application Server for z/OS product data sets and product file systems occasionally requires corresponding changes to be made to the configuration file systems for existing application serving environments that configure at a lower service level. Most of these ″post-maintenance″ or ″post-install″ updates can be performed automatically. This is done by the post-installer.

The WebSphere Application Server for z/OS post-installer is a set of scripts that can be used to:

- Automatically detect and apply post-PTF service.
- Back out (uninstall) service from the configuration HFS when returning to an older service level.

The post-install process is performed at the node level and must run against each node’s WebSphere Application Server home directory after maintenance is applied to the product data sets and HFS, and before the node is started. The post-install process can be performed:

- automatically, by leaving the configured JCL statements in the controller cataloged procedures that start the post-installer. A console message will be displayed whenever the post-install detects service to be applied. In some cases, some post-install steps might still have to be performed manually. The post-installer will detect these situations and refuse to start the server.
- manually. You can run the post-installer yourself against each node’s WebSphere Application Server home directory, after installing maintenance and before starting the nodes.

For more information about the post-installer, and how to use it to apply or back out service to the configuration HFS, see

- “Post-install tasks to complete after using SMP/E to apply a new service level”
- “Post-install tasks to complete before using SMP/E to restore to the previous accepted service level” on page 247

Note: You might find it useful to set up a WLM rule to combat performance problems related to post-install. See Workload management (WLM) tuning tips for z/OS for more information.

Post-install tasks to complete after using SMP/E to apply a new service level

This article describes post-install tasks you complete after applying a new service level. See “Applying product maintenance” on page 244, for more information about applying service to WebSphere Application Server for z/OS.

The post-install functionality includes actions you perform the first time the server is restarted after you install service with SMP/E. This applies to WebSphere Application Server for z/OS as well as WebSphere Business Integration Server Foundation for z/OS. Depending on your system restrictions, you can choose to initialize post-install processing either automatically or manually.

Automatic mode, which is recommended, includes a new step that automatically launches the ″applyPTF.sh″ shell script in the server procedure. The applyPTF.sh script, which is located in your bin directory (WAS_HOME/bin, where WAS_HOME is the absolute path of the WebSphere runtime home directory), verifies that any pending post-install actions are properly applied before starting the server.

Manual mode is necessary only if the automatic mode does not conform to your organization standards (if multistep procedures are not allowed for example), if there is something else in your configuration that prevents the applyPTF.sh shell script from running properly, or if you are applying a particular service release that requires manual intervention. Refer to the section below, ″Running the post installer manually,″ for more information.

Post-install processing executes under the controller proc, so it executes with the identity assigned to that particular proc. While that identity has sufficient authority to perform most file system actions required on the WebSphere runtime home directory, you cannot assume it to have the appropriate authority for your applyPTF.sh shell script. Therefore, you cannot perform any post-install action that requires special authority by ″inline″ apply processing. You must manually perform any such apply processing in ″batch″ mode, running applyPTF.sh or the appropriate jobs from a user that has the needed authority to apply a particular action.

Note:

- Before you run applyPTF.sh, ensure that WebSphere Application Server for z/OS is running with code page IBM-1047. See “Preparing z/OS” on page 42 for more information.
- When you launch the shell script from the server proc, it automatically runs in ″inline″ mode and can only apply service that qualifies as ″inline.″
- If batch service is delivered, you’ll find the needed user authority noted in the service level documentation.

You can run the post-installer either automatically or manually. Follow the set of steps below that applies to your circumstance.

1. Run the post installer automatically. This is running under the authority of the WebSphere Admin ID.
   a. Classify BPXBATCH using the WLM Workload Classification Rules for OMVS work. Use the started task jobnames to classify this work into the appropriate service class. This facilitates more efficient execution in automatic mode.
2. OR, run the post installer manually.
   a. Edit your server control process procedures (for example, BBO6ACR) to remove the BPXBATCH step that invokes the applyPTF.sh script.
   b. After rolling service to each system, run batch job BBOWAPLB for each stand-alone application server node, and run BBOWAPLD for each deployment manager node, and for each managed node.

The post installer component enforces that certain actions are performed successfully and preconditions are met before applying service, otherwise, warnings or error messages result. If you launched the applyPTF.sh script from the server proc, the output is appended to the WAS_HOME/properties/service/ logs/ applyPTF.out. If you ran applyPTF.sh manually from the shell, the output goes right to stdout (on the shell from where the applyPTF.sh script was run) and WTO messages are issued. In the latter case, no output is appended to the applyPTF.log log file.

Whenever actions are performed on the runtime home directory, a log file is kept. When warnings or errors occur, the absolute path to the log file is displayed so that you can examine the details of the problem. The following is a list of common errors that might occur when the post installer (applyPTF.sh) applies service. The errors will most likely appear in the form of error codes on the console on which applyPTF.sh was issued.

The post installer ran and determined that SMP/E restored the SMP/E home directory to a previous level of service than that at which the WebSphere runtime home directory is running.

This occurs if you used SMP/E to restore to the previous accepted service level without first running the backoutPTF.sh shell script. The server will not start if it detects this condition, and post-install action halts without applying service.

The post installer issued a warning message while installing service.

The WebSphere Application Server for z/OS multiproduct PTF post installer detected warnings that were issued during the application of post-install service for the product listed. The application of service was successful, but the warning messages should be examined. The warning messages are listed in the log file in the HFS file that is specified in the message.

The post installer incurred an error and stopped processing.

The WebSphere Application Server for z/OS multiproduct PTF post installer encountered an error while installing service for the product indicated in the message. The details of the error are contained in the HFS log file that is specified in the message. When this happens, review the log and correct the error. The servers will not be permitted to start until the error is corrected.

The post installer encountered warnings while applying service.

The WebSphere Application Server for z/OS multiproduct PTF post installer detected warnings that were issued during the application of post-install service. The warnings might have been issued while post-install service was being applied for WebSphere Application Server for z/OS or any of the extension products that are installed. A BBOO0250W message should have already been issued for each product that encountered warnings when the post-install service was applied.

Reply with `Continue` to continue starting the application server. Reply with `Cancel` to cancel starting the application server. Because service has already been completed, the multiproduct PTF post installer will not run again the next time the server is restarted.

Running the post installer manually.

As mentioned above, there are reasons why an organization might require the post installer to be run manually, also known as batch mode. The post installer will not allow post install service to be applied if a node is moved to a different system other than the system where it was configured. (In this case the WTO BBOO0287A will be issued) . Running the post installer in batch mode, however, will apply post install service to a node, even if it is now located on a system other than the one where it was originally configured.

The post installer can be invoked manually by using either of the two following procedures:

- Run the batch job that was created by the customization dialog walkthrough (CNTL data set) for each node. Use BBOWAPLB for Stand alone application server nodes and BBOWAPLD for Network Deployment cell nodes.
- Invoke the shell script directly. It is stored in the /bin directory of each node. From OMVS running under the authority of the WebSphere Admin ID, issue the `./applyPTF.sh batch` shell script.

Post-install tasks to complete before using SMP/E to restore to the previous accepted service level

This article describes those post-install tasks you might need to complete before using SMP/E to restore to the previous accepted service level.

If you install service and then find that, for some reason or another, you need to revert to a previous release, you typically need only to use SMP/E to restore to the previous accepted service level. However, some service releases contain defects that are incompatible with previous releases of WebSphere Application Server for z/OS.

Note: You will know if you are applying a service level that contains backward-incompatible defects if you receive message BBOO0286A (WTOR message 286) when the post installer applies the service. The console will prompt you to accept the backward-incompatible change. If you decide not to accept the change, you must either use SMP/E to roll back to the previous level or follow the post-install backout plans described in “Post-install tasks to complete after using SMP/E to apply a new service level” on page 245.

When you restore to the previous accepted service release, you need to first ″back off″ any post-install actions containing backwards-incompatible defects that were applied during that service before using SMP/E to restore to the previous accepted service level. This article describes the steps necessary to run the backoutPTF.sh shell script, which is located in your bin directory (WAS_HOME/bin, where WAS_HOME is the absolute path of the WebSphere runtime home directory) and which handles the backing out of applied post-install service for you.

1. Bring down those servers in the node for which you are doing back-off processing.
2. Launch the backoutPTF.sh script. You will need to specify the product and the committed service level to which you intend SMP/E to revert.

   Example: This is an example of the command used to run the backoutPTF.sh shell script, run from the shell by a user with the proper authority. It prepares the WebSphere Application Server for z/OS runtime home directory to run at service level ″o0511.05.″

   ```
   backoutPTF.sh WebSphere o0511.05
   ```

   Note: The shell script is case-sensitive.
3. After using the backoutPTF.sh shell script, follow the normal SMP/E procedures for restoring to the previous accepted service level.

You know you are done when you are successfully back on the service level that you want.

The following is a list of common errors that might occur when you use backoutPTF.sh to back out of service. The output goes right to stdout (on the shell from where the applyPTF.sh script was run).

A valid, currently installed PTF or APAR name and the target service level must be specified (case-sensitive).
If a PTF or APAR is not specified, then the ″usage″ of the command is printed.

Troubleshooting installation and configuration

This article describes troubleshooting the installation of the WebSphere Application Server for z/OS product.

If an installation is not successful, use this troubleshooting information to correct the problems.

For current information available from IBM Support on known problems and their resolution, see the IBM Support page.

In order to avoid problems when you install, review the steps listed on the “Ensuring problem avoidance” on page 249 page.

If your application fails repeatedly during operations causing the application servants to terminate, workload management (WLM) might terminate the application environment for the application. Refer to “Handling workload management and server failures” on page 253 for more information.

For a convenient way to find an error message, refer to “Installation problems” on page 253 to search the database.

The “Installation problems” on page 253 contains more detailed debugging and reporting instructions.

IBM Support has documents that can save you time gathering the information that you need to resolve a problem. Before opening a PMR, see the IBM Support page.

Ensuring problem avoidance

To implement WebSphere Application Server for z/OS, you must implement the necessary features, subsystems, and resources required for the runtime environment. This section provides checklists for tasks you should verify before running your WebSphere Application Server for z/OS system in order to prevent the most common errors encountered during the installation.

Before you begin: Perform the following steps to ensure problem avoidance, checking off each item as you complete it:

1. Prepare your z/OS environment:

Table 7.

Check off	Item
	Check that all the maintenance suggested in the PSP bucket WASAS610 subset H28W610 has been applied.
	Make certain your address space is large enough. Some WebSphere Application Server for z/OS servers must be able to get a 1GB virtual region to run any workload. Make sure that your installation exits (IEFUSI) do not limit the virtual region size. We recommend that you specify REGION=0M so as not to limit their size.
	Add another local page data set, two if your system does any paging of the WebSphere Application Server for z/OS server address spaces.

2. Prepare your DB2 subsystem (if you will use DB2):

Table 8.

Check off	Item
	Increase the MAX USERS (CTHREAD) and MAX BATCH CONNECT (IDBACK) in your DB2 environment settings. Use the sample job in DSN710.SDSNSAMP(DSNTEJ6Z) to display the ″ZPARMS″ settings of the running system. (An alternative is to use the DB2 Control Center to display these parameters.)
	Define at least 200 buffers to the DB2 BP32K buffer pool. Use this command to display the current bufferpool allocations: `-dis bpool(active)detail`. Verify JDBC 2.0 functionality. The JDBC IVT sample01 JAVA application does not exercise JDBC 2.0 drivers nor the RRS attach facility. A modified version that tests these functions can be found in the DB2 Conundrum white paper at http://www.ibm.com/support/techdocs/atsmastr.nsf/PubAllNum/WP100217. (This will also verify that the DSNJDBC plan is bound correctly and that it matches the .ser file.)
	Verify the level of DB2 code running on your system with the DSNTEJ6U sample job or run the DSNUTILB utility with the `DIAGNOSE DISPLAY MEPL` command. The module names, dates, and PTF number on the right of the report are in EBCDIC.
	Make sure that any updates to the DB2 ERLY code are installed, and that you have IPLed your system to activate them.

	Check the JDBC service installed on your system. Use the following Java program to display the service level:
	`export LIBPATH=/usr/lpp/db2/db2710/lib:$LIBPATH>`
	`java -cp /usr/lpp/db2/db2710/classes/db2j2classes.zip`
	`COM.ibm.db2os390.sqlj.util.DB2DriverInfo`
	The typical output message looks like this:
	`DB2 for OS/390 SQLJ/JDBC Driver build version is:DB2 7.1 PQ54756`

3. verify your UNIX System Service configuration:

Table 9.

Check off	Item
	Specify enough threads, files, and processes in your BPXPRMxx member of parmlib. Here is a starting list if you do not have it set up yet: • MAXTHREADS:10000 • MAXTHREADTASKS:5000 • MAXFILEPROC:10000 • MAXSOCKETS in the AF_INET domain:12000
	If you have an exit that checks for valid accounting codes, you might need to specify an accounting value for spawned address spaces. Use the _BPX_ACCT_DATA= variable in the was.env file.
	Ensure that the user ID associated with running the installation jobs that run the BPXBATCH shell scripts has an OMVS segment that directs PROGRAM('/bin/sh') to use the z/OS shell rather than the tcsh (C) shell (at '/bin/tcsh'). These particular shell scripts will not run from the tcsh shell.

4. Plan your SMP/E tasks:

Table 10.

Check off	Item
	You can install WebSphere Application Server for z/OS into an SMP/E environment (SMP/E 3.1 or later) separate from the one you use for z/OS. This includes target and distribution zones, as well as HFS data sets. We recommend that you use a separate environment, but you should enable the cross-zone checking so that any prerequisite service requirement can be checked between the WebSphere Application Server for z/OS and z/OS SMP/E zones.
	Verify that the DDDEF for the LTS data set describes a PDSE format data set. This will avoid LINK-EDIT errors during the SMP/E processing.
	You should carefully read the *WebSphere Application Server for z/OS: Program Directory*. This is a very large product and you have to make sure that there is sufficient space in all target and temporary data sets for receive and apply processing.

5. Plan for the ISPF Customization Dialog or the Profile management tool.
   - Plan for the ISPF Dialog considerations:

Table 11.

Check off	Item
	Do not use ISPF Dialog in Split Screen mode. You might not see everything on one screen. Keep in mind that there might be parameter values outside of the visible portion of your screen and that this might cause some of the WebSphere Application Server for z/OS installation options to be set to default values, which might not be what you want.
	Turn off PFSHOW on the ISPF Dialog screen. As with the split screen problem, there might be parameter values outside the visible portion of your screen and this might cause some of the WebSphere Application Server for z/OS installation options to be set to default values, which might not be what you want.
	You need to use a screen size of at least 32 lines to be able to invoke ISPF Dialog.

Table 11. (continued)

Check off	Item
	You might want to consider starting with new CNTL and DATA data sets as output of the ″Generate the Jobs Stream″ function. During this task, you get a large number of confirmation messages for every successful generated job. It is very easy to overlook an error message during this process. Unfortunately this sometimes causes the job generation process to stop without recreating all JCL. When you run the jobs you might run ″old″ JCL and this will result in unpredictable problems.

- Plan for the Profile Management tool

Table 12.

Check off	Item
	Install the Application Server Toolkit.
	Make sure you have an FTP server running on your target z/OS system.
	Complete the worksheets for each area before beginning.

6. Check your TCP/IP configuration:

Table 13.

Check off	Item

	Telnet into UNIX Systems Services and issue these commands to verify that you can find your host name by IP address or IP host-name:
	• Get the local host name: `hostname`
	– You will get a response such as: sc49.itso.ibm.com
	– Use the output from the hostname command for the following `nslookup` command.
	• Get host address by name: `nslookup sc49.itso.ibm.com`
	– You will get a response such as this:
	`Server:sc49.itso.ibm.com`
	`Address:9.12.6.15`
	`Name:sc49.itso.ibm.com`
	`Addresses:9.12.6.15`
	– Use the dotted IP address from this display for the following command.
	• Get host name by address: `nslookup 9.12.6.15`
	– You will get a response such as in the previous nslookup display.
	There is also a small Java program, InetInfo.java, that you can run to verify the same TCP/IP configuration. See techdocs for the program at http://www.ibm.com/support/techdocs/atsmastr.nsf/PubAllNum/TD100609.
	Example: This example shows you how to run the InetInfo Java code.
	`JAVA4 @SC42:/u/java4>export PATH=/usr/lpp/java/IBM/J1.3/bin`
	`JAVA4 @SC42:/u/java4>java InetInfo`
	`get Local Host`
	`IP Address:9.12.6.27`
	`get Host Name By Address using 9.12.6.27`
	`Host Name:wtsc42oe.itso.ibm.com`
	`get Host Address By Name using wtsc42oe.itso.ibm.com`
	`Host Address:9.12.6.27`

	Issue the `hometest` command from TSO. It should show the correct TCP Host name, corresponding IP address(es), and HOME IP addresses. If it does not produce the correct results, then TCP/IP is not configured correctly.
	If the fully qualified TCP/IP HostName is greater than 24 characters, then a DNS will be required. Otherwise, the /etc/hosts file can provide the naming lookup.

Table 13. (continued)

Check off	Item
	Verify that the DNS name you are using is definitive (authoritative) in your installation.

7. Verify that security is in place:

Table 14.

Check off	Item
	Check that the location service daemon has access to parmlib concatenation to retrieve CTRACE settings in the CTIBBOxx member.
	Verify that all WebSphere Application Server for z/OS servers must have READ access to any data sets or files in their JCL procedures.
	Verify that your installation has the RACF list-of-groups turned on. (SETROPTS LIST will show you if turned on or off.) Without this list of groups turned on, an ID cannot belong to more than one group and ASSR1 associates with only WSSR1 instead of both WSSR1 and WSCFG1. Use the command `SETROPTS GRPLIST` to turn on the list of groups.
	Define the profile BPX.SAFFASTPATH in the FACILITY class to enable SAF fastpath support.
	If you do not load SBBOLOAD into LPA, you must add it to the program control list in the RACF PROGRAM class. (With z/OS Version 1.2, you can use the FACILITY class profile BPX.DAEMON.HFSCTL class. This will cause only HFS files to be checked for program control.)
	Verify that the authorization bits for the WebSphere Application Server for z/OS HFS (default name is /usr/lpp/zWebSphere/V6R0) file are correctly set up for the WebSphere Application Server for z/OS configuration group.

8. Verify that product code is consistent:

Table 15.

Check off	Item
	After any maintenance has been applied, verify that the code loaded in LPALIB or LNKLST is synchronized with the code in the product file system. Check the location service daemon job log to verify that the correct maintenance level is in use.

You are done when you have checked all the applicable items.

Error when installing WebSphere Application Server for z/OS

This article provides troubleshooting tips for installation errors with WebSphere Application Server for z/OS.

CEE3250C The system or user abend SCC3 R=00020001 was issued. From compile unit BBODBDLD at entry point BBODBDLD at compile unit offset +xxxxxxxx at entryoffset +xxxxxxxx at address xxxxxxxx

This error occurs when SBBOLPA cannot be found in either LPA or STEPLIB. For example, you would see this error if SBBOLPA resides in the link list.

To resolve this problem, place SBBOLPA in LPA or STEPLIB when installing WebSphere Application Server for z/OS and OS/390.

For current information available from IBM Support on known problems and their resolution, see the IBM Support page.

IBM Support has documents that can save you time gathering information needed to resolve this problem. Before opening a PMR, see the IBM Support page.

Handling workload management and server failures

If your application fails repeatedly during operations causing the application servants to terminate, workload management (WLM) might terminate the application environment for the application. WebSphere Application Server for z/OS issues the following message if it tries to use a failed application environment:

```
BBOO0075E Unable to schedule work. WLM application environment applenv has stopped.
```

You need to fix the problem with your application, then restart the application environment with the RESUME option on the VARY WLM command.

Perform these steps to check and start the WLM application environment.

1. Display the application environment.
   - For static application environments, issue the following command:
     ```
     d wlm,applenv=*
     ```
   - For dynamic application environments, issue the following command:
     ```
     d wlm,dynappl=*
     ```
2. Start the application environment.
   - For static application environments, issue the following command:
     ```
     v wlm,applenv=environment_name,resume
     ```

     where *environment_name* is the application environment name.
   - For dynamic application environments, issue the following command:
     ```
     v wlm,dynappl=applenv,resume
     ```

You know that you have finished when a re-display of the application environment shows that it is available.

Installation problems

Select the problem that you are having with the WebSphere Application Server installation:

- I get an "Error when installing WebSphere Application Server for z/OS" on page 252.
- The installation completes but the administrative console does not start.

IBM Support has documents and tools that can save you time gathering information needed to resolve problems. Before opening a problem report, see the Support page:

- http://www.ibm.com/software/webservers/appserv/zos_os390/support/

Post-installation notes on the error log

After installation is complete, use the administrative console to change the log stream name or create new log stream names for servers or servants.

Note:

- A server’s error log stream setting overrides the general WebSphere Application Server for z/OS setting, and a servant setting overrides a server setting. Thus, you can set up general error logging, but direct error logging for servers or servants to specific log streams.
- If you create a new log stream name through the administrative console, you must configure a new log stream on z/OS and, if using the coupling facility, define a corresponding new coupling facility log stream.
- If you changed an existing log stream, or created a new one, you probably need to restart WebSphere Application Server for z/OS. When the name of a log stream is changed through the administrative console, in most cases a restart of WebSphere Application Server for z/OS is required before the change becomes effective. The only case when the change takes effect automatically is when the log stream name is changed for a server along with other changes that cause the server to be restarted.

If you want WebSphere Application Server for z/OS messages that occur during execution of a z/OS client to be recorded in an error log stream, code the `client_ras_logstreamname` WebSphere variable in its environment file then initialize the client. For more information about `client_ras_logstreamname` and the related variable `ras_log_logstreamName`, see the WebSphere variables in the administrative console or the information center.

Our RACF samples BBOWBRAC and BBODBRAC give UPDATE authority to the runtime control and servant user IDs for the log stream you created (they require that you supply a log stream name). After installation and customization, if you want to grant access to the log stream:

- For each server identity that writes to the log stream (or client identity, if you allow clients to write to the error log stream), assign UPDATE access to the log stream.
- For each user who browses the error log stream, assign READ access.

Follow the sample RACF commands in BBOWBRAC or BBODBRAC.

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Intellectual Property & Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300
2455 South Road
Poughkeepsie, NY 12601-5400
USA
Attention: Information Requests

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks and service marks

For trademark attribution, visit the IBM Terms of Use Web site (http://www.ibm.com/legal/us/).