WebSphere® Process Server for z/OS

Version 6.0.2

Installing and Configuring WebSphere Process Server

Note

Before using this information, be sure to read the general information in "Notices" on page 223.

30 March 2007

This edition applies to version 6, release 0, modification 2 of WebSphere Process Server for z/OS (product number 5655-N53) and to all subsequent releases and modifications until otherwise indicated in new editions.

To send us your comments about this document, email doc-comments@us.ibm.com. We look forward to hearing from you.

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.

Contents

Installing and configuring WebSphere Process Server

This section contains information you will need to prepare for, install and configure WebSphere® Process Server for z/OS®.

WebSphere Process Server documentation (in PDF format)

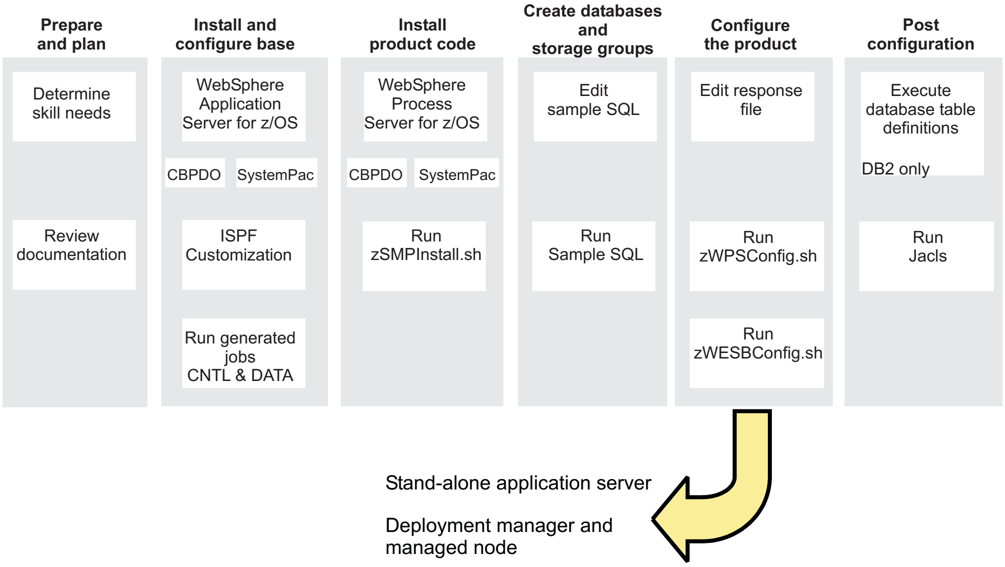
Task overview: installation and configuration

This article introduces the WebSphere Process Server for z/OS installation and configuration task flow for the supported configurations.

Before you begin

This article introduces the context of installing and customizing IBM® WebSphere Process Server, including the tasks you need to perform before and after installing.

The following diagram illustrates a high-level task flow for installing and configuring the product.



To create a complete, customized WebSphere Process Server application serving environment, you need to install the product binaries, create WebSphere Process Server definitions, augment your user profile as needed and bring up your server.

Overview of installation and configuration

This section contains information to help you understand the WebSphere Process Server for z/OS installation and configuration process. It includes information on WebSphere Process Server for z/OS packaging and terminology and the task flow for installing and configuring the software.

Before you install and configure WebSphere Process Server for z/OS, you should read all of the information contained in the **Overview of installation and configuration** section.

Relationship to the WebSphere Application Server for z/OS installation and configuration

WebSphere Process Server for z/OS installation and configuration is tightly integrated with and dependent on the installation and configuration of WebSphere Application Server for z/OS.

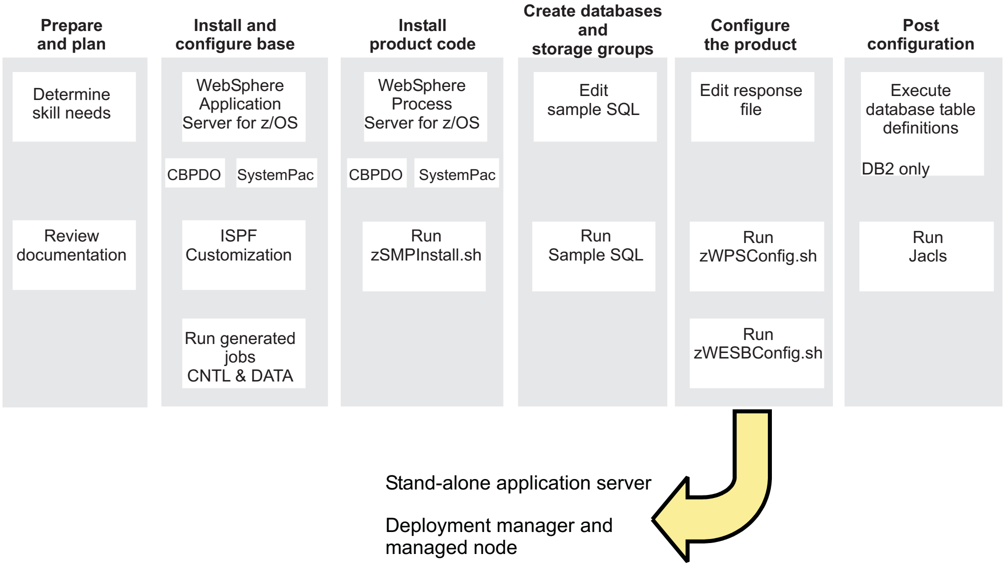
The relationship between WebSphere Application Server for z/OS and WebSphere Process Server for z/OS is highlighted during the process of creating a stand-alone WebSphere Process Server for z/OS server. Before you can create this server you need to have customized the WebSphere Application Server for z/OS, this results in the default profile for that stand-alone application server being ″augmented″ with WebSphere Process Server for z/OS stand-alone server configuration data, configuring that stand-alone application server into a stand-alone server.

WebSphere Application Server for z/OS is also used to create a deployment manager within WebSphere Process Server for z/OS. A deployment manager manages the configuration for all the managed nodes in its cell and deploys applications to any managed node in the cell.

Attention: On other platforms, the configuration files associated with a WebSphere Application Server runtime environment are called profiles; profiles can be copied and manipulated with the manageprofiles command. In WebSphere Application Server for z/OS, all runtime environments are created with the ISPF Customization Dialog using a profile name of **default.** The WebSphere Process Server for z/OS configuration process adds applications and resources to this **default** profile thus creating a WebSphere Process Server for z/OS runtime environment.

Task flow of installation and configuration

The following diagram illustrates a high-level task flow for installing and configuring WebSphere Process Server for z/OS.



To create a complete, customized WebSphere Process Server application serving environment, you need to install the product binaries, create WebSphere Process Server definitions, augment your user profile as needed and bring up your server.

Before you begin

To successfully install and configure WebSphere Process Server for z/OS you need to ensure that you are adequately prepared. Before you begin, evaluate your skills and educations needs, install the documentation, determine hardware and software requirements, plan your installation and configuration, and prepare the base operating system.

Installing the documentation

You can install and run a server copy of the WebSphere Process Server for z/OS information center on z/OS.

Requirements and considerations for installing and running an information center on z/OS

The documentation is packaged as Eclipse document plug-ins and must be viewed using the IBM WebSphere help system. The IBM WebSphere help system (or viewer) and document plug-in format are based on an open source approach developed by the Eclipse Project. IBM product document plug-ins are contained in folders that follow a consistent naming convention (com.ibm.*xxx*.doc).

The IBM WebSphere help system is packaged on the WebSphere Process Server for z/OS product tape. The information in this topic assumes that the IBM WebSphere help system has been loaded on to the z/OS server as part of the WebSphere Process Server for z/OS product install.

You should be aware of the following requirements and considerations if you want to install and run a server copy of the WebSphere Process Server for z/OS information center:

- Support for the information center on z/OS is only offered in server mode for remote access using a browser.

- The information center uses a JRE. If you want to run an information center on z/OS, you need to use the JRE provided with the operating system.
- The information center runs in UNIX® ® System Services (USS) on z/OS 1.4 or later.
- To get the best results when viewing the information center, it is recommended that you use one of the following browsers:
  - Microsoft® Internet Explorer 6.0
  - Mozilla 1.7
- To view PDF documents within the information center, you must have Adobe Acrobat Reader 4.05 or higher installed, and the Acrobat Web plug-in installed in your browser.
- Make sure that you have proper authority / permissions set by the systems administrator to work with files on the z/OS system.
- The first time you perform a search in the information center, there will be a delay where the pregenerated search indexes are unzipped and any additional files are indexed. In a server environment, the indexing takes place once for every browser locale that performs a search in the information center. For example, if a user has a browser locale of **en_gb** and another user has a locale of **en_us**, both users will create a search index.
- You can choose to download PDF versions of the documentation from the WebSphere Process Server for z/OS library page as another option for working with the information offline.

Note: If you already have an Eclipse help system, or an Eclipse-based IDE such as WebSphere Integration Developer, you can opt to install only the WebSphere Process Server for z/OS document plug-in and view the documentation within that tool.

Installing the document plug-ins

To view IBM product documentation in the viewer, you must install new or updated document plug-ins into the `eclipse/plugins` folder of the IBM WebSphere Help System. The help system works with any information that has been packaged as an Eclipse document plug-in, including IBM product document plug-ins. IBM product plug-in folders are easily identified because they use a common naming convention (com.ibm.*xxx*.doc).

To install document plug-ins, complete the following steps:

1. Install the Eclipse help system by performing the following steps:
   a. Change to the following directory: `cd` *`install_root`*`/zos.iehs/lib`.
   b. Locate the file **IBM-Help-zOS.tar**
   c. Create the directory into which you will untar **IBM-Help-zOS.tar**. For example:

      `mkdir<`*`dirname`*`>`

      Where <*dirname*> is a user-defined location.
   d. Untar the file into a directory of your choosing by using the command:

      ```
      tar -xvf IBM-Help-zOS.tar -C <directory name>
      ```

      This extracts the help system to the named directory.
2. Obtain the document plug-ins from the IBM product Web page http://www.ibm.com/software/integration/wps/library/infocenter/.
3. Download the document plug-ins that you require.

You can download zip file to a local directory on your machine and then use FTP to transfer the file to the following directory on the z/OS system.`<directory name>/eclipse/plugins`

This is the directory that was created when you un-tarred the help system.

- If you tarred the plug-ins before using FTP, you can use the tar command above to extract them.
- If you zipped the plug-ins before using FTP, you can use the jar command to unzip them to the right directory. For example, you could enter jar -xvf <*plugins.zip*>, where plugins.zip is a user-defined name.

These commands maintain the directory structure. Ensure that the plug-ins, when extracted, are placed in the **eclipse/plugins** directory.

Note: Keep in mind that this must be the `eclipse/plugins` folder of the IBM WebSphere Help system; you might have additional `eclipse/plugins` folders on your system.

4. Change the permissions on the **IC_start.sh** and **IC_end.sh** scripts so that you have permission to edit and execute the files that start and stop the information center.
5. Edit the start up file **IC_start.sh**, that is provided in the `ibm_help` directory, to point to the location of Java™.

   For example, you could add the following to the file: export PATH=/usr/lpp/java142/J1.4/bin:$PATH, where `/java142/J1.4/bin:$PATH` is the location of Java.

   Note: Eclipse requires Java™ 1.4.2 to run. This is provided with the information center on all platforms except z/OS.
6. The port number is also specified in the file IC_start.sh - the default is 8888. You can change this by editing the -port parameter.

Starting the viewer

Batch files are provided to run the information center, either locally on a single workstation or on a server for multiple users.

You can start the information center from ISPF or using telnet.

To start the information center on a server, perform the following steps:

1. Open the directory where you have installed the help system.
2. Run the file `IC_start.sh`.

   You can run the information center as a background task. This means that even if you log off the workstation or server, the information center will continue to run. Use the following command: ./IC_start.sh &.
3. To verify that the information center has started, open a browser and enter a URL using the name of the server and the port number that is specified in the file `IC_start`.

   For example, you could specify: `http://winmvs26.site.company.com:8888`. The default port is 8888, but you can change this value by editing the startup file `IC_start` and setting the -port parameter to a suitable value.

   Note: The first time you perform a search in the information center, there will be a delay where the pregenerated search indexes are unzipped and any additional files are indexed. This action is only performed once in the local information center. In a server environment, the indexing takes place once for every browser locale that performs a search in the information center. For example, if a user has a browser locale of **en_gb** and another user has a locale of **en_us**, both users will create a search index.

Shutting down the viewer

You can start the information center from ISPF or using telnet.

To shutdown the information center on a server, perform the following steps:

1. Open the directory where you have installed the help system.
2. Run the file `IC_end.sh`.

   You can run the information center as a background task. This means that even if you log off the workstation or server, the information center will continue to run. Use the following command: ./IC_end.sh &.

Installing the documentation using the help system packaged on the supplemental CD

In addition to packaging a server version of the IBM WebSphere help system on the WebSphere Process Server for z/OS product tape, there is a version of the help system included on the supplemental Message Service Client CD (LCD8-0507 for Linux® operating system or LCD8-0506 for Windows® operating system), which is included with the WebSphere Process Server for z/OS product.

To install the documentation plug-ins perform the following steps:

1. Install the IBM WebSphere Help System, which can be found on the Messaging Clients supplemental CD.

   You can find the help system component on the \IEHS directory.
2. Obtain the document plug-ins from the IBM product Web page at http://www.ibm.com/software/integration/wps/infocenter/
3. Copy the document plug-in folders to the eclipse\plugins folder of the help system.

   For example, if you installed the help system to C:\WebSphere Help System, you copy the document plug-in folders to: C:\WebSphere Help System\eclipse\plugins

   Keep in mind that this must be the eclipse\plugins folder of the IBM WebSphere Help system; you might have additional eclipse\plugins folders on your system.
4. To see the newly added document plug-in, start the viewer by following the instructions in the Starting the viewer section below (or shut it down by following the instructions in the Shutting down the viewer section below, and then restart it if it was running).

Starting the viewer for help system installed from CD

To start the viewer:

1. Open the WebSphere Help System folder.
2. Double-click the help_start.bat file.

Note: It might take a few minutes for the system to start and the document plug-ins to be displayed the first time you start it.

Shutting down the viewer from help system installed from CD

When you close the viewer by simply closing its window, its processes are still running in the background. This enables a much faster launch during subsequent sessions. However, you must shut down these processes when you install document plug-ins or update the help system with new plug-ins. Shutting down the viewer when not in use also frees up system memory. When you shut down your machine, all help system processes shut down.

To shut down the viewer:

1. Open the WebSphere Help System folder.
2. Double-click the help_end.bat file.

Viewing information in different languages

To view information in a different language, change the regional settings of your operating system to the locale and language of your choice.

Important: The system will display the translated documentation that has been provided by the product owner. If the product owner has not provided the documentation in your chosen language but has provided the English version, the system will display the English version by default, even if the browser language option is set to another language.

z/OS installation terminology

z/OS installation terminology refers to the terms and phrases used throughout the installation and configuration documentation for WebSphere Process Server for z/OS. Becoming familiar with these terms and phrases will facilitate an understanding of the z/OS installation and configuration concepts and tasks.

Key Terms

Because the product installs on top of WebSphere Application Server for z/OS, you should also become familiar with WebSphere Application Server for z/OS terminology as well. See WebSphere Application Server for z/OS terminology for descriptions of WebSphere Application Server for z/OS components and configuration types.

Although you may be familiar with these terms in general, the definitions below define the terms *within the context of the product installation.*

Custom-built Product Delivery Option (CBPDO)

A software delivery package consisting of uninstalled products and un-integrated service. Installing what you have ordered in the CBPDO (stand-alone products and/or z/OS elements and features) requires the use of SMP/E. CBPDO is one of the two entitled methods for installing z/OS, ServerPac being the other.

Ant script

An Ant script is an XML file that is used to define targets which run various *Ant tasks* as part of your installation procedure. The Ant scripts are associated with a configuration type. The actions registry sets the priority and sequence in which the Ant scripts run.

command line

The blank line on a display where commands, option numbers, or selections can be entered.

After you unload the contents of the installation media using SMP/E, you run the product installation and configuration scripts from a command line. The ant configuration scripts send messages to log files. Messages related to the installation and augmentation process do display to the user running the installation and configuration scripts.

The command line installation for this product does not prompt you for input values.

configuration type

The server types associated with the WebSphere Application Server for z/OS configuration onto which you will install and configure WebSphere Process Server for z/OS. WebSphere Application Server for z/OS configuration types are as follows:

- Stand-alone application server cell
- Network Deployment cell
- Managed node in a Network Deployment cell

The WebSphere Process Server for z/OS configuration script points to a response file that contains property values. These property values are used to augment the WebSphere Application Server for z/OS default profile with WebSphere Process Server for z/OS configuration data.

For detailed information on the server types, including a description of each type and information on how to install and configure WebSphere Application Server for z/OS for each of these server types, see the WebSphere Application Server for z/OS information center.

mount point

In Linux and UNIX operating systems, the directory at which a file system is mounted and under which other file systems may be mounted.

Installation HFS

When you unload the contents of the installation media using SMP/E, the result is an HFS commonly referred to as the *installation HFS* or *smpe_root*. The installation HFS is read-only.

Once UNIX System Services recognizes an installation HFS, other HFSs can be mounted at paths off this root. The installation HFS that is defined may contain directories such as /tmp, /bin, /lib, /etc, /usr/lpp and so forth. For any application that resides in UNIX Systems Services, an HFS exists to hold the product executable code and the parameter files.

WebSphere Process Server for z/OS and WebSphere Application Server for z/OS have a significant portion of their code delivered in the form of an HFS file system. The WebSphere Application Server for z/OS file system is packaged in an MVS™ data set called SBBOHFS. That data set is mounted at a mount point to form the *SMP/E Home Directory* for the WebSphere Application Server for z/OS installation. The WebSphere Process Server for z/OS file system is packaged in an MVS data set called SBPZHFS. That data set is mounted at a mount point to form the *SMP/E Home Directory* for the WebSphere Process Server for z/OS installation.

For information on naming conventions for the installation HFS and other directories, see Directory conventions.

Configuration HFS

Refers to the mount point for the configuration file system (sometimes referred to as the configuration HFS) in WebSphere Application Server for z/OS. 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, known as the configuration HFS (sometimes referred to as the *configuration_root*). These configuration directories are user-defined and are created through the WebSphere Application Server for z/OS configuration process and contain symbolic links to files in the installation HFS (smpe_root) directory.

For information on naming conventions for the configuration HFS and other directories, see Directory conventions.

ISPF Interactive System Productivity Facility (ISPF). An IBM licensed program that serves as a full-screen editor and dialog manager. Used for writing application programs, it provides a means of generating standard screen panels and interactive dialogs between the application program and terminal user. In WebSphere Application Server for z/OS, the WebSphere administrator uses ISPF customization dialogs to configure the WebSphere environment onto which you will install and configure WebSphere Process Server for z/OS.

Although WebSphere Process Server for z/OS itself does not utilize ISPF customization panels, you should understand the role of ISPF customization for the base product and you should be familiar with how base product configuration in general relates to the product installation and configuration.

Jacl Java command language. A scripting language for the Java 2 environment that is used to create Web content and to control Java applications.

A product administrator can run post installation Jacl scripts from a command line to further configure the environment. These scripts must be run using the WebSphere wsadmin tool (located in the install_root/bin directory).

The types of Jacl scripts and the number of Jacl scripts that a product administrator runs after installing or after configuring the product will depend on environment variables, such as which database will be used (and) how the site intends to use WebSphere Process Server for z/OS.

Installing the software

Installing the software refers to the multiphase process of loading the contents of the product installation media onto the z/OS system and then running the installation script to install the WebSphere product definitions that enable the product for use.

At sites that install and run products on z/OS, the responsibility for loading the contents of installation media is usually assigned to a system programmer. In most cases the system programmer uses SMP/E to load the contents of the installation media onto the system.

After successfully loading the contents of the installation media onto the system, the system programmer can continue the installation process by running the product installation script, or the system programmer can inform a product administrator that the product code has been loaded successfully and the product administrator can run the installation script named **zSMPInstall.sh**.

Running the installation script enables the product for use creating *symlinks* to the read-only installation HFS. These symlinks point to the read-only files in the installation root, such as JAR files, shell scripts, and so on.

Configuring the software

On z/OS, *configuring the software* refers to those activities that are performed by a product administrator *after the product has been loaded onto the system and after the installation script has been run*. Configuring the product involves the following tasks:

- Augmenting the default profile.

  To augment the default profile, the product administrator can run one of two configuration scripts from the command line as follows:

  – Run **zWPSConfig.sh** to augment the default profile with configuration data consisting of all the features and functions of WebSphere Process Server for z/OS.

  – Run **zWESBConfig.sh** to augment the default profile with configuration data consisting of the enterprise service bus subset functionality of WebSphere Process Server for z/OS.

  Note: When you license WebSphere Process Server for z/OS, and configure it as an ESB-only server, you always have the option of re-configuring it for full WebSphere Process Server functionality at some time in the future.

- Updating the administrative console using command line scripting implemented as jacl scripts.

Profile

Data that describes the characteristics of a user, group, program, device, or remote location.

In WebSphere Application Server for z/OS, a profile represents the collection of user data, along with the shared product binaries that define the WebSphere runtime environment. On z/OS, there is always one and only one profile and that profile is named **default** in each of the configurations.

When you run the WebSphere Process Server for z/OS configuration script you augment the WebSphere Application Server for z/OS default profile with WebSphere Process Server for z/OS configuration data. The action of augmenting the default profile is known as *profile augmentation*. Profile augmentation actions are Ant scripts that get called with the properties in the response file (the path to the response file is part of the product configuration command). These actions apply a changed template to the existing WebSphere Application Server for z/OS profile.

response file

An EBCDIC file that can be customized with setup and configuration data.

When you run the product configuration script from the command line, the command syntax includes a directory path reference to the response file, which automates the profile augmentation process.

A response file cannot be used interactively. The response files are associated with specific configurations supported by the product. When you run the configuration script, you specify a response file that is associated with the WebSphere Application Server for z/OS configuration into which you are installing the product.

shell script

A file of shell commands. If the file is executable, users can run it by specifying the file's name as a shell command or as an operand on an **sh** shell command or on the TSO/E OMVS command. A shell script is similar to a TSO/E REXX exec.

The product installer includes a shell script that installs and enables the product for use, as well as shell scripts that configure the product by augmenting the default profile with product configuration data and definitions.

keyword

One of the predefined words of a programming language, artificial language, application, or command.

The shell scripts that you run to install and configure the product, contain keywords.

In the shell script that you use to install the product, the keywords are preceded by a dash (-) and include the following:

- -smproot
- -runtime
- -install
- -prereqonly

  To check for installation prerequisites.
- -trace

  Optionally, to record trace information.

In the shell script that you use to configure the product, the keywords are preceded by a dash (-) and include the following:

- -augment
- -response
- -Z

  Optionally, to override values in the response file.
- -trace

  Optionally, to record trace information.

In the shell script that you use to uninstall the product, the keywords are preceded by a dash (-) and include the following:

- -uninstall
- -response
- -Z

  Optionally, to override values in the response file.
- -trace

  Optionally, to record trace information.

Some of the keywords require qualifiers in the form of *parameters* that provide specific information with regard to enabling the product for use. For detailed descriptions of the keywords and keyword parameters supported by the installation script, see About the installation script. For detailed descriptions of the keywords and keyword parameters supported by the configuration script, see About the configuration script.

keyword parameter

A parameter that consists of a keyword followed by one or more values.

The shell scripts that you run to install and configure the product, contain keywords followed by one or more values.

symbolic link

Also referred to as *symlink*, a symbolic link is a type of file that contains the path name of and acts as a pointer to another file or directory.

After unloading the contents of the installation media by using SMP/E, you run the installation script named **zSMPInstall.sh**. As a result of running the installation script you create the symbolic links to the read-only HFS.

A majority of the WebSphere Process Server for z/OS files in the configuration root are symbolic links to files in the WebSphere Application Server for z/OS and WebSphere Process Server for z/OS smpe_root (installation HFS). This allows for many server configurations to share the same installation product code and thereby reducing the size of the configuration HFS (since it does not contain actual product files). Symbolic links are processed by the operating system and resolved to the actual file in the smpe_root (installation HFS).

z/OS UNIX System Services (USS)

An element of z/OS that creates a UNIX environment that conforms to the XPG4 UNIX 1995 specifications and provides the two open systems interfaces on the z/OS operating system:

- an application program interface (API)
- an interactive shell interface

Previously, UNIX System Services was a component of OS/390®, formerly called OpenEdition.

For the product installation process, z/OS UNIX System Services provides the shell or command interface from which you run the installation and configuration scripts.

Installation: Resources for learning

Use the following links to find relevant supplemental information about installation and customization of WebSphere Process Server. 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 Process Server product, but is useful all or in part for understanding the product. When possible, links are provided to technical papers and Redbooks(TM) that supplement the broad coverage of the release documentation with in-depth examinations of particular product areas.

View links to additional information about:
- "Planning, business scenarios, and IT architecture"
- "Programming instructions and examples" on page 13
- "Programming specifications" on page 14
- zSeries® resources
- "Administration" on page 14
- "Support" on page 14

Planning, business scenarios, and IT architecture

- WebSphere Process Server detailed system requirements

The official site for determining product prerequisites for hardware and software for WebSphere Process Server.

- 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(R) zones, such as WebSphere Business Integration (of which WebSphere Process Server is a part), 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 developerWorks: New to WebSphere Business Integration

  Resource center for those getting started with WebSphere Business Integration. It provides an overview that explains how Eclipse-based WebSphere Business Integration version 6 products work together to take you from high-level business models to deployed applications integrated within a Service-Oriented Architecture (SOA).
- IBM WebSphere Process Server home page

  The IBM WebSphere Process Server home page contains useful information, including support links and downloads for fixes, APARs, tools, and trials.
- IBM WebSphere Process Server library and information center Web site

  The IBM WebSphere Process Server Library Web site contains a link to the WebSphere Process Server information center.
- 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.
- IBM developerWorks: IBM Patterns for e-business

  The IBM developerWorks site is the source for IBM patterns for e-business, a set of tested, reusable intellectual assets that you can use to design and implement your e-business network and architecture.
- The User centered design (UCD) for different project types, part 2

  This Web page is the latest of two articles that describes design activities that IBM scientists have found most useful in various types of projects. This article defines user interface design elements, including the design prototype, use case model, and design specification document.
- The z/OS Technical Overview: WebSphere Process Server and WebSphere Enterprise Service Bus

  This IBM Redpaper describes how WebSphere Process Server V6 for z/OS (used primarily to run business processes) and WebSphere Enterprise Service Bus V6 for z/OS (used to implement an ESB) benefit from the strengths of System z™ and z/OS.

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, the product which WebSphere Process Server extends.

Programming specifications

- J2EE information

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

zSeries resources

- z/OS Internet Library

  The z/OS Internet Library contains documentation for z/OS elements, features, and software products.
- IBM Education Assistant

  This site integrates narrated presentations, Show Me Demonstrations, tutorials, and resource links to help you successfully use the IBM software products. The materials include information on WebSphere Process Server for z/OS and WebSphere Application Server for z/OS.
- SMP/E Internet Library

  This SMP/E Internet Library contains SMP/E technical documentation and white papers.
- z/OS basic skills information center

  The z/OS basic skills information center includes topics on z/OS concepts, programming and security.

Administration

- The IBM Terminology Web site

  This glossary consolidates and defines the terminology from many IBM products in one convenient location. In addition to base computer terminology, terms and definitions from many different IBM brands and product families are included. It is not a comprehensive resource of all IBM computing terms. This resource is provided for information purposes only and is updated periodically. IBM takes no responsibility for the accuracy of the information it contains.

Support

- WebSphere Process Server Support page

  Take advantage of the Web-based Support and Service resources from IBM to quickly find answers to your technical questions. You can easily access this extensive Web-based support through the IBM Software Support portal at `http://www.ibm.com/software/support/` and search by product category, or by product name. For example, if you are experiencing problems specific to WebSphere Process Server, select **WebSphere Process Server** in the product list. The WebSphere Process Server Support page is displayed.
- z/OS Support page

  This support page provides Web-based Support and Service resources for the z/OS platform.
- IBM e-server Support: Fix Central

  A Web facility for downloading fixes for the zSeries family.

Determining your skill needs

In assembling your project team, consider the skills you need to implement WebSphere Process Server for z/OS.

Documentation to support the z/OS skills can be found at this web site: z/OS Internet Library

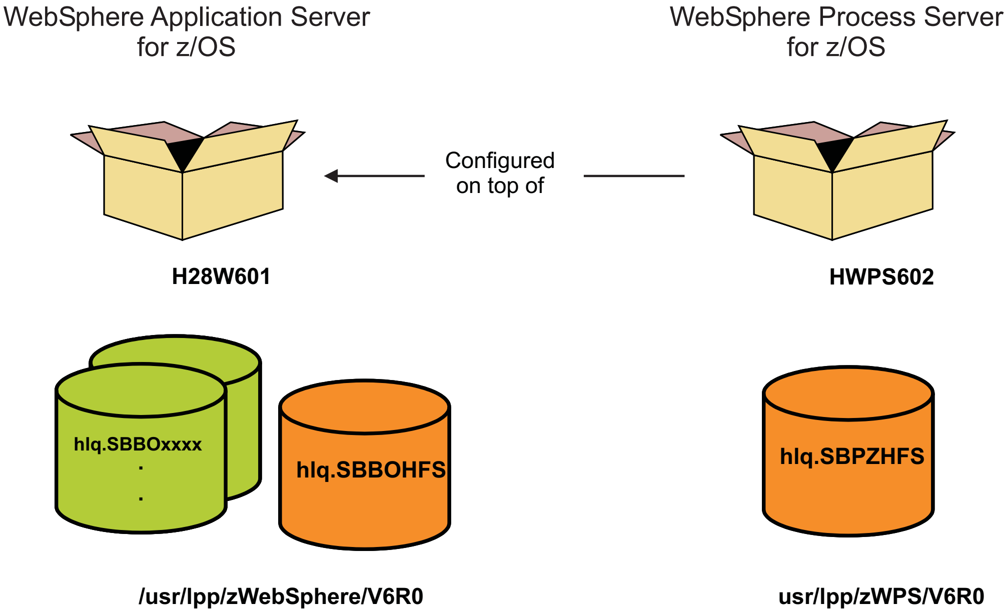
Documentation describing the skill needs required for WebSphere Application Server for z/OS can be found in the: WebSphere Application Server for z/OS information center.

Packaging

This topic lists and describes the components delivered with WebSphere Process Server for z/OS.

Program components intended to run on the z/OS platform are delivered on a single tape. WebSphere Application Server for z/OS V6.0.2 (with the latest available service level) is included on the WebSphere Process Server for z/OS product tape.

In addition to the components intended to run on the z/OS platform, there are several which are intended to run on non-z/OS systems. Some of these components are delivered on CD-ROM.



Software supplied with WebSphere Process Server for z/OS

The WebSphere Process Server for z/OS package contains software that you need to install WebSphere Process Server for z/OS, to set up your WebSphere Process Server environment, and to assemble and deploy applications. The tape media includes WebSphere Application Server for z/OS, which needs to be installed and configured prior to installing WebSphere Process Server for z/OS. CD-ROMs contain optional supplemental software programs that provide additional value and tool support for your production and development environments.

Table 1 on page 16 lists and describes the software that is provided with WebSphere Process Server for z/OS on the tape.

Table 1. Software delivered on Tape

Software	Description
WebSphere Process Server for z/OS	Based on service-oriented architecture (SOA) and as a single, simplified programming model, WebSphere Process Server is the next-generation business process server that delivers and supports all styles of integration based on open standards to automate business processes that span people, workflows, applications, systems, platforms, and architectures. New in WebSphere Process Server: • Service component architecture - One simplified integration framework that leverages existing IT • Support for all styles of integration - Including human tasks, roles based task assignments, and multilevel escalation. Visual editors for component assembly • Change business processes on the fly with relatively minimal skills • Business rules, business state machines, and selectors to dynamically choose interface based on business scenarios • Broadest reach in integration - Built on Enterprise Service Bus (ESB) technologies and support for IBM WebSphere Adapters
WebSphere Application Server for z/OS	WebSphere Application Server for z/OS is the industry's premier Java-based application platform, integrating enterprise data and transactions for the dynamic e-business world. WebSphere Application Server, upon which WebSphere Process Server is built, delivers a rich application deployment environment with application services that provide enhanced capabilities for transaction management, as well as the security, performance, availability, connectivity, and scalability expected from the WebSphere family of products. WebSphere Application Server in a network deployment configuration also enables clustering, edge-of-network services, Web services enhancements, and high availability for distributed configurations. See more details about WebSphere Application Server for z/OS packaging..
WebSphere Application Server Application Clients	Delivered as part of WebSphere Application Server for z/OS, the application client module is a Java Archive (JAR) file that contains a client for accessing a Java application. Running J2EE and Thin application clients that communicate with the underlying WebSphere Application Server product requires that elements of the Application Server are installed on the workstation on which the client runs. However, if the system does not have the Application Server installed, you can install Application Clients, which provide a stand-alone client run-time environment for your client applications.

Table 1. Software delivered on Tape (continued)

Software	Description
IBM Eclipse Help System	Delivered on tape as well as on Message Service Clients CD-ROMs, the IBM Eclipse Help System provides downloadable versions of the WebSphere Process Server documentation are packaged as Eclipse document plug-ins and must be viewed using the IBM Eclipse Help System. The help system (or viewer) and document plug-in format are based on an open source approach developed by the Eclipse Project.
DataDirect Java(TM) Database Connectivity (JDBC) drivers	Delivered as part of WebSphere Application Server for z/OS, the two JDBC drivers produced by DataDirect Technologies for enabling connectivity to Microsoft(R) SQL Server. These drivers are the SequeLink and Connect JDBC drivers.

Table 2 lists and describes the software that is provided with the WebSphere Process Server for z/OS on CD-ROM.

Table 2. Software delivered on CD-ROM

Software	Description
WebSphere Process Server Clients	WebSphere Process Server Clients provide the WebSphere Application Server for z/OS configuration with an SOA-Core and Business Process Choreographer functionality without the need of a full WebSphere Process Server installation. • WebSphere Process Server 6.0.2 Client for Windows • WebSphere Process Server 6.0.2 Client for AIX® • WebSphere Process Server 6.0.2 Client for Solaris on SPARC • WebSphere Process Server 6.0.2 Client for Solaris on x86 64-bit • WebSphere Process Server 6.0.2 Client for HP-UX • WebSphere Process Server 6.0.2 Client for HP-UX on Itanium® 64-bit • WebSphere Process Server 6.0.2 Client for Linux on x86 32-bit • WebSphere Process Server 6.0.2 Client for Linux on x86 64-bit • WebSphere Process Server 6.0.2 Client for Linux on iSeries™ and pSeries® • WebSphere Process Server 6.0.2 Client for Linux on zSeries31-bit • WebSphere Process Server 6.0.2 Client for Linux on zSeries 64-bit

Table 2. Software delivered on CD-ROM (continued)

Software	Description
IBM Message Service Clients	Software that provides messaging and Web services capabilities in non-Java environments. Extend interaction between applications and WebSphere Process Server by using the provided clients: • IBM Message Service Client for C/C++ extends the JMS model for messaging to C and C++ applications. • IBM Message Service Client for .NET enables .NET applications to participate in JMS-based information flows. • IBM Web Services Client for C/C++ is a JAX-RPC-like Web services client for C++ that enables C and C++ applications to connect to Web services hosted on WebSphere.
WebSphere Application Server Toolkit	Provides basic assembly and deployment tooling for publishing to an application server, such as WebSphere Application Server Network Deployment. You can also use the toolkit to perform basic unit testing, debugging, and profiling functions.
WebSphere Application Server Edge Components	The WebSphere Application Server Edge Components address the needs of highly available, high-volume environments with the Edge components. The Edge components include sophisticated load balancing, caching, and centralized security capabilities. See the WebSphere Application Server Network Deployment Edge Components Web page for more information.
IBM Tivoli® Access Manager Servers	The IBM Tivoli Access Manager for e-business integrates with e-business applications right out of the box, to deliver a secure, unified, and personalized e-business experience. By providing authentication and authorization APIs and integration, Tivoli Access Manager for e-business helps you secure access to business-critical applications and data that might be spread across the extended enterprise. See IBM Tivoli Access Manager for e-business for more information.

How to acquire WebSphere Process Server for z/OS

You can obtain the product code in any of the following ways:

- IBM Custom-Built Product Delivery Option (CBPDO) – the system programmer uses SMP/E to unload the product code onto the z/OS system.
- IBM SystemPac® / ServerPac – the system programmer copies SMP/E data sets that correspond to the CustomPac service level onto the z/OS system.

To buy the software, contact your IBM representative or IBM reseller, or visit the WebSphere Process Server home page at http://www.ibm.com/software/integration/wps/ and select the *How to buy* link. For additional information refer to "Installation phases" on page 21.

Directory conventions

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

The WebSphere Process Server for z/OS product follows closely the directory conventions used by WebSphere Application Server for z/OS, with WebSphere Process Server for z/OS specifics described below.

For information on the WebSphere Application Server for z/OS directory naming conventions, see Directory naming conventions in the WebSphere Application Server for z/OS information center.

smpe_root

Refers to the root directory for product code installed with SMP/E. WebSphere Application Server for z/OS and WebSphere Process Server for z/OS each has its own smpe_root.

The default smpe_root for WebSphere Application Server for z/OS is `/usr/lpp/zWebSphere/V6R0`.

The default smpe_root for WebSphere Process Server for z/OS is `/usr/lpp/zWPS/V6R0`.

The term *smpe_root* is synonymous with the term product *installation HFS*.

The WebSphere Process Server for z/OS installation script command includes the path to the WebSphere Process Server for z/OS smpe_root. In the installation script command the path to the smpe_root is preceded by the **-smproot** keyword.

configuration_root

Refers to the mount point for the configuration file system (sometimes referred to as 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 node type under the configuration_root requires its own cataloged procedures under z/OS.

The default configuration_root for WebSphere Application Server for z/OS is `/WebSphere/V6R0`.

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. Older versions of the WebSphere Application Server for z/OS product information often referred to this as the ″WAS_HOME″ directory. Corresponding product variables are was.install.root and WAS_HOME.

The default value varies based on node type.

For a stand-alone application server configuration type, the common default is:

- */configuration_root*/AppServer

For a deployment manager configuration type, the common default is:

- */configuration_root*/DeploymentManager

The WebSphere Application Server for z/OS */configuration_root/app_server_root* is referenced in the WebSphere Process Server for z/OS installation script command syntax and is preceded by the keyword **-runtime**.

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/profile_name because only the profile name "default" is used in WebSphere Application Server for z/OS.

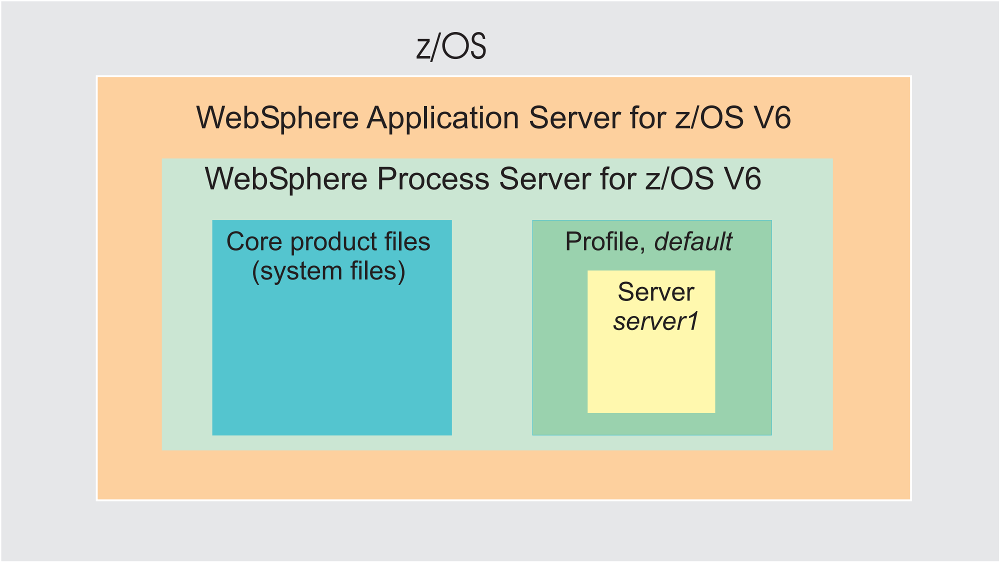
The WebSphere Process Server for z/OS configuration process "augments" the WebSphere Application Server for z/OS default profile, extending it with WebSphere Process Server for z/OS functionality.

About installing on z/OS

Installing the product on z/OS involves loading the product from the installation media onto the z/OS system and running the installation script from a command line.

Installing the product on z/OS is a multiphase process that can span multiple roles. For a description of the phases that make up an installation, see “Installation phases” on page 21.

You must install and configure WebSphere Application Server for z/OS, prior to installing WebSphere Process Server for z/OS. Installing and configuring WebSphere Application Server for z/OS creates the default profile that you will augment with a WebSphere Process Server for z/OS configuration.


z/OS
WebSphere Application Server for z/OS V6
WebSphere Process Server for z/OS V6
Core product files (system files)
Profile, *default*
Server *server1*

WebSphere Application Server for z/OS it is included as part of the WebSphere Process Server for z/OS package.

Installation and configuration scripts

Unlike many products installed on z/OS, WebSphere Process Server for z/OS does not rely on ISPF dialogs to create the installation-specific configuration data. Instead, the product comes with an installation script named **zSMPInstall.sh** and two configuration scripts named **zWPSConfig.sh** and **zWESBConfig.sh**. For detailed information on the configuration script, see About the configuration script.

Installation script results

As a result of running the installation script, all of the components and features are installed onto the system. In this sense, you cannot "customize" the product installation by picking and choosing which features to install onto your system. However, through keyword and keyword parameters in the configuration command, you can configure the installed product features. For detailed information on the configuration script, see About the configuration script.

Installation phases

Installing WebSphere Process Server for z/OS on z/OS is a multiphase process that can span multiple roles.

In general, phase 1 of the installation pertains to *loading the contents of the installation media* onto the z/OS system and is the responsibility of a system programmer. A system programmer is one who plans, maintains, and controls the use of an operating system with the aim of improving overall productivity of an installation.

Phase 2 of the installation pertains to *running the installation script* to create the required WebSphere definitions that enable the product for use. This second phase is the responsibility of a product administrator role.

The relationship between roles and installation phases is illustrated below.

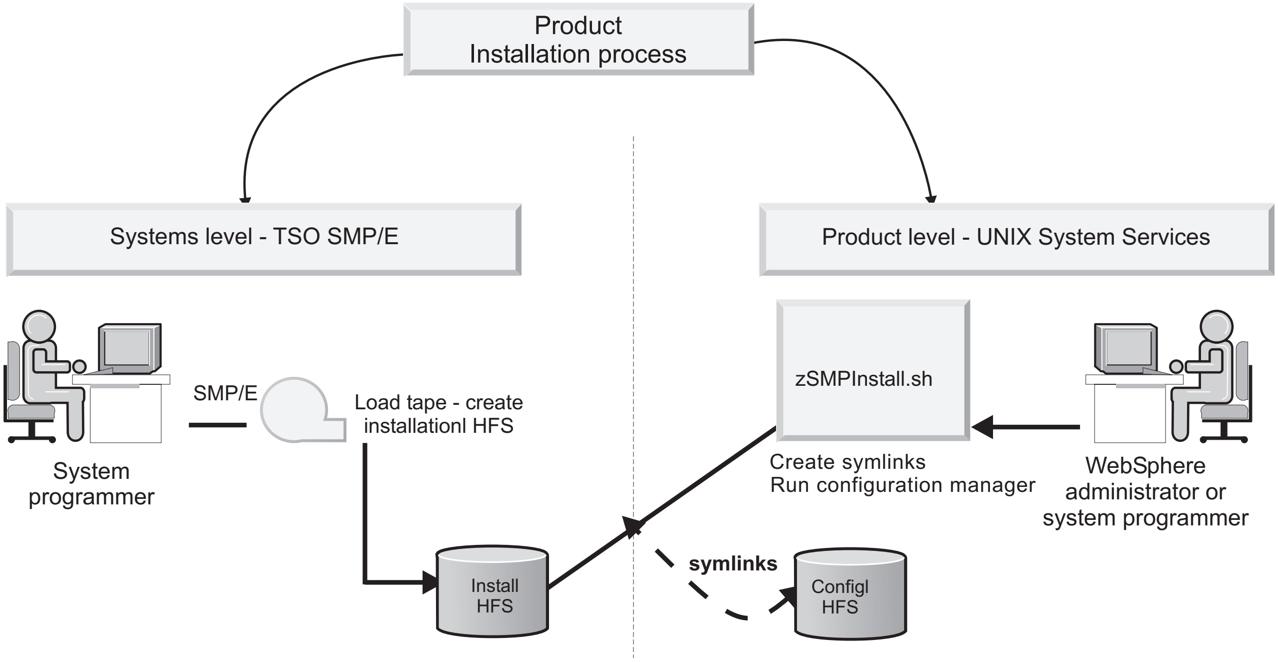


Figure 1. Installation process for WebSphere Process Server for z/OS

The product installation is not considered to be complete until both phases (loading the product code and running the installation script) have been performed successfully.

The following illustration depicts installation phases:

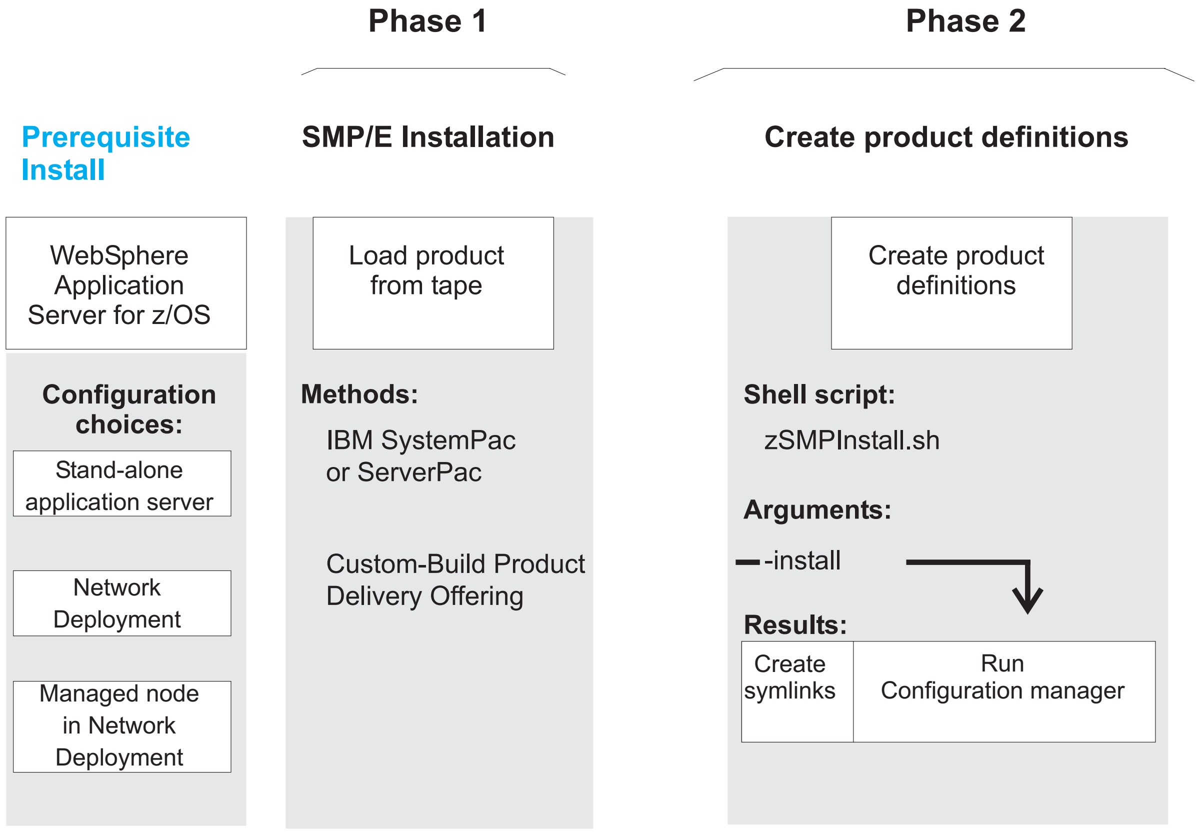

Phase 1
Phase 2
Prerequisite Install
SMP/E Installation
Create product definitions
WebSphere Application Server for z/OS
Configuration choices:
Stand-alone application server
Network Deployment
Managed node in Network Deployment
Load product from tape
Methods:
IBM SystemPac or ServerPac
Custom-Build Product Delivery Offering
Create product definitions
Shell script:
zSMPInstall.sh
Arguments:
-install
Results:
Create symlinks
Run Configuration manager

Figure 2. Installation phases

Phase 1 - Loading the product code from the installation media

If the product delivery package was an IBM Custom-Built Product Delivery Option (CBPDO), the system programmer uses System Modification Program/Extended (SMP/E) to unload the product code onto the z/OS system. The system programmer relies on the Program Directory to guide them through the process. The Program Directory contains information concerning the material and procedures associated with the installation of the product code. You can download the program directory in PDF format from the WebSphere Process Server for z/OS download page, at http://www-306.ibm.com/software/integration/wps/library/infocenter/.

If the product delivery package was an IBM SystemPac / ServerPac, the system programmer copies SMP/E data sets that correspond to the CustomPac service level onto the z/OS system.

The following illustration depicts the methods supported for loading the contents from the installation media onto the z/OS system.

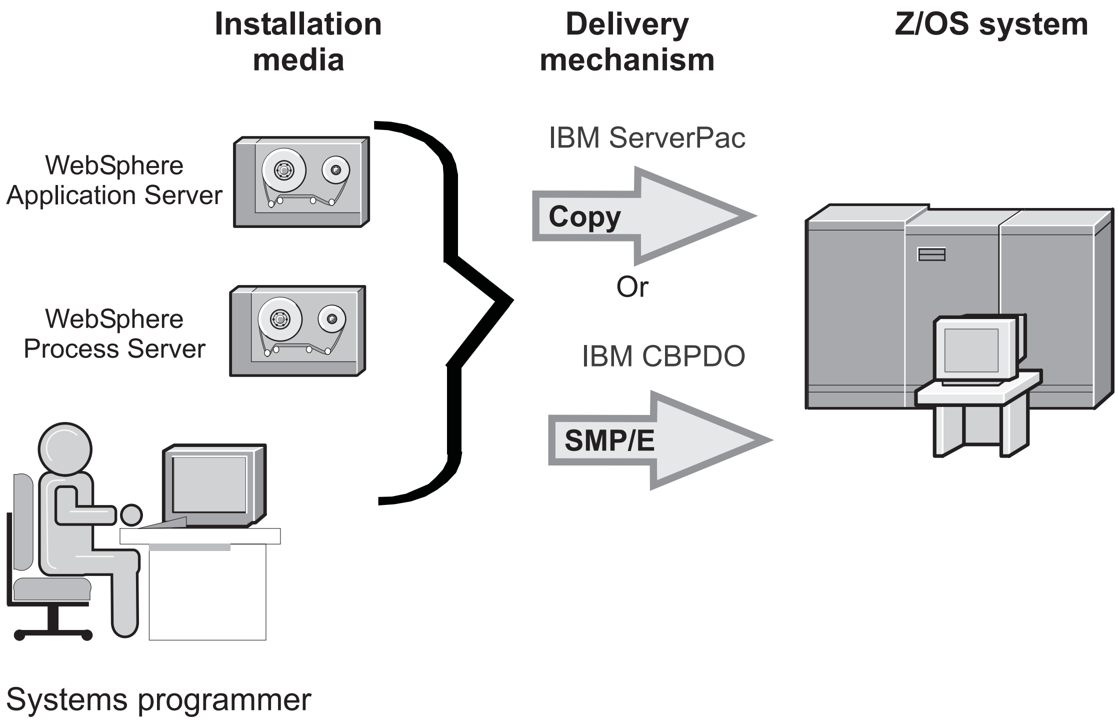


Figure 3. Loading the product code onto z/OS

Note: This phase of the installation assumes that the correct version of WebSphere Application Server for z/OS has been installed and customized. The SMP/E installation process checks the version level of WebSphere Application Server for z/OS to ensure version compatibility. If the version of WebSphere Application Server for z/OS is not at the level required to support WebSphere Process Server, the system programmer will need to take the appropriate actions to update the version of WebSphere Application Server for z/OS. For information on how to apply service to WebSphere Application Server for z/OS, see Applying product maintenance in the WebSphere Application Server for z/OS information center.

The result of completing the first phase of the installation is a read-only installation HFS, sometimes referred to as the `smpe_root`.

After successfully loading the product code from the installation media onto the system, the WebSphere administrator implements phase 2 of the installation by running the installation script.

About System Modification Program/Extended (SMP/E)

Here are some facts about SMP/E:

- SMP/E is an integral part of the product installation, service, and maintenance process on z/OS.
- SMP/E can be used to install and service any software that is packaged in SMP/E system modification (SYSMOD) format.
- SMP/E can be run either using batch jobs or using dialogs under Interactive System Productivity Facility/Program Development Facility (ISPF/PDF).
- SMP/E dialogs help you interactively query the SMP/E database, as well as create and submit jobs to process SMP/E commands.

- The guidance for doing the SMP/E installation is a program directory.

The system on which SMP/E runs is called the *driving system*.

The product code is loaded from the tape into a directory referred to as the *smpe_root*. The term *smpe_root* and *installation HFS* are used synonymously in this documentation. The smpe_root is read-only and is on its own HFS so it can be mounted and unmounted independently.

WebSphere Application Server for z/OS and WebSphere Process Server for z/OS each has its own smpe_root.

For WebSphere Process Server for z/OS the default path name of the smpe_root is `/usr/lpp/zWPS/V6R0`.

For WebSphere Application Server for z/OS the default path name of smpe_root is `/usr/lpp/zWebSphere/V6R0`.

Note: When you run the WebSphere Process Server for z/OS installation script, the keyword **-smproot** is followed by the fully-qualified path name of the WebSphere Process Server for z/OS smpe_root directory as created by the SMP/E installation.

Phase 2 - Running the installation script

In phase 2 of the installation process, the WebSphere administrator creates the required product definitions by running the installation script, **zSMPInstall.sh** from a command line.

As a result of completing this phase of the installation, the product administrator will have:

- Created the symlinks from the WebSphere Application Server for z/OS configuration HFS directory to the WebSphere Process Server for z/OS smpe_root .
- Invoked Ant script actions that update the administrative console with the WebSphere Process Server for z/OS product definitions.
- Run applyPTF processing as required.

The installation script updates the configuration HFS (sometimes referred to as a *configuration_root*) for each server instance. The default configuration HFS for WebSphere Application Server for z/OS is `/WebSphere/V6R0`.

The configuration root is a writable HFS that holds customized configuration documents and files for the configured product installation.

Note: The configuration HFS contains the various *app_server_root* directories (`/AppServer` and `/DeploymentManager`) and certain symbolic links associated with them. Each different node type under the configuration HFS requires its own cataloged procedures under z/OS.

The configuration HFS also has symbolic links (sometimes referred to as *symlinks*) to the smpe_root.

The symlinks point to read-only files from the smpe_root, such as JAR files, shell scripts, and so on. SMP/E updates the smpe_root, and the post -SMP/E configuration tasks update the configuration HFS on a per-node basis during the start-up of the product.

After running the installation script successfully, the WebSphere administrator can begin to configure the product for use. For information on the product configuration process, see About product configuration on z/OS.

The figure below shows an installation process that includes loading the contents of the tape via SMP/E and creating the product definitions by running the installation script named **zSMPInstall.sh**.

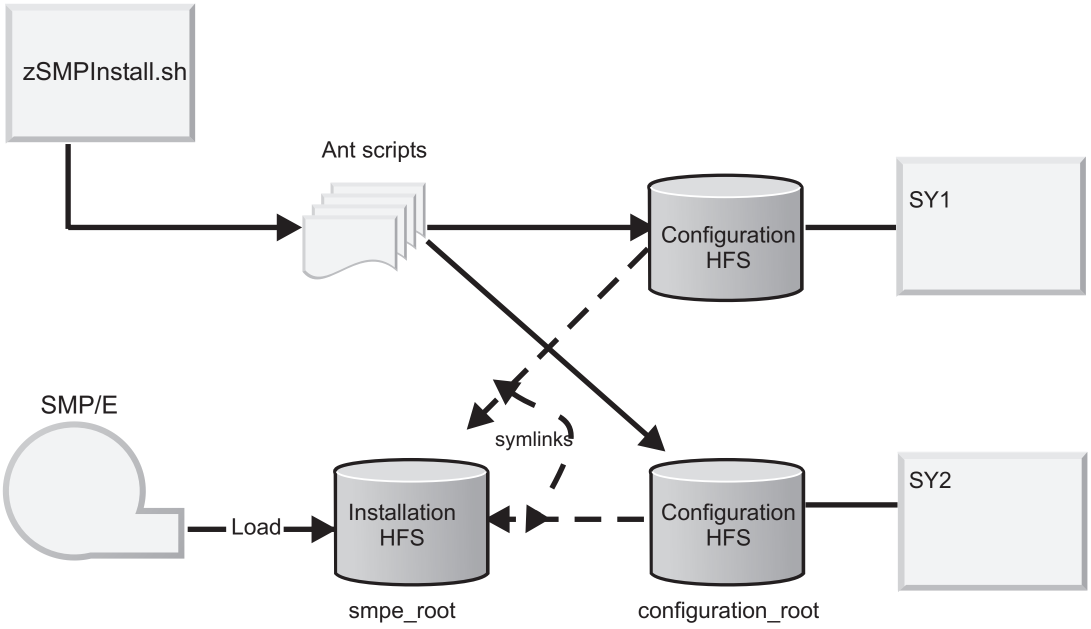


Figure 4. Process flow of WebSphere Process Server for z/OS installation

Notes:

- The read-only shared installation HFS is shown as shared, however, the system programmer who runs the installation can decide to share the disks or create copies of them. This decision has no impact on the post-installation configuration steps.
- The HFS is installed in the smpe_root, which means that the unpacking steps are controlled by SMP/E and the service of the code is also controlled by SMP/E.

About the installation script

The installation script creates the WebSphere Process Server for z/OS definitions that enable the product for use.

The commands described in this article are contained in, and should be run from the fully qualified WebSphere Process Server for z/OS path name as created by the SMP/E installation.

The default for the fully qualified path named is as follows:/usr/lpp/zWPS/ V6R0/zos.config/bin.

An administrator can run the installation script **(zSMPInstall.sh)** with different command arguments.

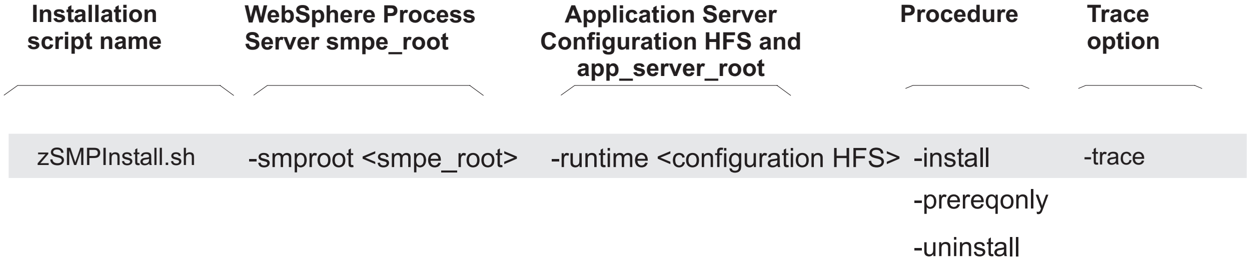
The script invokes a series of actions against the product code that was installed via SMP/E.

Running **zSMPInstall.sh** creates symbolic links and updates the administrative console with WebSphere Process Server for z/OS product definitions.

You run the uninstall command from the same script that you use to install WebSphere Process Server for z/OS. For details on the **-uninstall** command, see About the uninstall command.

Command line structure

The following illustration demonstrates the command line structure of **zSMPInstall.sh** by labeling the various portions of the command line and by providing an example of the key words and keyword parameters.


Installation script name
WebSphere Process Server smpe_root
Application Server Configuration HFS and app_server_root
Procedure
Trace option
zSMPInstall.sh
-smproot <smpe_root>
-runtime <configuration HFS>
-install
-prereqonly
-uninstall
-trace

Command syntax

The following syntax diagram illustrates all of the allowed syntax for the installation command.

Note: For information on how to read syntax diagrams, see Reading the syntax diagrams.

Installation script syntax diagram

```
►►──zSMPInstall.sh──-smproot── /usr/lpp/zWPS/V6R0─────────────────────────────►

                                                 (1)
►──-runtime──/WebSphere/V6R0M0/AppServer──────┬─-install─────┬────────────────►
                                              ├─-prereqonly──┤
                                              │          (2) │
                                              └─-uninstall───┘

                                    (3)
►──-trace──┬─trace arguments──┬───────────────────────────────────────────────►◄
           └─*=all=enabled────┘
```

Notes:

1 Type the keyword parameter that applies to your installation environment. The keyword parameters for -smproot and -runtime are user defined and

may vary from the default values shown in this example. The keyword parameter value for -smproot represents the smpe_root directory (installation HFS) for the WebSphere Process Server for z/OS product that resulted from the SMP/E installation. The keyword parameter value for -runtime represents the WebSphere Application Server for z/OS configuration HFS and app_server_root directory name combined (/WebSphere/V6R0/AppServer for example). 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 WebSphere Application Server for z/OS configuration process and contain symbolic links to files in the product directory. The -runtime path must match what was set when you created your WebSphere Application Server for z/OS application serving environment and will vary by configuration type. For example, the configuration root for a deployment manager may be `/WebSphere/V6R0/DeploymentManager`, while the configuration root for a stand-alone application server may be `/WebSphere/V6R0/AppServer`.

2 These arguments determine the product installation actions to be performed by **zSMPInstall.sh**. Choose one of these arguments to control how the product installation proceeds. The **-install** argument implements actions that update the administrative console with WebSphere Process Server for z/OS product definitions. For information on how to use the **-uninstall** command, see About the uninstall command.

3 By including **-trace** the command writes to the appropriate trace file.

Installation script command line arguments and parameters

Run **zSMPInstall.sh** from command line, incorporating the sequence of command arguments necessary to install the product definitions.

When you run **zSMPInstall.sh**, you are extending WebSphere Process Server for z/OS features into the directory specified by the **-runtime** parameter value. The -runtime parameter value is the configuration HFS and app_server_root directory name combined (/WebSphere/V6R0/AppServer for example) for WebSphere Application Server for z/OS.

Results will vary based on the arguments and parameters incorporated into the command.

The following table describes the installation script command arguments (keywords and parameters) and the actions that result. For a more detailed description of the various actions that result from running the zSMPInstall.sh, see the **Installation actions** section that follows this table.

Table 3. Command line arguments for the installation script

Keywords and parameters	**Description**
-smproot	This keyword and its associated keyword parameter are required. The parameter value is the WebSphere Process Server for z/OS installation Hierarchical File System (HFS) directory, also referred to as the *smpe_root* that resulted from the SMP/E installation of WebSphere Process Server for z/OS. The standard path to this directory is:`/usr/lpp/zWPS/V6R0`.
-runtime	This keyword and its associated keyword parameter are required. The parameter value is the configuration HFS and app_server_root directory name combined (/WebSphere/V6R0/AppServer for example) for WebSphere Application Server for z/OS. The standard path to this directory for a stand-alone configuration is`/WebSphere/V6R0M0/AppServer`. The standard path to this directory for a deployment manager configuration is`/WebSphere/V6R0/DeploymentManager`.
The following keywords are mutually exclusive. Specify one, and only one in the script command: • -prereqonly • -install • -uninstall	**-prereqonly** verifies arguments and environment. Results are: • Validates prerequisites • Verifies arguments **-install** results are: • Validates prerequisites (see **-prereqonly**) • Creates *symbolic links* from the WebSphere Process Server read-only HFS directories to the WebSphere Application Server for z/OS **/lib** and **/bin** directories, installing the WebSphere Process Server definitions. • Creates post installer file • Updates code base permissions • Enables WebSphere Process Server features by running Configuration Manager scripted actions. This will create any new administrative console plugin extensions. These resources include the following: – Business Process Choreographer – WebSphere Process Server core resources – Common Event Infrastructure (CEI) **-uninstall**. For information on how to use the **-uninstall** command, see About the uninstall command.

Table 3. Command line arguments for the installation script (continued)

Keywords and parameters	Description
-trace	The -trace keyword is optional. If you incorporate this argument into the script command, you must enter a parameter as qualifier to this argument. If you choose not to use the -trace argument, then ″*=all=disabled″ is used as the trace specification. Valid -trace parameters are as follows: • *=all=enabled To record trace information on all trace-enabled Java classes • Installer=all=enabled To record trace information on only the Installer group of Java classes

Installation actions

Verifying the arguments and the environment

The *Verifying the arguments and the environment*, encompasses the following actions related to installing the product and enabling the product for use:

1. Verify that the **-smproot** argument and its parameter value are specified and that the smproot directory specified in the value exists.
2. Verify that the **-runtime** argument and its parameter value are specified and the runtime directory specified in the value exists.
3. Verify that one and only one of the product installation arguments (-prereqonly, -install or -uninstall) is specified.
4. If the -trace keyword is specified, verify a value is also specified.

   If the -trace keyword is not specified, set to default ″*=all=disabled″.
5. Verify that any unrecognized keywords are not specified.
6. Create the trace file in the runtime directory if it does not exist.

   Default file name is */configuration_root/app_server_root*/logs/wbi/zSMPInstall.trace. For example,*/WebSphere/V6R0M0/AppServer*/logs/wbi/zSMPInstall.trace for a stand-alone configuration and */WebSphere/V6R0M0/DeploymentManager*/logs/wbi/zSMPInstall.trace for a deployment manager configuration.
7. Create the log file in the runtime directory.

   Default file name is */configuration_root/app_server_root*/logs/wbi/zSMPInstall.log. For example, */WebSphere/V6R0M0/AppServer*/logs/wbi/zSMPInstall.log for a stand-alone configuration and */WebSphere/V6R0M0/DeploymentManager*/logs/wbi/zSMPInstall.log for a deployment manager configuration.
8. Verify that the prerequisite WebSphere Application Server for z/OS is at required level.

Create symbolic links

The *Create symbolic links*, encompasses the following actions related to installing product and enabling the product for use:

1. Create a symbolic link file in the runtime directory for every file in the read-only WebSphere Process Server for z/OS smpe_root directory.

   For Example: For a stand-alone configuration, a file named */configuration_root/app_server_root*/profileTemplates/default.bfm/actionRegistry.xml (of type ″Syml″) is created that is a symbolic link to the SMP/E installation HFS file named /usr/lpp/zWPS/V6R0/profileTemplates/default.bfm/actionRegistry.xml .
2. Create a file in */configuration_root/app_server_root*/bin/zWESBConfig.sh that is a symbolic link to SMP/E installation HFS file named /usr/lpp/zWPS/V6R0/zos.config/bin/zWESBConfig.sh.

   Note: In the directory path above, */configuration_root/app_server_root* represents the configuration HFS and app_server_root directory name combined for WebSphere Application Server for z/OS. For example */WebSphere/V6R0M0/AppServer/* for a stand-alone server or */WebSphere/V6R0/DeploymentManger* for a deployment manager configuration.
3. Create file in */configuration_root/app_server_root*/bin/zWPSConfig.sh that is a symbolic link to SMP/E insstallation HFS file named /usr/lpp/zWPS/V6R0/zos.config/bin/zWPSConfig.sh

   Note: In the directory path above, */configuration_root/app_server_root* represents the configuration HFS and app_server_root directory name combined for WebSphere Application Server for z/OS. For example */WebSphere/V6R0M0/AppServer/* for a stand-alone server or */WebSphere/V6R0/DeploymentManger* for a deployment manager configuration.

Create post installer file

The *Create post installer file*, encompasses the following actions to installing product and enabling the product for use:

1. Create the service product directory, if it does not already exist.

   Default file name is */configuration_root/app_server_root*/properties/service/product/WBI. For example, */WebSphere/V6R0M0/AppServer*/properties/service/product/WBI for a stand-alone configuration and */WebSphere/V6R0M0/DeploymentManager*/properties/service/product/WBI for a deployment manager configuration.
2. Create the service product backup directory, if it does not already exist.

   Default file name is */configuration_root/app_server_root*/properties/service/product/WBI/backup. For example, */WebSphere/V6R0M0/AppServer*/properties/service/product/WBI/backup for a stand-alone configuration and */WebSphere/V6R0M0/DeploymentManager*/properties/service/product/WBI/backup for a deployment manager configuration.
3. Create the service log directory, if it does not already exist.

   Default name is /WebSphere/V6R0M0/AppServer/properties/service/logs/WBI

   Default file name is */configuration_root/app_server_root*/properties/service/logs/WBI. For example, */WebSphere/V6R0M0/AppServer*/properties/service/logs/WBI for a stand-alone configuration and */WebSphere/V6R0M0/DeploymentManager*/properties/service/logs/WBI for a deployment manager configuration.

4. Copy service level properties from the WebSphere Process Server for z/OS smpe_root read-only directory - default name is /usr/lpp/zWPS/V6R0/properties/service/product/WBI/service/service-level.properties, to the service product directory - default name is: */configuration_root/app_server_root*/properties/service/product/WBI/service-level.properties.

   Note: In the service product directory path above, */configuration_root/app_server_root* represents the configuration HFS and app_server_root directory name combined for WebSphere Application Server for z/OS. For example */WebSphere/V6R0M0/AppServer/* for a stand-alone server or */WebSphere/V6R0/DeploymentManger* for a deployment manager configuration.
5. Create a symbolic link file in the runtime directory - default name is */configuration_root/app_server_root*/properties/service/product/WBI/service, to the read-only installation HFS directory service directory - default name is /usr/lpp/zWPS/V6R0/properties/service/product/WPS/service/.

   Note: In the runtime directory path above, */configuration_root/app_server_root* represents the configuration HFS and app_server_root directory name combined for WebSphere Application Server for z/OS. For example */WebSphere/V6R0M0/AppServer/* for a stand-alone server or */WebSphere/V6R0/DeploymentManger* for a deployment manager configuration.
6. Create an Installer properties file in the runtime directory - default name is */configuration_root/app_server_root*/properties/service/product/WBI/zWPSPostInstaller.properties.

   Note: In the directory path above, */configuration_root/app_server_root* represents the configuration HFS and app_server_root directory name combined for WebSphere Application Server for z/OS. For example */WebSphere/V6R0M0/AppServer/* for a stand-alone server or */WebSphere/V6R0/DeploymentManger* for a deployment manager configuration.

   The **zWPSPostInstaller.properties** file written contains the following lines.

   Note: Some lines, such as WBI_TIME.created and WBI_BUILD_LEVEL.current_level, will contain variable data.

```
#---------------------------------------------------------------------------
# WPS Post Installer Properties File
#
# WebSphere Process Server 6.0
#
# Licensed Materials - Property of IBM
# 5655-N53
# (C) Copyright IBM Corporation 2006. All Rights Reserved.
# US Government Users Restricted Rights - Use, duplication or disclosure
# restricted by GSA ADP Schedule Contract with IBM Corp.
#
#---------------------------------------------------------------------------
WBI_TIME.created=Apr 5, 2006 3:53:47 PM EDT
WBI_TIME.modified=
WBI_PATH.smpe_root=/usr/lpp/wbi/zWPS/V6R0
WBI_PATH.config_root=/WebSphere/V6R0M0/AppServer
WBI_PATH.service_root=/WebSphere/V6R0M0/AppServer/properties/service/product/WBI
WBI_PATH.backup_root=/WebSphere/V6R0M0/AppServer/properties/service/product/WBI/backup
WBI_BUILD_LEVEL.target_rollback_level=
WBI_BUILD_LEVEL.initial_install=o0611.10
WBI_BUILD_LEVEL.current_level=o0611.10
WBI_BUILD_LEVEL.last_component_install=o0611.10
```

Update code base permissions

The *Update code base permissions*, encompasses the following actions related to installing product and enabling the product for use:

1. Adds the SMP/E read-only installation HFS directory to the JVM options by inserting the following string: -Dwps.smpe.install.root=WPS SMP/E HFS in value clause for symbolic name WAS_SERVER_ONLY_default_jvm_options in the variables.xml file. Default location for variables.xml file is: /WebSphere/V6R0M0/AppServer/profiles/default/config/cells/SY1/nodes/SY1/variables.xml

   Example:

   ```
   <entries xmi:id="VariableSubstitutionEntry_1122997753573"
     symbolicName="WAS_SERVER_ONLY_default_jvm_options"
     value="-Dwps.smpe.install.root=/usr/lpp/zWPS/V6R0  ..."
   ```
2. Saves the original **variables.xml** file (i.e. without the above change) to file **variables.xml.wps** in the same directory.
3. Updates the **server.policy** by inserting the following lines in the **server.policy** file. Default server.policy location is: /WebSphere/V6R0M0/AppServer/profiles/default/properties/server.policy.

   ```
   // Added the following for WPS for z/OS
      grant codeBase "file:${wps.smpe.install.root}/-" {
         permission java.security.AllPermission;
       };
   ```
4. Saves the original server.policy file (i.e. without the above change) to file server.policy.wps in the same directory.

Update Configuration Manager

The *Update Configuration Manager*, encompasses the following actions related to installing product and enabling the product for use:

1. Rebuilds the administrative console on the WebSphere configuration HFS by invoking the Config Manager with the 6.0 full installation directory. The full installation directory (a symbolic link to the WebSphere Application Server for z/OS smpe_root directory) contains a number of Ant files to be executed. Default name for the installation directory is: /WebSphere/V6R0M0/AppServer/properties/version/install.wbi/6.0.0.0/config/full/install

   Ant scripts to be run include the following:

   ```
   90SConfigNoProfileFirstStepsESB.ant
   90SConfigNoProfileFirstStepsWBI.ant
   90SConfigureWSProfileForWBI.ant
   90SInstallCEI.ant
   90SUpdateJavaOptions.ant
   91SConfigNoProfileFirstStepsCharset.ant
   98SDeployBPCAdminConsolePlugins.ant
   98SDeployServerAdminConsolePlugins.ant
   99SDeployCoreAdminConsolePlugins.ant
   ```

About the uninstall command

You run the uninstall command from the same script that you use to install WebSphere Process Server for z/OS.

The uninstall actions are implemented by running zSMPInstall.sh with the **-uninstall** keyword.

Uninstall command line arguments and parameters

The following table describes the command arguments used to uninstall WebSphere Process Server for z/OS.

Table 4. Command line arguments for un-installing WebSphere Process Server for z/OS.

Description	Implementation method	Result
Remove the product definitions from the administrative console and un-augment the default profile.	Run **zSMPInstall.sh** from command line, incorporating the sequence of command arguments necessary to uninstall the product. To uninstall WebSphere Process Server for z/OS, run **zSMPInstall.sh** with the following command arguments and parameters: • **-smproot** This keyword and its associated keyword parameter are required. The parameter value is the WebSphere Process Server for z/OSsmpe_root directory (installation HFS) that resulted from the SMP/E installation. The standard path to this directory is:*/usr/lpp/zWPS/V6R0* • **-runtime** This keyword and its associated keyword parameter are required. The parameter value is the configuration HFS and app_server_root directory name combined for WebSphere Application Server for z/OS. The standard path to this directory for a stand-alone configuration is*/WebSphere/V6R0M0/AppServer*. The standard path to this directory for a deployment manager configuration is*/WebSphere/V6R0/DeploymentManager*. • **-uninstall** The uninstall keyword used to drive the uninstall actions. • **-response** Because the uninstall actions include un-augmenting the default profile, you must include the -response keyword. • **-trace** This is an optional argument. If you incorporate this argument into the script command, you must enter a parameter as qualifier to this argument.	Removes the product features from the WebSphere Process Server for z/OS product directory. **-uninstall** results are: • Disables WebSphere Process Server features by running Configuration Manager scripted actions. This will remove any administrative console plug-in extensions. • Remove profile augmentation using WSPROFILE scripted actions. This will un-augment the default profile. • delete post install file • remove code base permissions **-trace** results are: If you choose not to use the -trace argument, then "*=all=disabled" is used as the trace specification. Valid -trace parameters are as follows: • *=all=enabled To record trace information on all trace-enabled Java classes • Installer=all=enabled To record trace information on only the Installer group of Java classes

Command syntax

The following syntax diagram illustrates the allowed syntax for the uninstall command.

Note: For information on how to read syntax diagrams, see Reading the syntax diagrams.

Uninstall script syntax diagram

```
►►──zSMPInstall.sh──-smproot── /usr/lpp/zWPS/V6R0─────────────────────────────►


                                       (1)            (2)
►──-runtime──/WebSphere/V6R0M0/AppServer──────-uninstall──────────────────────►

►──┬─-response─────────────────────────────────────────────────────┬─────────►
   │                                                          (3)  │
   └─┬─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp─────┬───┘
     ├─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp──┤
     ├─/usr/lpp/zWPS/V6R0/zos.config/dmgrProfile.rsp───────────┤
     └─/usr/lpp/zWPS/V6R0/zos.config/managedProfile.rsp────────┘


                                (4)
►──-trace──┬─trace arguments─┬───────────────────────────────────────────────►◄
           └─*=all=enabled───┘
```

Notes:

1 Type the keyword parameter that applies to your installation environment. The keyword parameters for -smproot and -runtime are user defined and may vary from the default values shown in this example. The keyword parameter value for -smproot represents the smpe_root directory (installation HFS) for the WebSphere Process Server for z/OS product that resulted from the SMP/E installation. The keyword parameter value for -runtime represents the WebSphere Application Server for z/OS configuration HFS and app_server_root directory name combined (/WebSphere/V6R0/AppServer for example). 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 WebSphere Application Server for z/OS configuration process and contain symbolic links to files in the product directory. The -runtime path must match what was set when you created your WebSphere Application Server for z/OS application serving environment and will vary by configuration type. For example, the configuration root for a deployment manager may be `/WebSphere/V6R0/DeploymentManager`, while the configuration root for a stand-alone application server may be `/WebSphere/V6R0/AppServer`.

2 The **-uninstall** argument implements actions that remove the product definitions from the administrative console and un-augment the default profile.

3 Type the absolute path of the response file that is associated with the profile that you are un-augmenting.

4 By including **-trace** the command writes to the appropriate log file.

Uninstall actions

Revert Configuration Manager

The **Revert Configuration Manager**, encompasses the following actions related to un-installing the product and un-augmenting the default profile:

1. Reverts the adminconsole program to pre-install state by invoking the Config Manager with the 6.0 full uninstall directory.

   The full uninstall directory (a symbolic link to the WebSphere Application Server for z/OS installation HFS) contains a number of Ant files to be executed.

   Default name for the install directory is: /WebSphere/V6R0M0/AppServer/properties/version/install.wbi/6.0.0.0/config/full/uninstall

   Ant scripts to be run include the following:

   ```
   90SDeleteFirstStepsFilesESB.ant
   90SDeleteFirstStepsFilesWBI.ant
   90SRemoveJavaOptions.ant
   90SUninstallCEI.ant
   98SUndeployBPCAdminConsolePlugins.ant
   98SUndeployServerAdminConsolePlugins.ant
   99SUndeployCoreAdminConsolePlugins.ant
   ```

Reading the syntax diagrams

Use the conventions described here to interpret the syntax diagrams.

In this documentation the commands for the WebSphere Process Server for z/OS installation script and configuration script are documented in syntax diagrams.

A syntax diagram is a visual representation for a command that indicates how to enter the command on the command line. The graphical nature of a syntax diagram allows you to understand the connectivity of concepts through the structure of the command grammar.

To read the syntax diagrams, start at the top leftmost line and read from left to right and from top to bottom. The horizontal line represents the main path of parameters that you use when you enter the commands. Parameters that are off the main path are optional and may have a default, if you omit them.

Note: The syntax diagrams in this documentation are examples only. Although the syntax diagrams are descriptive of the commands that you run to install and configure WebSphere Process Server for z/OS, some of the values that you will enter on the command line are variable.

Symbols and punctuation

Syntax diagrams use the following symbols:

Symbol	Description
>>-	Marks the beginning of the command syntax.
>- -	Indicates that the command syntax is continued.
|	Marks the beginning and end of a fragment or part of the command syntax.
><	Marks the end of the command syntax.

You must include all punctuation such as colons, semicolons, commas, quotation marks, and minus signs that are shown in the syntax diagram.

Parameters

The following types of parameters are used in syntax diagrams:

Required
Required parameters are displayed on the main path.

Optional
Optional parameters are displayed below the main path.

Parameters are classified as keywords or variables. You must enter keywords exactly as written in the diagram. The diagrams display variables, which are names or values you supply. In the examples these variables display in italics.

Table 5. Description of syntax diagram items

Syntax Diagram Description	Example
keywords: Keywords are what drive the actions of the command line. Keywords are preceded by a dash (-) and are followed by a space and either another keyword, or a keyword parameter.	>>-keyword1 -keyword2------------->< >>-keyword *keyword parameter*---><
Symbols: Enter these symbols exactly as they appear in the syntax diagram.	- Hyphen

Optional choices:
When two or more items are in a stack and one of them is on the line, you must select one item.
In this example, you must select either A, B, or C.
	>>-+-A-+--------------------------------><
	   +-B-+
	   '-C-'

Variables: Italicized lowercase items (*var_name*) indicate variables. In this example, you can specify a var_name when you enter the keyword command.	>>-keyword--*var_name*-----------------><

Examples of command syntax

The following are examples of WebSphere Process Server for z/OS installation and configuration commands that adhere to the syntax diagram guidelines described in the previous section.

Installing the product

```
zSMPInstall.sh -smproot /usr/lpp/zWPS/V6R0 -runtime /WebSphere/V6R0M0/AppServer
-install
```

In the example above, the shell script command includes the keyword of **-smproot** and its keyword parameter value, which represents the smpe_root directory (installation HFS) for the WebSphere Process Server for z/OS product that resulted from the SMP/E installation (*/usr/lpp/zWPS/V6R0*). The command also includes the **-runtime** keyword and its associated keyword parameter, which represents the WebSphere Application Server for z/OS configuration HFS and app_server_root directory name combined (*WebSphere/V6R0M0/AppServer*). The directory paths are italicized because they are variable.

Configure the product for a stand-alone server

```
zWPSConfig.sh -augment -response /usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp
```

In the example above, the configuration command includes the **-augment** keyword to perform profile augmentation actions, which are Ant scripts that get called with the properties in the response file. The **-augment** keyword is followed directly by the **-response** keyword and its associated keyword parameter (the absolute path of the response file). Notice that the response file path is italicized because it is variable.

A syntax diagram showing all of the options for product configuration would display as follows:

Syntax diagram for configuring WebSphere Process Server for z/OS

```
►►──zWPSConfig.sh──-augment──────────────────────────────────────────────────►


   (1)
►──────┬-response──────────────────────────────────────────┬─────────────────►
       └─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp──────┘
        ├─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp─┤
        └─/usr/lpp/zWPS/V6R0/zos.config/dmgrProfile.rsp──────────┘

►─-Z <Name of the value in the property file to override>.───────────────────►


►─-trace (By including -trace the command writes to the appropriate log file)──────►◄
```

Notes:

1 Type the absolute path of the response file that is associated with the configuration onto which you are installing WebSphere Process Server for z/OS. This absolute path name in this example assumes the installer is using the default response files. In most cases the user will make a copy of the default response file and edit the values to suit the needs of their environment. So, in most cases the path name would represent the absolute path of a customized response file.

Notice the keyword parameter values for **-response**. They are stacked in the syntax diagram to signify that you have choice with regard which response file to include in your command.

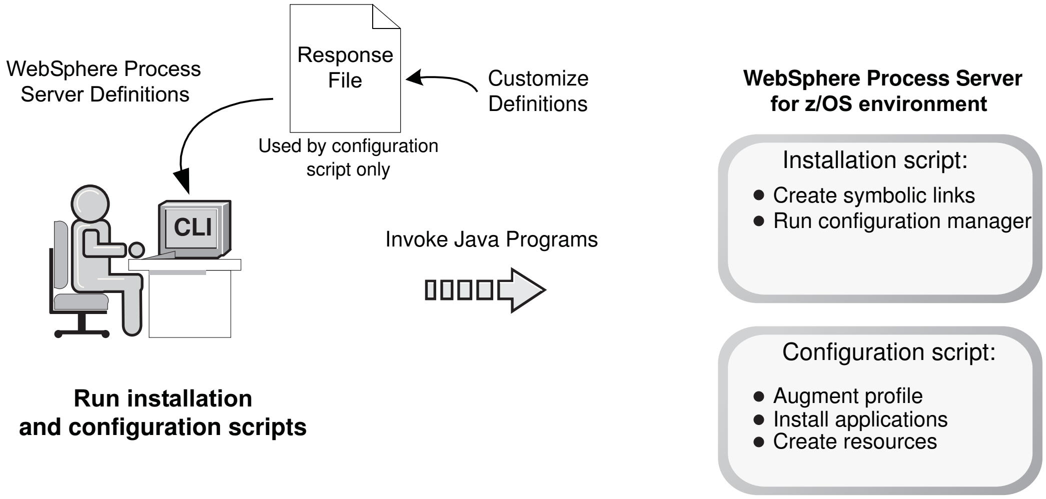
About the command line procedure

You install and configure WebSphere Process Server for z/OS by running shell scripts from the command line.

To perform installation and configuration tasks using scripts, you establish a Telnet session with your z/OS system and then run the scripts. The installation script (**zSMPInstall.sh**) and either the configuration script (**zWPSConfig.sh**) or the configuration script (**zWESBConfig.sh**) use input from a response file to determine how to set up the runtime environment.

Note: Alternatively, you can run the scripts from an OS/390 UNIX command shell by entering the TSO command OMVS at the ISPF Command Shell or TSO OMVS from any other ISPF panel.

The *response file* is an EBCDIC file that can be customized with setup and configuration data. A response file cannot be used interactively, and therefore you must supply the environment definitions you will use before running a script.


WebSphere Process Server Definitions
Response File
Customize Definitions
Used by configuration script only
CLI
Invoke Java Programs
Run installation and configuration scripts
WebSphere Process Server for z/OS environment
Installation script:
● Create symbolic links
● Run configuration manager
Configuration script:
● Augment profile
● Install applications
● Create resources

When you run a script using the command-line interface (CLI), the script invokes Java programs in the z/OS environment. The programs perform tasks needed to install and configure the runtime environment, including creating symbolic links and running the configuration manager and augmenting the default profile by running dozens of ant scripts to install applications, create resources, buses etc.

Because the configuration script relies on WebSphere Process Server definitions that are specified in the response file, the command line procedure is sometimes referred to as a *silent* or *unattended* installation.

About the upgrade process

Upgrading WebSphere Process Server for z/OS is a multiphase process that can span multiple roles.

You can upgrade the WebSphere Process Server for z/OS product using one of these methods:

- Overlaying an existing Installation HFS with a newer product version
- Running the upgrade script to update an older version to a newer installed product version

Scenario 1: Overlaying an existing Installation HFS with a newer version

In this scenario, you upgrade WebSphere Process Server for z/OS by using SMP/E to load the newer version of the product over the existing Installation HFS (Figure 1).

Note: A configuration has access only to its Configuration HFS. Symbolic links in the Configuration HFS provide access to the code in the Installation HFS.

Subsequently, when the server controller starts, the *applyPTF.sh* script is run. It checks the level of the Configuration HFS against the level of the Installation HFS. If the two are at the same maintenance level, the server is started. If the Configuration HFS is at a *lower* level than the Installation HFS, then the *applyPTF.sh* script performs the changes to the Configuration HFS specified by the maintenance level of the Installation HFS and then starts the server.

Note: If the Configuration HFS is at a higher level than the Installation HFS, for example in the case where maintenance is backed off, the server is not allowed to start.

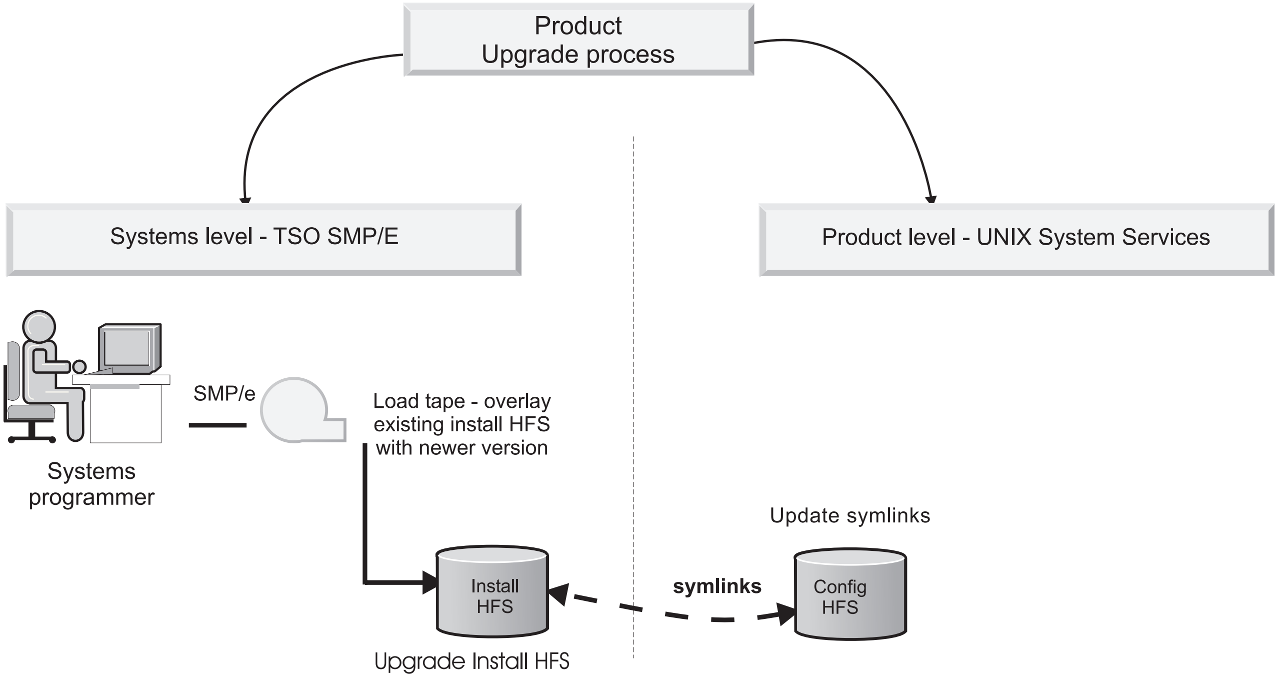

Product
Upgrade process
Systems level - TSO SMP/E
Product level - UNIX System Services
SMP/e
Load tape - overlay existing install HFS with newer version
Systems programmer
Update symlinks
Install HFS
symlinks
Config HFS
Upgrade Install HFS

Figure 5. Upgrading WebSphere Process Server for z/OS product by overlaying the existing Installation HFS

Scenario 2: Using the upgrade script

In this scenario, you upgrade WebSphere Process Server for z/OS by using SMP/E to load the newer version of the product separate from the existing Installation HFS (Figure 2).

You can then run the upgrade script for each application server that you wish to upgrade. The upgrade changes the service level symlink from the existing Installation HFS to the newer Installation HFS for the application server.

Subsequently, when the server controller starts, the *applyPTF.sh* script is run. It checks the level of the Configuration HFS against the level of the Installation HFS. In this case, the Configuration HFS is at a *lower* level than the Installation HFS, and the *applyPTF.sh* script performs the changes to the Configuration HFS specified by the maintenance level of the Installation HFS and then starts the server.

In Figure 2, the newer version of the WebSphere Process Server for z/OS has been installed and configured to run on application server A. To upgrade application server B, the systems programmer can run the upgrade script. The script updates the application server B Configuration HFS by pointing its service level symlink to the new Installation HFS associated with application server A. The installation upgrade is completed when the *applyPTF.sh* script is run.

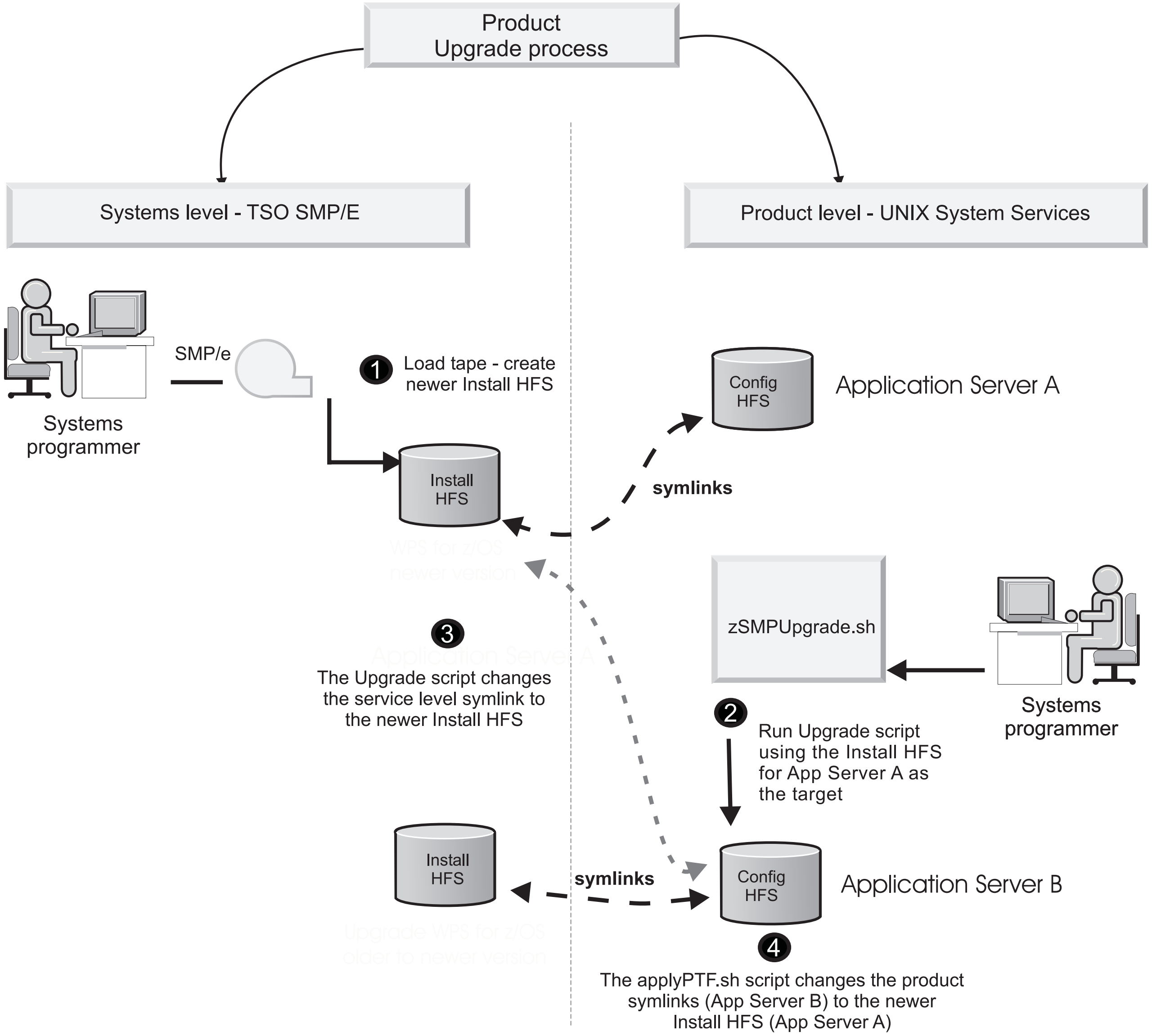

Product
Upgrade process
Systems level - TSO SMP/E
Product level - UNIX System Services
Systems programmer
SMP/e
1
Load tape - create newer Install HFS
Install HFS
WPS for z/OS newer version
Config HFS
Application Server A
symlinks
zSMPUpgrade.sh
Systems programmer
3
Application Server A
The Upgrade script changes the service level symlink to the newer Install HFS
2
Run Upgrade script using the Install HFS for App Server A as the target
Install HFS
symlinks
Config HFS
Application Server B
Upgrade WPS for z/OS older to newer version
4
The applyPTF.sh script changes the product symlinks (App Server B) to the newer Install HFS (App Server A)

Figure 6. Upgrading the WebSphere Process Server for z/OS product using the upgrade script

Using the upgrade script with intermediate symbolic links

This example is similar to scenario in Figure 2, except that rather than a direct symlink between the Installation HFS and Configuration HFS, an intermediate symbolic link is used. The intermediate symbolic link is a standard symlink that points to the Installation HFS and the Configuration HFS points to the intermediate symbolic link. Changing a node to another service level simply involves changing the single intermediate symbolic link.

You can run the upgrade script for each application server that you wish to upgrade. The script uses the newer version of the installed HFS to update the symlinks for the Configuration HFS that you want to update. There is an additional layer of indirection, in that the Configuration HFS points to the intermediate symbolic link.

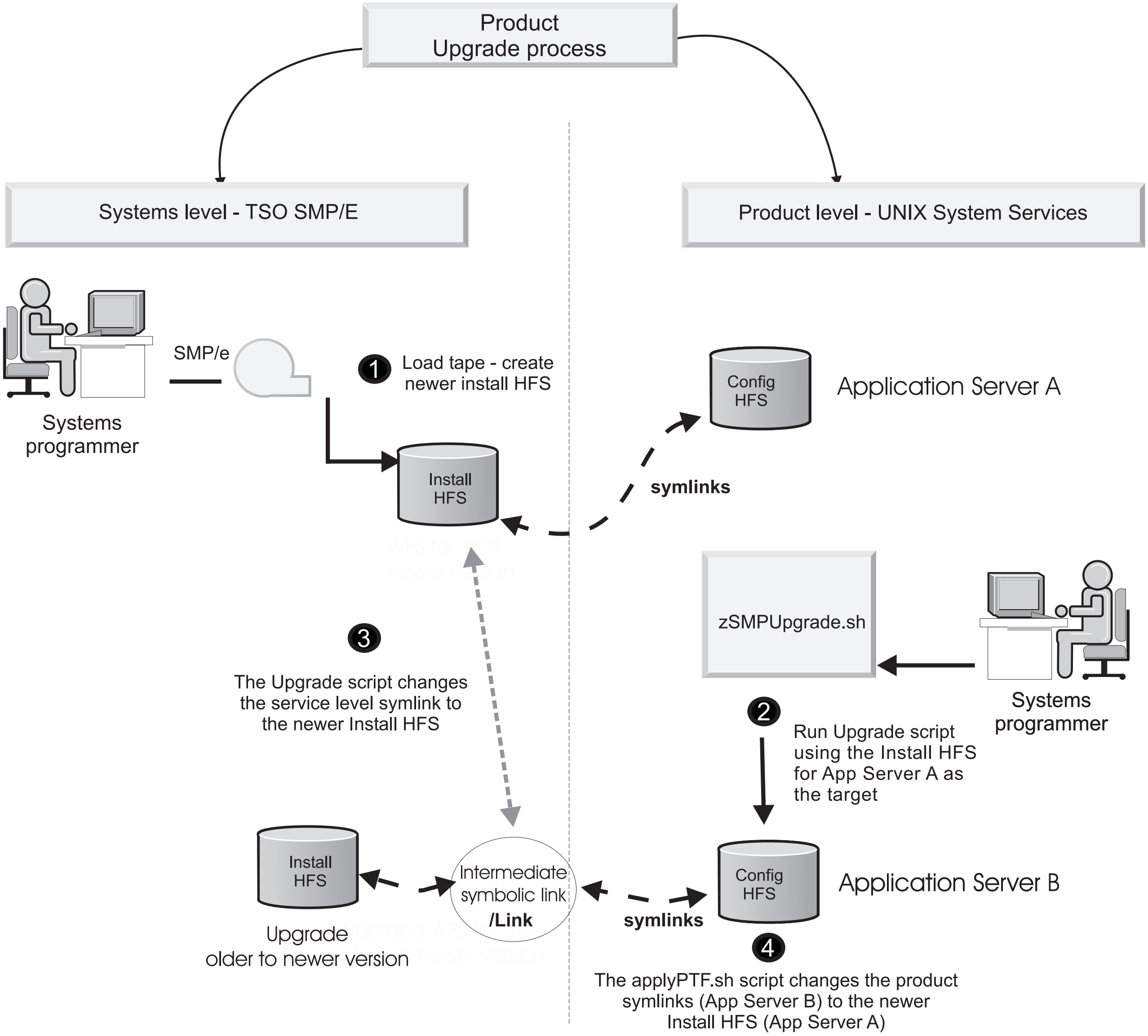


Figure 7. Upgrading the WebSphere Process Server for z/OS product using the upgrade script

About the upgrade script

When run, the upgrade script performs actions to update the WebSphere Process Server for z/OS product service level to a newer version.

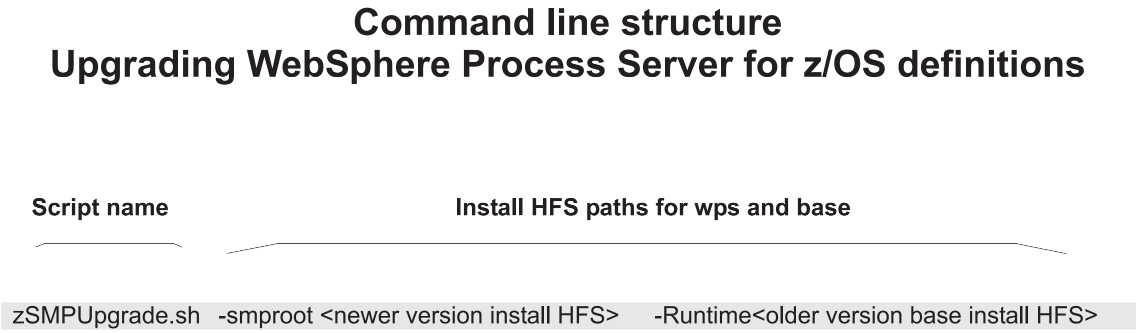
The commands described in this topic are contained in, and should be run from the WebSphere Process Server for z/OS SMP/E installation HFS. The default installation HFS directory for the product is as follows:/usr/lpp/zWPS/V6R0/zos.config/bin.

An administrator can run the upgrade script **(zSMPUpgrade.sh)**.

The script uses the installation HFS path for the newer WebSphere Process Server for z/OS version as a source installation directory for the -smproot. This is used to re-configure the configuration HFS of the target application server that you wish to upgrade.

Command line structure

The following illustration demonstrates the command line structure of **zSMPUpgrade.sh** by labeling the various portions of the command line and by providing an example of the key words and keyword parameters.



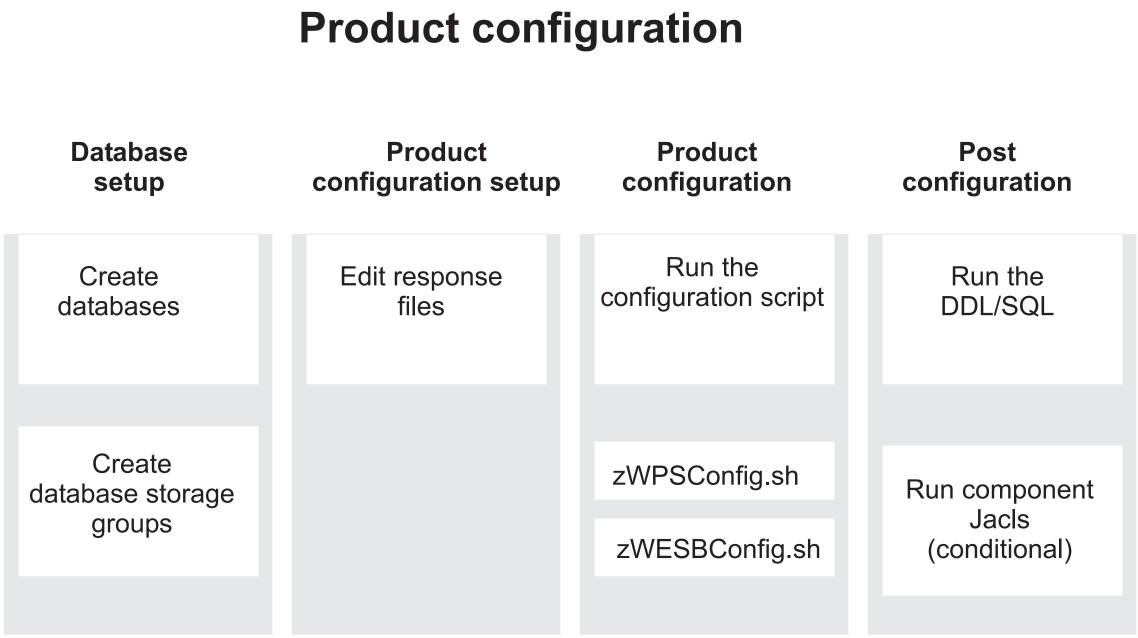
Command line structure
Upgrading WebSphere Process Server for z/OS definitions

```
zSMPUpgrade.sh   -smproot <newer version install HFS>     -Runtime<older version base install HFS>
```

About product configuration on z/OS

Configuring WebSphere Process Server for z/OS involves running the configuration script from a command line to enable profile augmentation using scripted actions.

Before you configure WebSphere Process Server for z/OS you must first install, customize and configure WebSphere Application Server for z/OS. Also, prior to configuring WebSphere Process Server for z/OS, you need to create the appropriate database and storage groups for some database types. For information on how to create the databases and storage groups, see Creating databases and storage groups. For information on specifications for the databases used by WebSphere Process Server for z/OS, see Database specifications.



The product configuration script is run from a command line.

There are several configuration options, which on z/OS are driven by the WebSphere Process Server for z/OS configuration script's use of response files.

The content in the response file is used to augment the WebSphere Application Server for z/OS profile with WebSphere Process Server for z/OS configuration data.

Profile augmentation

Profile augmentation is the process by which WebSphere Process Server for z/OS configuration data is added to an existing WebSphere Application Server for z/OS default profile, transforming it into a WebSphere Process Server for z/OS profile.

Note: On z/OS, there is always one and only one profile and that profile is named **default** in each of the configurations.

Profile augmentation extends the capabilities of your WebSphere Application Server for z/OS product configuration by adding to it the features and functions of WebSphere Process Server for z/OS.

Before running the scripts that augment the default profile, one must edit the response file that is specifically associated with the WebSphere Process Server for z/OS configuration you want to create. The augment actions are Ant scripts that get called with the properties in the response files.

To configure WebSphere Process Server for z/OS, the product administrator runs one of the following configuration scripts:
- zWPSConfig.sh
- zWESBConfig.sh

The two configuration scripts allow for flexibility when configuring WebSphere Process Server for z/OS. Run the script **zWESBConfig.sh** if your process server will run Mediations only. A configuration that runs Mediations only, reduces the footprint of the server by excluding unused features and enabling the server to start quicker and to consume less memory. Run **zWPSConfig.sh** if your process server will be used for business processes. The script **zWPSConfig.sh** is required if the application servers are configured for processes.

The manner in which profile augmentation occurs will vary by WebSphere Application Server for z/OS configuration type. The directory structure (which will vary by product configuration) is passed into the configuration manager portion of the installer. The configuration manager runs the Ant scripts in the `/actions` subdirectory, which results in the administrative console being updated with WebSphere Process Server definitions.

About the configuration script

Use the configuration script to augment the default profile.

WebSphere Process Server comes with two configuration scripts named **zWPSConfig.sh** and **zWESBConfig.sh** respectively.

The commands described in this topic are contained in, and should be run from the /WebSphere/V6R0M0/AppServer/bin directory when creating a stand-alone configuration and from /WebSphere/V6R0/DeploymentManager for a deployment manager configuration.

The two configuration scripts allow for flexibility when configuring WebSphere Process Server for z/OS. The script **zWESBConfig.sh** allows the product administrator to configure the server as an ″ESB-only″ configuration, reducing the footprint of the server by excluding unused features and enabling the server to start quicker and to consume less memory. The script **zWPSConfig.sh** configures your server as a process server, enabling the server to handle processes. The decision on which script to run is based on business objectives and server performance considerations. For information on enterprise service bus functionality, see *Overview of enterprise service bus* in the Product overview. For information on WebSphere Process Server functionality, see *Architectural overview of WebSphere Process Server* in the Product overview.

Depending on how you will use WebSphere Process Server for z/OS, you run one of the two configuration scripts, **zWPSConfig.sh** or **zWESBConfig.sh** from a command line prompt. The scripts support the same command arguments.

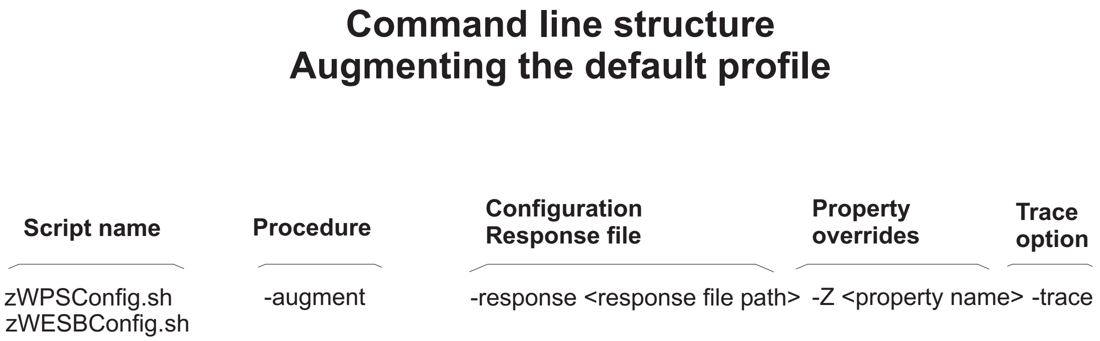
As a result of running either of the configuration scripts, the default profile is updated with WebSphere Process Server for z/OS configuration data.

If you choose to create an ESB-only configuration, you have the capability later on to change to a Process Server configuration. For more information, see About the reconfiguration process.

The command for running the configuration script includes the path to the response file. The property values that you set in the response file are used to create your product configuration. The choices you make when setting certain parameters in the response file may result in the need to perform additional post configuration tasks. For information on which parameter settings will result in having to perform post configuration tasks, see Working with response files.

Command line structure

The following illustration demonstrates the command line structure of the configuration scripts by labeling the various portions of the command line and by providing an example of the command arguments.



Command syntax

The following syntax diagram illustrates the command syntax of the configuration script.

Syntax diagram for configuring WebSphere Process Server for z/OS

```
►►──zWPSConfig.sh──-prereqonly──-augment──────────────────────────────────►


►──┬─-response────────────────────────────────────────────────────┬──────►
   │                                                      (1)     │
   └──┬─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp───┬───┘
      ├─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp─┤
      ├─/usr/lpp/zWPS/V6R0/zos.config/DmgrDB2.rsp──────────────┤
      └─/usr/lpp/zWPS/V6R0/zos.config/ManagedDB2.rsp───────────┘


►──-Z─(Name of the value in the property file to override)────────────────►


                              (2)
►──-trace──┬─trace arguments─┬────────────────────────────────────────────►◄
           └─*=all=enabled───┘
```

Notes:

1 Type the absolute path of the response file that is associated with the configuration you are creating. This example shows the default path to the response files packaged with the product. When you run the command, the **-response** parameter will represent the absolute path to the response file that you have modified with the values specific to your environment. The information required to create the DB2® for z/OS database in the response files named standAloneProfileDB2.rsp, DmgrDB2.rsp and ManagedDB2.rsp, contain properties that can be set to support DB2 for z/OS version 7 or DB2 for z/OS version 8. The default values are set for DB2 for z/OS version 8.

2 By including **-trace** the command writes to the appropriate trace file.

Syntax diagram for configuring WebSphere Process Server for z/OS as an ESB-only server

```
►►──zWESBConfig.sh──-prereqonly──-augment──────────────────────────────────►

►──┬─-response────────────────────────────────────────────┬────────────────►
   │                                                  (1) │
   └─┬─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp──┬─┘
     ├─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp─┤
     ├─/usr/lpp/zWPS/V6R0/zos.config/DmgrDB2.rsp──────────────┤
     └─/usr/lpp/zWPS/V6R0/zos.config/ManagedDB2.rsp───────────┘

►──-Z─(Name of the value in the property file to override)─────────────────►

                               (2)
►──-trace──┬─trace arguments─┬────────────────────────────────────────────►◄
           └─*=all=enabled───┘
```

Notes:

1 Type the absolute path of the response file that is associated with the configuration you are creating. This example shows the default path to the response files packaged with the product. When you run the command, the **-response** keyword and its parameter represent the absolute path to the response file that you have modified with the values specific to your environment. The information required to create the DB2 for z/OS database in the response files named standAloneProfileDB2.rsp, DmgrDB2.rsp and ManagedDB2.rsp, contain properties that can be set to support DB2 for z/OS version 7 or DB2 for z/OS version 8. The default values are set for DB2 for z/OS version 8.

2 By including **-trace** the command writes to the appropriate trace file.

Command line arguments and parameters

As a result of running the product configuration command, you augment the default profile with WebSphere Process Server for z/OS configuration data.

Run **zWPSConfig.sh** from a command line, incorporating the sequence of keywords and keyword parameters to achieve the desired configuration.

The following table describes the product configuration command arguments (keywords and parameters) and the result of running the script with the command arguments.

Table 6. Configuring WebSphere Process Server for z/OS - Running the server as a Process Server

Keywords and parameters	Description
–prereqonly	Validates the arguments and the response file properties. **Note:** The actions **–prereqonly** and **–augment** are mutually exclusive (you run one or the other – not both!)

Table 6. Configuring WebSphere Process Server for z/OS - Running the server as a Process Server (continued)

Keywords and parameters	Description
–augment	• Validates prerequisites • Verifies arguments • Enable profile augmentation using scripted actions. Augments the WebSphere Application Server for z/OS default profile with WebSphere Process Server for z/OS configuration data, as specified by property values in the response file.
-response	You must enter a keyword parameter as qualifier to the -response keyword. For example, **-response <absolute path of response file>** • The response file containing the properties for configuring WebSphere Process Server for z/OS. • Sample response files are provided for the following configurations: – Stand-alone using Cloudscape™ – Stand-alone using DB2 for z/OS – Managed node using DB2 for z/OS – Deployment Manager using DB2 for z/OS **Note:** This response files supporting DB2 for z/OS contain values for DB2 for z/OS version 7 and DB2 for z/OS version 8. The default values are set to DB2 for z/OS version 8.

-Z	**-Z**<propertyOverride>
	This is an optional keyword. If you incorporate this keyword into the script command, you must enter a parameter as qualifier to this argument.
	This option will allow the user to override any individual property specified in the provided response file. This will not update the response file.
	For example, the parameter and value of **nodeName=NODE1** displayed in the command line below overrides only the *nodeName* specified in the response file:
	`zWPSConfig.sh -augment`
	`  -response /web/usr/wbi/zWPS/V6R0/zos.config/standAloneProfile.rsp`
	`  -ZnodeName=NODE1'`
	This particular use of the -Z override would make sense if you wanted to run the product configuration script with the essentially the same properties on multiple nodes.
	Note: If you will be overriding more than 3 properties specified in the response file, the command line may become quite long. In such cases you should consider updating the response file to address the propertyOverride values.

-trace	You must enter a keyword parameter as qualifier to the -trace keyword. The default **-trace** specification is ″*=all=disabled″. To record trace information on all trace-enabled Java classes, specify ″*=all=enabled″. To record trace information on only the Installer group of Java classes, specify ″Installer=all=enabled″.

Run **zWESBConfig.sh** from a command line, incorporating the sequence of command keywords and keyword parameters necessary to achieve the desired configuration.

As a result of running the **zWESBConfig.sh** product configuration script, you augment the default profile with configuration data for enterprise service bus functionality.

The following table describes the product configuration command arguments (keywords and parameters) and the result of running the script with the command arguments.

Table 7. Configuring WebSphere Process Server for z/OS - Running the server as an ESB-only server

Keywords and parameters	**Description**
–prereqonly	Validates the arguments and the response file properties. **Note:** The actions **–prereqonly** and **–augment** are mutually exclusive (you run one or the other – not both!)
–augment	• Validates prerequisites • Verifies arguments • Enable profile augmentation using scripted actions. Augments the WebSphere Application Server for z/OS default profile with the configuration data for WebSphere Process Server for z/OS configured as an ESB-only server.
-response	You must enter a keyword parameter as qualifier to the -response keyword. For example, **-response <absolute path of response file>** • The response file containing the properties for configuring WebSphere Process Server for z/OS. • Sample response are provided for the following WebSphere Application Server for z/OS configurations: – Stand-alone using Cloudscape – Stand-alone using DB2 for z/OS – Managed node using DB2 for z/OS – Deployment Manager using DB2 for z/OS **Note:** This response files supporting DB2 for z/OS contain values for DB2 for z/OS version 7 and DB2 for z/OS version 8. The default values are set to DB2 for z/OS version 8.

Table 7. Configuring WebSphere Process Server for z/OS - Running the server as an ESB-only server (continued)

Keywords and parameters	Description

-Z	**-Z**<propertyOverride>
	This is an optional keyword. If you incorporate this keyword into the script command, you must enter a parameter as qualifier to this argument.
	This option will allow the user to override any individual property specified in the provided response file. This will not update the response file.
	For example, the parameter and value of **nodeName=NODE1** displayed in the command line below overrides only the *nodeName* specified in the response file:
	`zWESBConfig.sh -augment`
	` -response /web/usr/wbi/zWESB/V6R0/zos.config/standAloneProfile.rsp`
	` -Z`**`nodeName=NODE1`**`'`
	This particular use of the -Z override would make sense if you wanted to run the product configuration script with the essentially the same properties on multiple nodes.
	Note: If you will be overriding more than 3 properties specified in the response file, the command line may become quite long. In such cases you should consider updating the response file to address the propertyOverride values.

-trace	You must enter a keyword parameter as qualifier to the -trace keyword. The default **-trace** specification is ″*=all=disabled″. To record trace information on all trace-enabled Java classes, specify ″*=all=enabled″. To record trace information on only the Installer group of Java classes, specify ″Installer=all=enabled″.

Augmentation actions

Verify arguments and environment

Verifying the arguments and the environment encompasses the following actions:

1. Verify that one and only one of the arguments (-prereqonly or -augment) is specified.
2. Verify that the -response argument and its parameter value are specified and that the response file that is specified in the value exists.
3. Verify that any **-Z** arguments contain an equal sign (=).
4. If the -trace argument is specified, verify a value is also specified.

   If the -trace argument is not specified, set to default ″*=all=disabled″.
5. Verify that any unrecognized keywords are not specified.
6. Create the trace file in the runtime directory if it does not exist.

   For a WebSphere Process Server configuration, the default file name is /WebSphere/V6R0M0/AppServer/logs/wbi/zWPSConfig.trace

For an ESB-only server configuration, the default file name is /WebSphere/V6R0M0/AppServer/logs/wbi/zWESBConfig.trace.

7. Create the log file in the runtime directory.

   For a WebSphere Process Server configuration, the default file name is /WebSphere/V6R0M0/AppServer/logs/wbi/zWPSConfig.log

   For an ESB-only server configuration, the default file name is /WebSphere/V6R0M0/AppServer/logs/wbi/zWESBConfig.log.
8. Read the Response File and load its values to the response properties table.
9. Add or replace any -Z keyword command line values in the response properties table.
10. Verify that the profileName, profilePath, nodeName, cellName, and serverName properties are specified in the response properties table and that the properties specified are valid.

    Note: The serverName property is validated only if specified. The severName property is not required in the managed node response file.
11. Verify that the prerequisite WebSphere Application Server is at the supported level.

Augment profile(s)

Augmenting profile(s) encompasses the following actions:

1. Enables Profile Augmentation using (WSPROFILE) scripted actions.

   This is accomplished by invoking the Profile Updater and passing it a profileTemplate directory.

   Which profileTemplate is passed is based on the templatePath value in the response file, which is referenced in the **zWPSConfig.sh** or **zWESBConfig.sh** command line.

   As with all response file properties, the templatePath value may be overridden using the -Z argument on the command line.

   The default response file templatePath looks like this: templatePath=/WebSphere/V6R0M0/AppServer/profileTemplates/default.*, where **default.*** represents a wildcard that invokes the Profile Updater for all the WebSphere Process Server for z/OS profileTemplates.

   The templates for a stand-alone configuration include the following:

   ```
   /WebSphere/V6R0M0/AppServer/profileTemplates/default.wbicore
   /WebSphere/V6R0M0/AppServer/profileTemplates/default.bfm
   /WebSphere/V6R0M0/AppServer/profileTemplates/default.wbiserver
   ```

   The templates for a network deployment configuration include the following:

   ```
   /WebSphere/V6R0M0/DeploymentManager/profileTemplates/default.wbicore
   /WebSphere/V6R0M0/DeploymentManager/profileTemplates/default.bfm
   /WebSphere/V6R0M0/DeploymentManager/profileTemplates/default.wbiserver
   ```

   For more detailed information on the **profileTemplate** property, see the Response file samples.

About the reconfiguration process

You can re-configure an ESB-only server to a process server.

Running **zWPSConfig.sh** against an ESB-only configuration re-configures WebSphere Process Server for z/OS as a full functioning process server.

The reason for re-configuring an ESB-only server to a full process server might be driven by a change in the business model. For example, there may be a new business requirement for using and executing process flows. Because an ESB-only configuration does not include Business Process Choreographer functionality, you would need to re-configure the server to include that functionality.

When you run **zWPSConfig.sh**, it validates that there is an existing ESB-only configuration and as part of its actions, it adds to the ESB-only configuration the components (Business Process Choreographer for example) required to configure the server to a full Process server.

Command syntax

The following syntax diagram illustrates the syntax of the configuration script for re-configuring an ESB-only server to a full Process server.

Syntax diagram for re-configuring an ESB-only server to a full Process server

```
►►──zWPSConfig.sh──-augment───────────────────────────────────────────────────►


►──┬─-response─────────────────────────────────────────────────┬──────────────►
   │                                                      (1)  │
   └─┬─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp───┬─┘
     ├─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp─┤
     ├─/usr/lpp/zWPS/V6R0/zos.config/DmgrDB2.rsp──────────────┤
     └─/usr/lpp/zWPS/V6R0/zos.config/ManagedDB2.rsp───────────┘


►──-Z(Name of the value in the property file to override)─────────────────────►


                                (2)
►──-trace──┬─trace arguments─┬────────────────────────────────────────────────►◄
           └─*=all=enabled───┘
```

Notes:

1 Type the absolute path of the response file that is associated with the configuration you are creating. This example shows the default path to the response files packaged with the product. When you run the command, the **-response** parameter will represent the absolute path to the response file that you have modified with the values specific to your environment. The information required to create the DB2 for z/OS database in the response files named standAloneProfileDB2.rsp DmgrDB2.rsp and ManagedDB2.rsp contain properties that can be set to support DB2 for z/OS version 7 or DB2 for z/OS version 8. The default values are set for DB2 for z/OS version 8.

2 By including **-trace** the command writes to the appropriate log file.

For a detailed description of the keywords and keyword parameters in the configuration script, see the table, *Configuring WebSphere Process Server for z/OS - Running the server as a Process Server* in the topic, About the configuration script

Planning to install and configure WebSphere Process Server for z/OS

A WebSphere Process Server for z/OS installation and configuration includes planning activities for WebSphere Process Server for z/OS as well as WebSphere Application Server for z/OS.

The information in the sections below list the tasks that you need to perform and provide references to the documentation resources that can help you install and configure WebSphere Process Server for z/OS.

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

Planning the system

Table 8. Planning the system

Check off	Task description	Information center resource
	Determine the skills that you need.	See *Determining your skill needs* for information.
	Determine the system requirements.	See the system requirements web site for WebSphere Process Server and select *Version 6.0.1 Process Server for z/OS*. WebSphere Process Server for z/OS installs on top of WebSphere Application Server for z/OS . For a list of system requirements see, the *Hardware and software requirements* as documented in the WebSphere Application Server for z/OS Information Center.
	Understand security options and prepare for securing your system.	Security options are set in WebSphere Application Server for z/OS. See Security planning overview information in the WebSphere Application Server for z/OS Information Center.
	Implement Workload Management in goal mode on each z/OS system if necessary.	See *Workload management (WLM) plan strategy* in the WebSphere Application Server for z/OS Information Center.
	Implement Resource Recovery Services (if not already implemented) on each z/OS system.	See *Preparing Resource Recovery Services* in the WebSphere Application Server for z/OS Information Center.
	Plan for your performance and monitoring systems.	See *Monitoring end user response time* in the WebSphere Application Server for z/OS Information Center.
	Plan and define your problem diagnosis procedures.	See *Problem diagnostic plan strategy* in the WebSphere Application Server for z/OS Information Center.
	Consider automatic restart management before you install WebSphere Application Server for z/OS.	See *Automatic restart management* in the WebSphere Application Server for z/OS Information Center.

Table 8. Planning the system (continued)

Check off	Task description	Information center resource
	Perform planning tasks in preparation for loading the program materials from the installation media onto z/OS.	For information on planning tasks associated with unloading the WebSphere Application Server for z/OS install media, see *Planning for installation* in the WebSphere Application Server for z/OS Information Center. For information on planning tasks associated with unloading the WebSphere Process Server for z/OS install media, see *Planning to unload the product code*.

Install WebSphere Application Server for z/OS

Table 9. Load the code from installation media

Check off	Task description	Information center resource
	Load the contents of the WebSphere Application Server for z/OS install media onto the system.	For information on how to do this, see *Installing the product and additional software* in the WebSphere Application Server for z/OS information center.
	Prepare the base operating system.	For information on how to do this, see *Preparing the base operating system* in the WebSphere Application Server for z/OS information center.

Customize WebSphere Application Server for z/OS

Table 10. Customize WebSphere Application Server for z/OS

Check off	Task description	Information center resource
	Plan the WebSphere Application Server for z/OS runtime environment.	For information on how to do this, see *Planning for product configuration* in the WebSphere Application Server for z/OS information center.

Table 10. Customize WebSphere Application Server for z/OS (continued)

Check off	Task description	Information center resource
	Customize WebSphere Application Server for z/OS **Note:** You will customize the WebSphere Application Server for z/OS runtime environment to support the WebSphere Process Server for z/OS configuration that you will be creating. **Important:** Make sure that when you create the empty managed node in WebSphere Application Server for z/OS, that **you do not run the job BBOWMNAN**. This job will federate your node into the specified Deployment Manager cell. You do not want to federate the node at this point, as you will federate the Deployment Manager cell as part of the WebSphere Process Server for z/OS configuration process.	For information on how to do this, see *Configuring the product after installation* in the WebSphere Application Server for z/OS information center. If you are creating a stand-alone configuration server for WebSphere Process Server for z/OS, you must first create a stand-alone application server configuration as described in the WebSphere Application Server for z/OS information center. Creating a WebSphere Application Server for z/OS stand-alone server configuration provides you with the default profile that you will augment with WebSphere Process Server for z/OS configuration data. If you are creating a deployment manager configuration for WebSphere Process Server for z/OS, you must first create a deployment manager server configuration as described in the WebSphere Application Server for z/OS information center. Creating a WebSphere Application Server for z/OS deployment manager provides you with the default profile that you will augment with WebSphere Process Server for z/OS configuration data.

Install WebSphere Process Server for z/OS

Table 11. Install WebSphere Process Server for z/OS

Check off	Task description	Information center resource
	Print the Program Directory for WebSphere Process Server for z/OS.	The Program Directory is packaged with the product install media and is also available as a PDF from the WebSphere Process Server for z/OS library page.
	Load the WebSphere Process Server for z/OS product code from the installation media onto the system.	See *Load the product code from the installation media onto the system*.

Table 11. Install WebSphere Process Server for z/OS (continued)

Check off	Task description	Information center resource
	Run the installation script	Before you run the installation script (**zSMPInstall.sh**) you should understand the various keyword and keyword values supported by the command line installation. To obtain an understanding of the keyword and keyword parameter's that drive the installation, see "About the installation script" on page 25. For a description of how to run the installation script, see *Running the installation script*.

Configure WebSphere Process Server for z/OS

After running the installation script, you must run another script from the command line to create the product configuration for WebSphere Process Server for z/OS. Depending on how you plan to use WebSphere Process Server for z/OS, you can run one of two product configuration scripts as follows:

- zWESBConfig.sh

  Run this configuration script if the server you are configuring will run Mediation flows only.
- zWPSConfig.sh

  Run this configuration script if the server you are configuring will run Process flows.

Table 12. Product configuration tasks for WebSphere Process Server

Check off	Task description	Information center resource
	Understand the product configuration script	See "About product configuration on z/OS" on page 42. To understand the various keyword and keyword values supported by the command line configuration script, see "About the configuration script" on page 44.
	Understand database considerations and perform the necessary setup procedures prior to running the WebSphere Process Server for z/OS configuration script.	As part of your WebSphere Process Server for z/OS configuration planning you should make sure that DB2 for z/OS is setup to support the features that utilize it. See *Considerations for creating the database*.
	Create the databases and storage groups	Create the databases and the associated storage groups for those WebSphere Process Server for z/OS features that will utilize databases See *Creating databases and storage groups*.

Table 12. Product configuration tasks for WebSphere Process Server (continued)

Check off	Task description	Information center resource
	Understand how to use the response files.	The WebSphere Process Server for z/OS configuration process utilizes response files to determine the configuration parameters relevant to the configuration that you are creating. Response file templates, with set default values are packaged with the product. You must edit some of the default values in the response file template that you choose to use. See *Working with response files*.
	Create the product configuration by running the configuration script.	See "About running the configuration script" on page 92.

Depending on environment configuration variables and how you configured your response file, you may need to perform additional configuration tasks to complete the WebSphere Process Server for z/OS configuration.

Table 13. Post-configuration tasks for WebSphere Process Server

Check off	Task description	Information center resource
	(Conditional) - If the product configuration script did not create the database definitions automatically, run the SQL to create the database definitions.	See *Configuring the database manually*.

Hardware and software requirements

This topic includes a link to additional information on the hardware requirements and software corequisites and prerequisites needed for installing WebSphere Process Server.

For the latest information on platform-specific disk space requirements, supported operating systems, and the operating system fixes and patches that you must install to have a compliant operating system, see WebSphere Process Server detailed system requirements at http://www.ibm.com/support/docview.wss?uid=swg27006205 and select the link to your version of WebSphere Process Server.

Refer to the documentation for non-IBM prerequisite and corequisite products to learn how to migrate to their supported versions.

Preparing the base operating system

Prepare the base z/OS environment and all of the subsystems associated with z/OS to be able to support your WebSphere Process Server for z/OS installation.

1. Ensure you have an adequate skill level to support the base z/OS environment.
2. Complete the steps in the WebSphere Application Server for z/OS information center that describe how to install the product and additional software.

For information see *Installing the product and additional software*

3. Identify the z/OS systems on which you plan to run WebSphere Application Server for z/OS.

This topic provides information on prerequisite products and z/OS subsystem prerequisites with which you must comply before you plan for and customize WebSphere Application Server for z/OS.

You prepare the base operating system after you have installed the WebSphere Application Server for z/OS product code.

By performing these tasks you prepare your z/OS target systems to run WebSphere Application Server for z/OS.

By preparing the base operating system to run WebSphere Application Server for z/OS, you are by default preparing it to run WebSphere Process Server for z/OS.

For a description of how to prepare the base operating system, see *Preparing the base operating system* in the WebSphere Application Server for z/OS information center.

Once you complete the steps to prepare the base operating system, you need to complete the WebSphere Application Server for z/OS product configuration before moving onto installing the WebSphere Process Server for z/OS product code.

Preparing to install on z/OS

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

Prepare z/OS subsystems and do the other tasks in this section before you start installation and configuration activities.

This task assumes that you have installed and customized WebSphere Application Server for z/OS successfully and that you have prepared base operating system.

Print and review Planning to install and configure WebSphere Process Server for z/OS.

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 Process Server for z/OS. See “Determining your skill needs” on page 14 for more information.

Perform the tasks listed in this topic before you unload the contents of the WebSphere Process Server for z/OS media onto your system and before you run the installation and configuration scripts.

Proper preparation can help you avoid problems later on in the installation and configuration process.

1. Perform the necessary planning for an SMP/E installation of WebSphere Process Server for z/OS. See “Planning to load the product code onto the system” on page 60.
2. Know the name of the read-only installation HFS (the smpe_root) that is created as a result of loading the contents of the WebSphere Process Server for z/OS installation media onto the system.

The read-only installation HFS is often referred to as the *smpe_root* and the default value would look something like this: /usr/lpp/zWPS/V6R0 .

It is in the /zos.config/bin subdirectory of this read-only installation HFS that you should run the installation script (zSMPInstall.sh).

The smpe_root for WebSphere Process Server for z/OS is also the value of the **-smproot** keyword that you must specify when running the installation script (zSMPInstall.sh). For example, -smproot /usr/lpp/zWPS/V6R0.

3. Know the combined WebSphere Application Server for z/OS configuration HFS and the app_server_root root directory name.

   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 HFS directories. These configuration HFS directories are created through the WebSphere Application Server for z/OS configuration process and contain symbolic links to files in the WebSphere Application Server for z/OS product directory.

   When you run the WebSphere Process Server for z/OS installation script, the product configuration scripts (zWPSConfig.sh and zWESBConfig.sh) are copied into the /bin subdirectory of the WebSphere Application Server for z/OS configuration HFS directory structure. It is from this directory that you run the configuration scripts.

   The WebSphere Application Server for z/OS configuration HFS and app_server_root name combined is also the value of the **-runtime** keyword that must be specified when running the installation script (zSMPInstall.sh). As this app_server_root forWebSphere Application Server for z/OS will vary, so will the value you specify for the **-runtime** keyword. For example, the configuration root default value for a stand-alone application server node would be `/WebSphere/V6R0M0/`**`AppServer`**. While the configuration root default value for a Network Deployment cel would be `/WebSphere/V6R0M0/`**`DeploymentManager`**.

   For information on the WebSphere Application Server for z/OS file system, see Product file system in the WebSphere Application Server for z/OS information center.

4. Make sure you have authority to run the installation and configuration scripts.
5. Understand the use of the task keywords on both the installation script (zSMPInstall.sh) and the configuration scripts (zWPSConfig.sh and zWESBConfig.sh). For a description of the keywords and keyword parameters, see "About the installation script" on page 25 and "About the configuration script" on page 44.
6. Understand the database specifications for the WebSphere Process Server for z/OS components and features that use databases. For more information, see Database specifications.
7. Make sure you have created the necessary databases and the database storage groups, as some of the WebSphere Process Server for z/OS features use databases. You must create the databases and the database storage groups before you run the product configuration script. See Create the databases and storage groups for information on how to create the required databases and database storage groups.
8. Understand how the values in the response file apply to your environment configuration. You will need to edit values in the response files to suit your environment. In the product configuration command, you include the path name of the modified response file. See Working with response files for information on the contents of the response files.

When you have successfully finished these preparations you are ready to begin the process of installing and configuring WebSphere Process Server for z/OS.

Using a heterogeneous cell to support mixed platforms within a cell

With careful planning, you can create a deployment manager cell that includes nodes on both distributed and z/OS operating system platforms.

For example, you can create a deployment manager cell that includes z/OS nodes, Linux nodes, UNIX$^{(R)}$ nodes, and Windows nodes. 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.

If you use the administrative console to create a new server, you choose the *server template* that provides the initial configuration settings for the server. After you select a managed node on which to create a server, the administrative console offers you the choice of templates that can be used for the operating system platform of that node.

Product version information

The properties/version directory in the *smpe_root* contains important data about the product and its installed components, such as the build version and build date.

Product information files

This information is included in `WBI.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 `config_root/properties/version/history` directory.

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

This article describes the XML data files that store product version information for WebSphere Process Server for z/OS Version 6.0.x. By default, the document type declarations (DTDs) for these files are in the `properties/version/dtd` folder of the*smpe_root*, or the server root directory. See the **Directory locations** section of this topic for more information.

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

platform.websphere

The existence of this file 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 "webspehereId" "websphere.dtd"
<websphere name="IBM WebSphere Application Server" version="6.0"/>
```

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

The existence of this file 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 websphere PUBLIC "productId" "product.dtd"
<product name name="IBM WebSphere Application Server - ND">
<id>ND</id>
<version>6.0.0version>6.0.0>
<build-info date="02/03/05" level="s0461.18"/>
</product
```

WBI.product

The existence of this file indicates the particular WebSphere Process Server for z/OS 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>WBI</id> product indicates that the installed product is WebSphere Process Server for z/OS on a Network Deployment configuration. An example of the file follows:

```
<?xml version="1.0" encoding="UTF-8">
<!DOCTYPE websphere PUBLIC "productId" "product.dtd">
<product name name="IBM WebSphere Process Server">
<id>WBI</id>
<version>6.0.0version>6.0.0>
<build-info date="02/03/05" level="s0461.18"/>
</product
```

Reports

After completing the installation, the WebSphere Application Server for z/OS reports will reflect the installed WebSphere Process Server for z/OS product. Refer to the product version information topic in the WebSphere Application Server for z/OS information center for details.

Planning to load the product code onto the system

Use this task to prepare to install WebSphere Process Server for z/OS program materials from the installation media onto the z/OS system.

1. Review the pertinent information in *Planning to install and configure WebSphere Process Server for z/OS*
2. Print a copy of the Program Directory for WebSphere Process Server for z/OS.

   The Program Directory provides specific *program information* that is needed to install, maintain, service, and use the product.
3. Review *Installation: Resources for learning* for reference information you might need during installation.

Load the product code from the installation media onto the system before running the WebSphere Process Server for z/OS installation script.

Obtain WebSphere Process Server for z/OS 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 listed in this section before you begin the SMP/E installation of WebSphere Process Server for z/OS. 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 Process Server for z/OS meets the hardware and software requirements. . See *Hardware and software requirements* for more information.
2. Identify the software delivery option you will use. To review your product delivery options, see the z/OS and OS/390 delivery options. See *Using and IBM Custom-Build Product Delivery Option* or *Using an IBM SystemPac or ServerPac* for more information.
3. Learn about WebSphere Process Server for z/OS product data sets. See *Planning for product data sets* for more information.
4. Learn about WebSphere Process Server for z/OS product directories, and plan a mount point convention. See *Planning for product HFS directories* for more information.

When you have finished the planning process, you are ready to install theWebSphere Process Server for z/OS program materials from the installation media onto the system. See . *Loading the product code from the installation media*.

Planning for product HFS directories

WebSphere Process Server for z/OS product code resides in MVS partitioned data sets (the product data sets) and MVS hierarchical file system (HFS) directories (the product directory and its subdirectories).

This article describes the WebSphere Process Server for z/OS product directory.

Product directory

All WebSphere Process Server for z/OS product HFS files reside in the installation HFS directory and its subdirectories. Throughout the product and documentation, WebSphere Process Server for z/OS `smpe_root` is used to represent the fully-qualified path name of the WebSphere Process Server for z/OS product directory.

The default value for the WebSphere Process Server for z/OS smpe_root directory is `/usr/lpp/zWPS/V6R0`.

The product directory and all of its subdirectories should reside 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 used just for WebSphere Process Server for z/OS. The installation jobs and program directory assume that such a separate data set is allocated; and we refer to it as wps_hlq.SBPZHFS, where *wps_hlq* represents the product data set name high-level qualifiers.

Product directory and configuration directory

Each WebSphere Process 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 contain symbolic links to files in the product directory.

For information on directory naming conventions for both WebSphere Process Server for z/OS and WebSphere Process Server for z/OS, see Directory conventions.

Planning for product data sets

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

WebSphere Process 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 BPZ.

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

Product data set contents

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

wps_hlq.**SBPZEXEC**
CLIST scripts

wps_hlq.**SBPZJCL**
JCL for installation jobs

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

wps_hlq.**ABPZANT**
HFS files

wps_hlq.**ABPZEBCD**
HFS files (EBCDIC)

wps_hlq.**ABPZEXEC**
Customization dialog

wps_hlq.**ABPZJCL**
JCL for installation jobs

See WebSphere Process Server for z/OS Program Directory (GI10-0781) 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 Process Server for z/OS.

Product data set naming convention

As noted above, certain WebSphere Process 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.

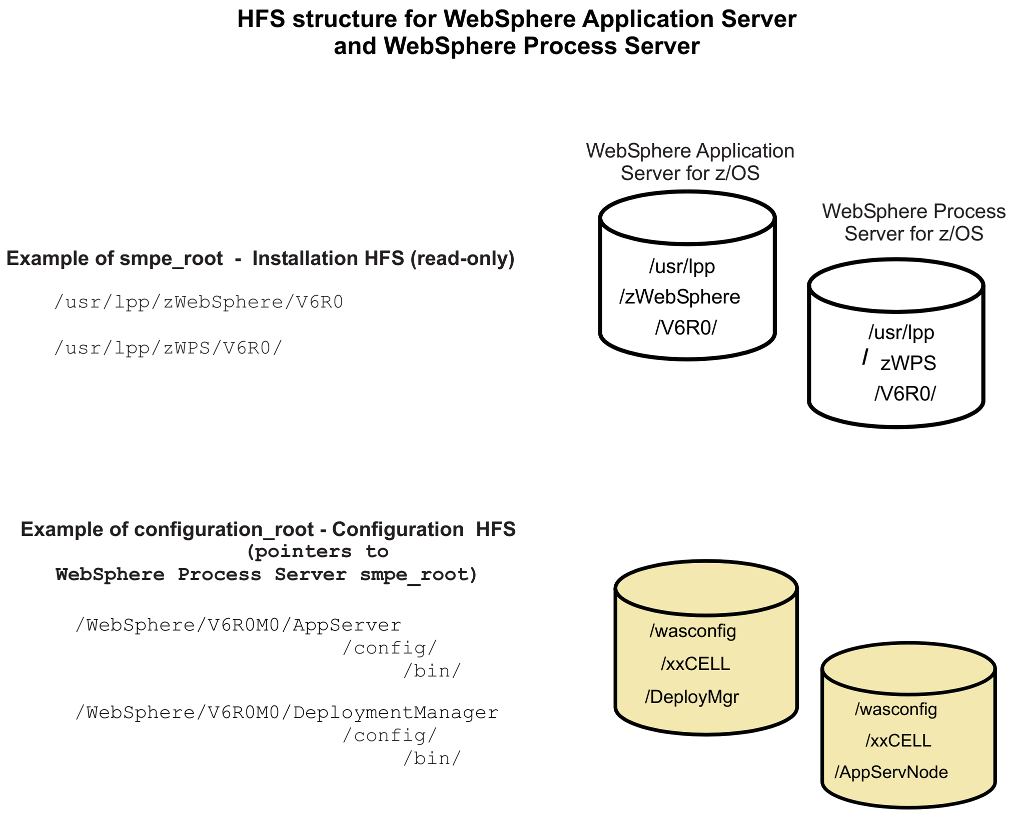
On the other hand, in order to continue to run WebSphere Process Server for z/OS while applying maintenance, you must have at least two copies of the product data sets: one for the running application execution environment and one to which service is applied.

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

Understanding the product file system

The files in the file system resulting from the WebSphere Process Server for z/OS SMP/E installation are organized into a hierarchical structure.


HFS structure for WebSphere Application Server
and WebSphere Process Server
Example of smpe_root - Installation HFS (read-only)
/usr/lpp/zWebSphere/V6R0
/usr/lpp/zWPS/V6R0/
WebSphere Application Server for z/OS
/usr/lpp /zWebSphere /V6R0/
WebSphere Process Server for z/OS
/usr/lpp / zWPS /V6R0/
Example of configuration_root - Configuration HFS (pointers to WebSphere Process Server smpe_root)
/WebSphere/V6R0M0/AppServer
/config/
/bin/
/WebSphere/V6R0M0/DeploymentManager
/config/
/bin/
/wasconfig /xxCELL /DeployMgr
/wasconfig /xxCELL /AppServNode

All of the files within a hierarchical file system (HFS) are members of a directory. Each directory is, in turn, a member of another directory at a higher level in the hierarchy.

The installation HFS (smpe_root) represents a file organization that stores data in a top-to-bottom organization structure. All access to the data starts at the top and proceeds downward throughout the levels of the hierarchy.

At the highest level or the starting point in the HFS is the *root directory*. The root directory is the directory that begins with a forward slash (/). All file system entries, including mounted partitions, hang off of the root directory.

Each directory contains information about itself as well as the files contained within, including the file names, their size, the date the files were created or modified, and other relevant information.

The entire HFS file structure, from the root directory on down, is stored as a single data set on an IBM mainframe, and z/OS manages access to specific directories and files. z/OS can manage the hierarchical files through its own HFS facility. As a result, HFS files can be used in both the UNIX System Services and z/OS environments. This makes it possible for application programs that are designed to run in a UNIX environment to handle files as they normally would under z/OS.

Although most HFS files store data, some can be executable modules and still others can consist of UNIX shell scripts. Executable modules, or programs, are similar to the compiled and linked programs found in z/OS load libraries. UNIX shell scripts are similar to procedures. They consist of a series of UNIX commands (and, optionally, scripting commands) that are executed in order whenever the file is invoked.

For information on directory naming conventions for both WebSphere Process Server for z/OS and WebSphere Process Server for z/OS, see Directory conventions.

Planning the configuration

Servers can be configured in different application serving environments to provide its functions, and the configuration procedure you use varies according to the application serving environments.

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 for more information.

Types of configurations

You can configure the server on server hardware as individual *stand-alone servers* or as a *managed group of servers.*

- **A stand-alone server** configuration has its own administrative console and all sample applications (if you selected to create the sample applications gallery through property settings in the stand-alone response file). Each stand-alone server is fully operational and is managed independently from all other servers.

  A stand-alone application server cell is the simplest WebSphere Process 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 configured as a process server that runs the administrative console application. You can deploy and run additional applications on this server.

  A WebSphere Process Server for z/OS stand-alone configuration supports a Cloudscape or a DB2 for z/OS database. If you are planning on using WebSphere Process Server for z/OS in a stand-alone configuration that uses a DB2 for z/OS database, you will need to perform additional configuration steps related to the use of DB2 for z/OS.

  For more complicated or robust WebSphere Process Server for z/OS application-serving environments, the Network Deployment cell configuration is recommended.

- **Network deployment** configuration, uses a deployment manager for centralized administration tasks, such as managing the configuration for all of the managed nodes in its cell and deploying applications to selected managed nodes in the cell.

  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.

  A WebSphere Process Server for z/OS network deployment configuration supports a DB2 for z/OS database. You cannot use Cloudscape for in a network deployment configuration. You will need to perform additional configuration steps related to the use of DB2 for z/OS.

- **Managed node** configuration, involves creating a new managed node in a Network Deployment cell in order to add process servers to the cell.

  This part of the configuration process creates a 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.

Choosing between a stand-alone or Network Deployment configuration

Configuring WebSphere Process Server for z/OS into a network deployment configuration is very different from configuring a WebSphere Process Server for z/OS stand-alone server.

The WebSphere Process Server for z/OS configuration process requires you to first install and configure WebSphere Application Server for z/OS. The WebSphere Application Server for z/OS installation and configuration provides you the application serving environment into which you will create your WebSphere Process Server for z/OS configuration.

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

Table 14. Differences between stand-alone configuration and a network deployment configuration

Type	Configuration process	Address spaces	Administration isolation	Operational isolation?	Multiple servants allowed?	Clustering available?
Stand-alone	Set up a stand-alone server node through the ISPF Customization Dialog. Set up additional servers within the node through the administrative console or scripting. Configure the stand-alone application server as a process server by running the configuration script with the standalone response file.	**Minimum:** four (location service daemon, controller, servant, control region adjunct) **Maximum:** Limited only by resources.	Each stand-alone server node is a separate administrative domain.	You can start and stop servers independently. Each server has an independent, unshared JNDI namespace.	Yes	No
Network deployment	Set up each deployment manager node through the ISPF Customization Dialog. Do not federate the node Configure as a process server deployment manager by running the configuration script with the deployment manager response file. Create managed node configuration by running the configuration script with the managed node response file.	**Minimum:** 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.	All nodes in the cell are in the same administrative domain.	You can start and stop servers independently. The JNDI namespace is shared among all servers in the cell.	Yes	Yes

Planning considerations for a stand-alone configuration

The most important planning consideration with regard to a WebSphere Process Server for z/OS stand-alone configuration is related to choosing the database that will support the stand-alone configuration.

If you configure WebSphere Process Server for z/OS into a stand-alone server using Cloudscape, all the parts of the configuration, including the databases, are within the UNIX file system (that is, within the stand-alone server's configuration HFS.) This makes it easy to configure the server because the shell scripts can do all the work without you having to make decisions about the configuration.

When you configure a stand-alone server that uses databases defined in DB2 for z/OS, there are more things for you to plan and more manual steps to configure the databases. However the configuration of the resources within WebSphere is fairly straight-forward because the WebSphere stand-alone configuration is simple.

Planning considerations for a network deployment configuration

In a network deployment configuration there are now more choices in the WebSphere configuration as well as choices in the database configuration in DB2. Therefore there are more steps to the configuration and more things for you to consider. A key difference is that in a network deployment configuration the Deployment Manager resides in one node and the Application Server in another. Both nodes must be 'augmented' to add support for WebSphere Process Server for z/OS and the augmentation of the Application Server node must be done before that node is federated into the cell.

Topology planning considerations

One of the first steps in planning any large software installation is determining the topology in which it will reside. Since WebSphere is middleware, the topologies can become quite complex and require some forethought.

The following list provides the considerations that need to be made while planning an appropriate WebSphere topology. Typically, you will address these issues when you plan your WebSphere Process Server for z/OS installation.

- Security
- Cost
- Administration (maintainability)
- Performance (throughput, response times)
- Availability
- Scalability
- Session state

For detailed information about these topics, refer to the WebSphere Application Server for z/OS documentation.

A stand-alone (single server) topology providing a framework for a quick start or development environment. Also, smaller corporations may find that a single server topology is all they need to meet their requirements.

A network deployment (multi-server) topology facilitates administration of application servers that may be dispersed among multiple sysplexes in a network environment.

See ″Choosing between a stand-alone or network deployment configuration″ topic in the WebSphere Process Server for z/OS documentation infocenter for specifics on the differences between a WebSphere Process Server for z/OS stand-alone cell and network deployment cell.

Stand-alone configuration

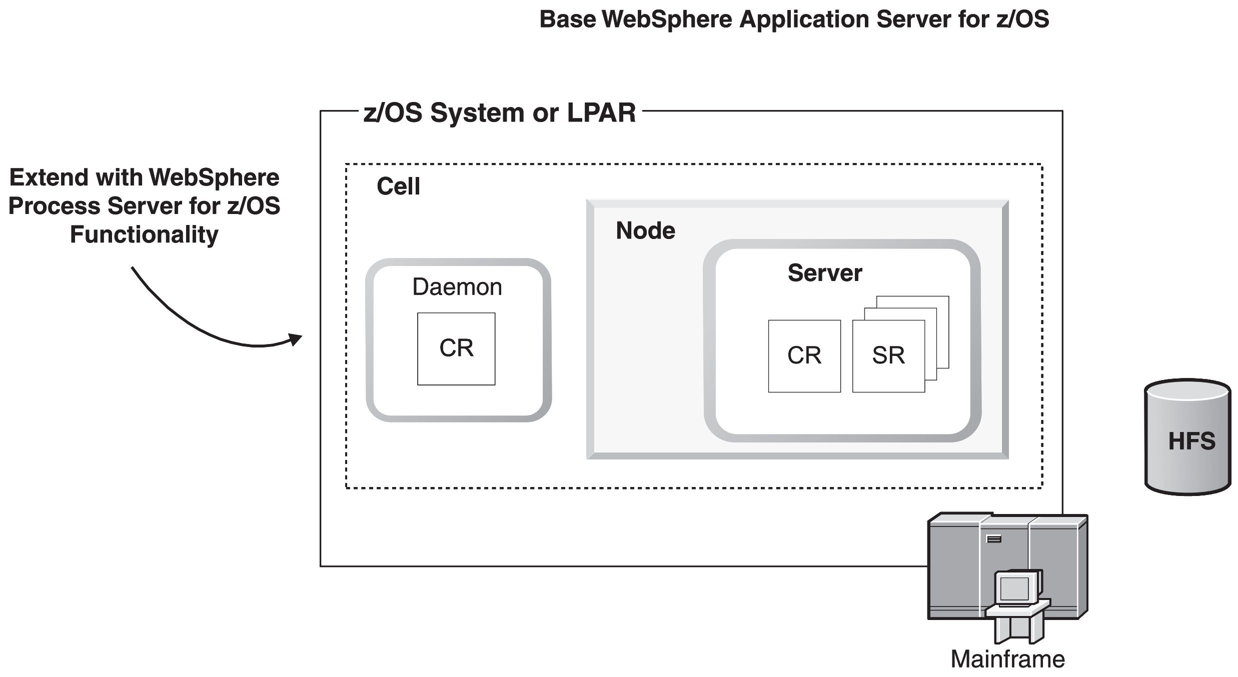
WebSphere Process Server for z/OS can be installed and configured to run in a WebSphere Application Server for z/OS *stand-alone* configuration. This configuration is also called a *base* WebSphere Application Server for z/OS configuration.

Stand-alone Application Server

The stand-alone application server is the easiest and simplest operating structure in WebSphere Application Server for z/OS. It consists of one or more server instances, a daemon server, one node and one cell running in a single z/OS system or LPAR.

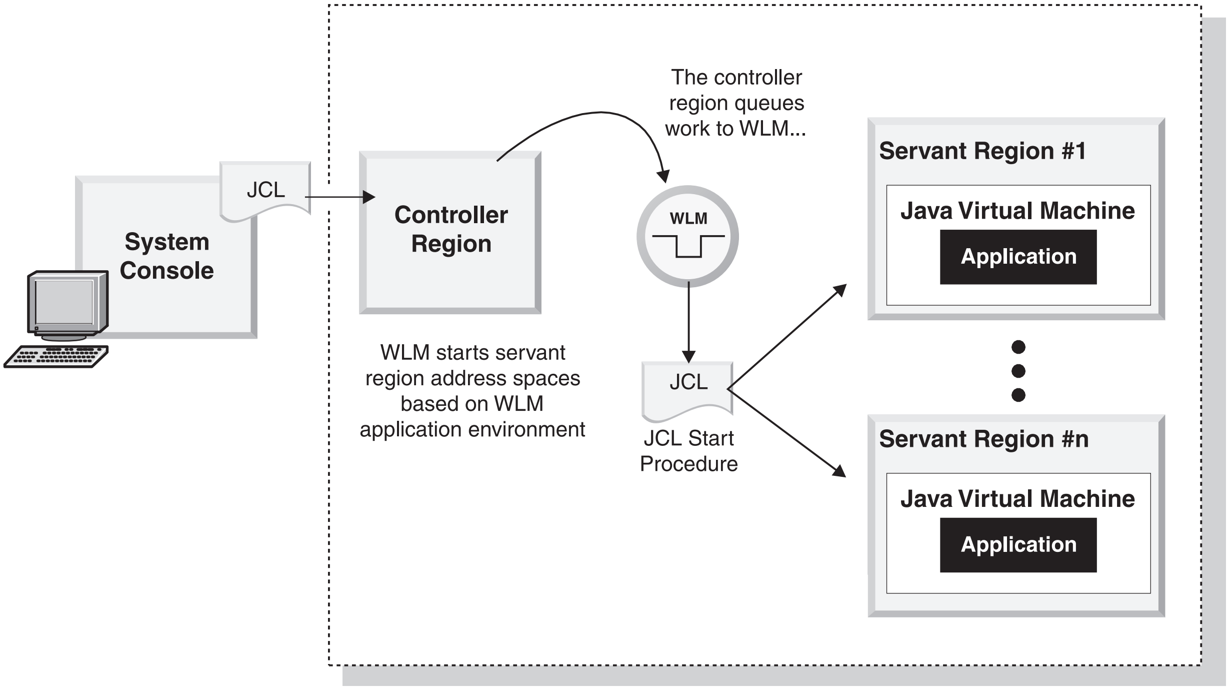
- A *cell* represents the boundary of an administrative domain.
- A *node* is a collection of servers grouped together on a single system for the purposes of administration.
- A *daemon* manages requests and owns the interaction with workload management.

The definitions and configuration files are all kept in the HFS directory structure for the stand-alone application server. The application server relies on a set of XML/XMI files for its configuration repository. The application server administration is based on two tools: the Web browser-based graphical user interface called the administrative console, and the non-graphical command line scripting client.


Base WebSphere Application Server for z/OS
z/OS System or LPAR
Extend with WebSphere Process Server for z/OS Functionality
Cell
Node
Daemon
CR
Server
CR
SR
HFS
Mainframe

Application Server

The server instance consists of a *controller region* and a *servant region*. In WebSphere Application Server, the Java Virtual Machines (JVM) reside in the controller and servant address spaces. A controller region and one or more servant regions are called a *server*. The controller and servant structure design makes it possible to start multiple servants by the workload manager (WLM), based on the workload queued by the controller region.



Daemon Server

The *daemon* server is a special server with only one controller region. It is a supporting daemon that is responsible for accepting and initiating application requests. To perform its work, the daemon must know which servers are active and know all the applications in them. There is only one daemon per cell per system.

Node

A *node* is a collection of servers on a given system or LPAR. The node name in a cell has to be unique. The purpose of a node becomes clearer when the configuration includes several systems or LPARs, and the whole environment is managed from a central administrative console. This would assume that we also have a *Node Agent*, which is a specialized server that receives commands from the central administrator and issues those commands against the servers in its node. The stand-alone application server does not have such a Node Agent, because the administrative Web interface is running within one of its own servers.

A stand-alone application server installation is a single node that runs one application server process independent of a Network Deployment cell.

Cell

A *cell* is a collection of nodes that makes up an administrative domain or boundary. A cell name must be unique and cannot be extended beyond the sysplex. The simplest example of this is the stand-alone application node. The administrative domain extends only to itself, which makes it very small. The cell can be extended to exist across multiple systems or LPARs in a sysplex and consist of many nodes, which makes the administrative domain very large. Even a stand-alone Application Server node may contain multiple servers, which also expands the administrative domain. Whether it is a very simple or a very large and more complicated configuration, an administrative domain is always required.

Benefits of a standalone configuration

For many reasons, complete isolation between test and production systems is always recommended. The nature of testing implies brand new changes are injected into an application the company depends on. If the changes cause a major failure during test and there is no isolation, the production system could be slowed by bad application code in the test environment.

There are three typical reasons to have WebSphere focused on one LPAR.

- The company is very large and gives each of their test groups an LPAR to exercise the application.
- The company is smaller and needs to run production and test on one zSeries system. Maybe they give one LPAR to test and all others to production.
- The company is trying to decide whether to use WebSphere and gives up a small bit of resources for feasibility testing.

Multiple stand-alone application server nodes can be configured to run in a single LPAR. However, each stand-alone application server node is isolated from the others and contains its own server root (HFS), cell, node, TCP ports and admin interface.

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.

A WebSphere Process Server for z/OS stand-alone configuration supports the following databases:

- Cloudscape

  A stand-alone WebSphere Process Server for z/OS configuration with a Cloudscape database can be created with minimal customization to the response file. For this configuration you should be able to achieve a working WebSphere Process Server for z/OS cell with the least amount of effort.

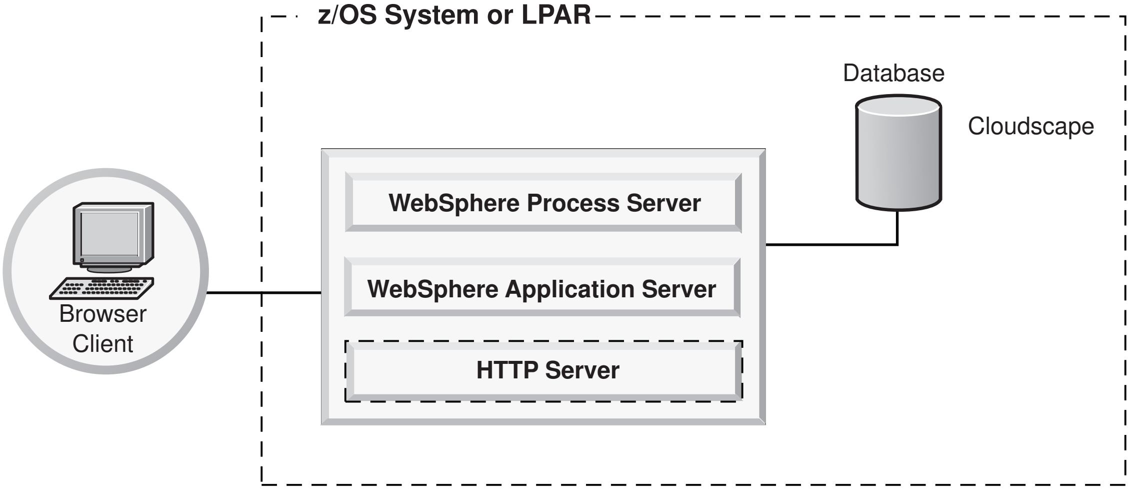
  Because this configuration provides you with a fully functional server for testing applications, with minimum effort, we recommend this as your first configuration type.
- DB2 for z/OS Version 7 and Version 8

  A stand-alone WebSphere Process Server for z/OS configuration with a DB2 for z/OS database requires additional procedures for setting up and configuring the database.

Task flow: Creating a stand-alone configuration:

This topic provides a task flow for installing WebSphere Process Server for z/OS and creating a stand-alone configuration.

A WebSphere Process Server for z/OS stand-alone configuration supports a Cloudscape or a DB2 for z/OS database. The following illustration is of a stand-alone configuration using Cloudscape.



Installing WebSphere Process Server for z/OS and creating a stand-alone configuration involves a series of steps that include WebSphere Application Server for z/OS installation and customization procedures.

The list of steps below represent a sequential task flow for installing WebSphere Process Server for z/OS and creating a stand-alone configuration. The details on how to perform these tasks are described in this information center and, where appropriate, in the WebSphere Application Server for z/OS information center.

The task flow described in this topic assumes that users have already loaded WebSphere Process Server for z/OS and WebSphere Application Server for z/OS onto the system using SMP/E.

Note: Use the program directory for WebSphere Application Server for z/OS and for WebSphere Process Server to guide you through the SMP/E process of unloading the modules from the media onto your z/OS system. The result of completing the SMP/E installation is an installation HFS (smpe_root).

1. Prepare and plan your installation.

   Make sure you have access to the required documentation for both WebSphere Process Server for z/OS and WebSphere Application Server for z/OS.

   Prepare your z/OS target systems to run WebSphere Process Server, as described in .
2. Determine which database you will use (Cloudscape or DB2 for z/OS).

   Tip: A first time user should to try to install and create a stand-alone configuration with Cloudscape before trying the other complex topologies that are available, such as the use of DB2.

   If you are creating a stand-alone configuration that will use DB2 for z/OS, you will need to perform the following DB2-related tasks:

   - Decide a naming convention that will apply to all the databases, storage groups, tables, indexes and views that you will create in DB2.

     The naming convention may be predetermined by your own site standards.
   - Decide whether to execute the DB2 Data Definition Language (DDL) statements using z/OS tools like SPUFI or Unix tools like shell and ANT scripts.
   - Decide on an approach to the security that will apply to the WebSphere Process Server for z/OS objects in DB2.

3. Create a stand-alone application server cell.

   Use the ISPF customization dialog to create the stand-alone application server cell, as described in the WebSphere Application Server for z/OS information center.

   As a result of completing the ISPF customization dialogs and running the generated jobs, you will have a default profile associated with the stand-alone configuration.

   Several values specified on ISPF customization will need to be replicated in the response file that is used by the WebSphere Process Server for z/OS configuration script. These values include:

   - Cell name
   - Node name
   - Server name
   - Template path
   - Profile path
4. Run the installation script against the WebSphere Process Server for z/OS smpe_root to create the symbolic links and product definitions.

   This step involves running the WebSphere Process Server for z/OS installation script (zSMPInstall.sh) from a command line.

   The installation command contains keyword parameters for the WebSphere Process Server for z/OS smpe_root, as well as the WebSphere Application Server for z/OS configuration HFS / app_server_root.

   When the installation script completes, it extends the WebSphere Process Server for z/OS features into the WebSphere Application Server for z/OS configuration HFS. For example, /WebSphere/V6R0/ AppServer.
5. Create the databases and storage groups for the WebSphere Process Server for z/OS components that use databases.

   Before you run the WebSphere Process Server for z/OS configuration script you need to create the databases and storage groups utilized by WebSphere Process Server for z/OS.

   Note: This step applies only if the database being used is DB2 for z/OS.

   You work with Data Definition Language (DDL) statements to define databases to DB2 z/OS.
6. Edit the properties in the stand-alone response file.

   WebSphere Process Server for z/OS includes sample response files for each of the supported configurations.

   There are two sample response files for the stand-alone configuration as follows:

   - standAloneProfile.rsp

     The standAloneProfile.rsp includes configuration parameters for the Cloudscape database.

     You should use this response file for your first configuration, because it provides you with a fully functional WebSphere Process Server for z/OS server for testing applications, with a minimum amount of effort.

     Note: A stand-alone configuration created with the standAloneProfile.rsp is not meant for use in a production environment.
   - standAloneProfileDB2.rsp

The standAloneProfileDB2.rsp includes configuration parameters for the DB2 for z/OS database.

The use of this response file requires some additional configuration requirements to set up the DB2 for z/OS databases.

This response file supports DB2 for z/OS version 7 and DB2 for z/OS version 8. The default properties assume a DB2 for z/OS version 8.

If you have DB2 for z/OS version 7 installed, you will need to modify the default values in this response file to support DB2 for z/OS version 7. The response file contains comments on those properties that apply to DB2 for z/OS version 7 specifically.

You need to edit some of the property values in the response file to tailor it to your environment.

This response file is used as a keyword parameter in the configuration command. It contains the WebSphere Process Server for z/OS profile information that will be used to augment the default profile that was created when you filled out the customization dialogs and ran the resulting jobs to create the stand-alone application server cell.

7. Configure WebSphere Process Server for z/OS by running the configuration script.

   As a result of running the configuration script, configuration actions augment the stand-alone default profile with WebSphere Process Server for z/OS configuration data from the stand-alone response file. Additionally, the DDL and SQL for the database tables is generated.

   Note: For the standAloneProfileDB2.rsp response file you can set a property to invoke actions that run the generated DDL and SQL, thereby creating the database definitions automatically. Because the database administrator may not allow automatic updates to the database, you have the option to delay running the DDL and SQL scripts that create the database definitions. You will then need to run these scripts manually, after the configuration process.

8. Create the database definitions manually by running the DDL and SQL that was generated by the WebSphere Process Server for z/OS configuration script.

   (DB2 for z/OS only) If you did not configure the response file to create the database definitions automatically, you need to run the generated DDL and SQL manually to create the database definitions.

9. Tune for performance. For best performance on any platform, see Tuning the application serving environment.

Network deployment configuration

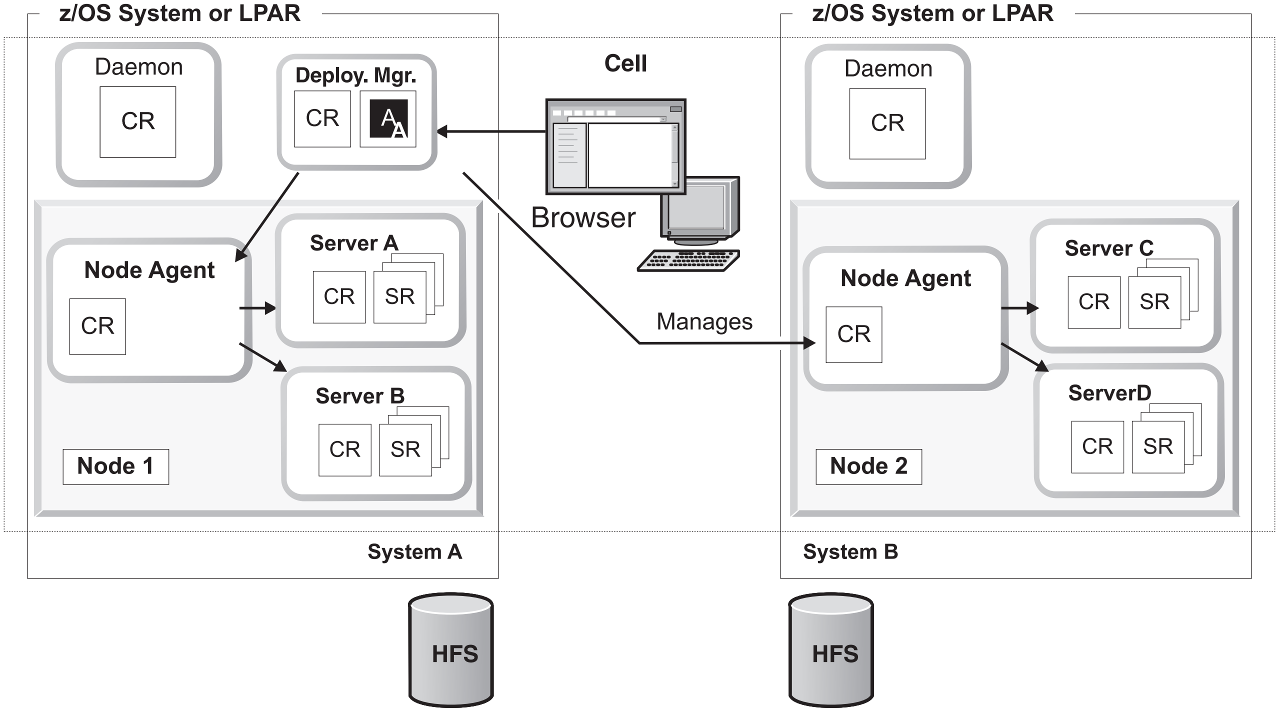
WebSphere Process Server for z/OS can be installed and configured to run in a WebSphere Application Server *network deployment* configuration. This configuration is also referred to as a WebSphere Application Server for z/OS *deployment manager* configuration.

Network deployment configuration

A network deployment configuration facilitates administration of application servers that may be dispersed among multiple sysplexes in a network environment. This feature allows you to share applications and servers.

- A *deployment manager* is a special kind of application server that provides administrative services in a network.

- A *node agent* is a special single controller region server that the deployment manager communicates with to manage the application servers in the node.



Sysplex

A *sysplex* is a set of z/OS systems communicating and cooperating with each other, through certain multisystem hardware components and software services, in order to process customer workloads. Notice that in the example, we have a single cell with multiple nodes that spans systems in a sysplex. The cell is under the management of the deployment manager, and only one deployment manager is permitted per cell, even when the cell spans z/OS systems.

Deployment Manager

The *deployment manager* node is a central point of administrative control. An administrative application (see A in diagram) runs in the deployment manager. The deployment manager does not interact with the servers directly, it communicates with node agents, which in turn control the servers in the node. There is only one deployment manager per cell, no matter how many systems the cell spans in the sysplex.

Base application nodes can be federated (joined) into a deployment manager cell using the addNode shell script or through the administrative console.

Node Agent

The *node agent* is responsible for administration at the node level. Before the deployment manager can manage application servers, those applications servers need to be grouped into nodes. Nodes must stay with a system or LPAR. There is one node agent per node.

The node agent is also a special server in that it has no servant regions. The node agent communicates all configuration changes to the deployment manager at scheduled intervals. Changes can also be forced up to the deployment manager for faster change notification at any time.

Benefits of a network deployment configuration

In a network deployment configuration, the applications and servers are now truly shared. In this configuration, anyone who is logged on to the administrative console can start and stop servers and applications. In a stand-alone configuration, starting and stopping is only allowed from the administrative application contained in the individual server. Taking administration out of the individual server allows for expanded administrative functionality.

The main advantage to use managed nodes in a cell rather than using the same number of stand-alone servers is the centralized administration that the deployment manager provides for the cell.

If you want to balance workload, such as service requests, over a set of servers, you can create a server cluster, then add servers as members of that cluster. You can also create a backup cluster, to provide failover support for the server cluster to which it is assigned.

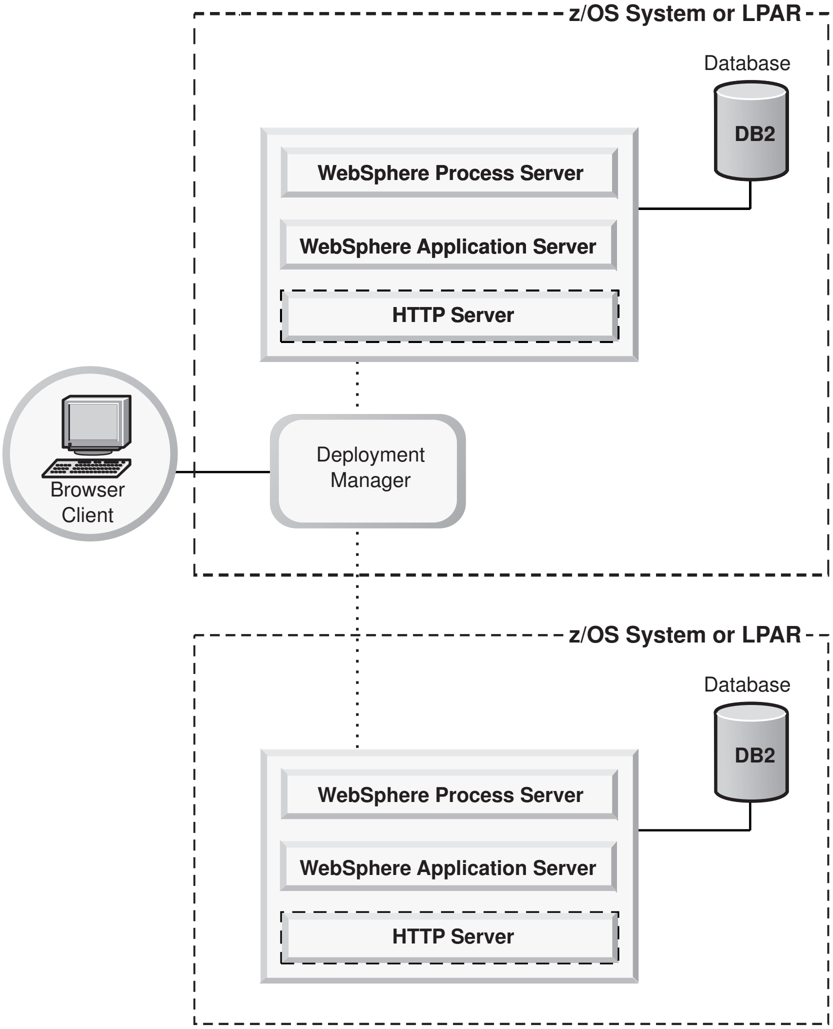
This part of the configuration process creates the initial cell configuration, the deployment manager, and a location service daemon for the z/OS system on which the deployment manager runs. Once the Network Deployment cell is created, add application server nodes by creating and federating new managed nodes, or by federating stand-alone application server nodes into the Network Deployment cell.

A WebSphere Process Server for z/OS Network deployment configuration supports a DB2 for z/OS database only.

Network Deployment (ND) cell is more complex to configure in terms of WebSphere Application Server for z/OS profiles, resources, containers, applications and other infrastructure, but similar to the stand-alone server with DB2 in terms of the database setup requirements.

Task flow: Creating a network deployment configuration:

This topic provides a task flow for installing WebSphere Process Server for z/OS and creating a network deployment configuration.



Installing WebSphere Process Server for z/OS and creating a network deployment configuration involves a series of steps that include WebSphere Application Server for z/OS installation and customization procedures.

The list of steps below represent a sequential task flow for installing WebSphere Process Server for z/OS and creating a network deployment configuration. The details on how to perform these tasks are described in this information center and, where appropriate, in the WebSphere Application Server for z/OS information center.

The task flow described in this topic assumes that users have already loaded WebSphere Process Server for z/OS and WebSphere Application Server for z/OS onto the system using SMP/E.

Note: Use the program directory for WebSphere Application Server for z/OS and for WebSphere Process Server to guide you through the SMP/E process of unloading the modules from the media onto your z/OS system. The result of completing the SMP/E installation is an installation HFS.

Create the network deployment cell

1. Prepare and plan your installation and configuration.

Make sure you have access to the required documentation for both WebSphere Process Server for z/OS and WebSphere Application Server for z/OS.

Prepare your z/OS target systems to run WebSphere Process Server, as described in .

2. Perform the SMP/E installation of WebSphere Application Server for z/OS and WebSphere Process Server for z/OS to create the smpe_root directory for each product.
3. Create a network deployment cell.

   Use the ISPF customization dialog to create the network deployment cell, as described in the WebSphere Application Server for z/OS information center.

   Several values specified on ISPF customization will need to be replicated in the deployment manager response file that is used by the WebSphere Process Server for z/OS configuration script. These values include:

   - Cell name
   - Node name
   - Server name
   - Template path
   - Profile path

   As a result of completing the ISPF customization dialogs and running the generated jobs, you will have a default profile associated with the network deployment configuration.
4. Run the WebSphere Process Server for z/OS installation script against the WebSphere Process Server for z/OS smpe_root to create the symbolic links and product definitions.

   This step involves running the WebSphere Process Server for z/OS installation script (zSMPInstall.sh) from a command line.

   The installation command contains keyword parameters for the WebSphere Process Server for z/OS The smpe_root, as well as the WebSphere Application Server for z/OS configuration HFS / app_server_root, and when it completes, it extends the WebSphere Process Server for z/OS features into the WebSphere Process Server for z/OS configuration HFS directory. For example: /WebSphere/V6R0/DeploymentManager
5. Complete the DB2 for z/OS prerequisite actions

   The network deployment configuration supports DB2 for z/OS database only.
6. Create the databases and storage groups for the WebSphere Process Server for z/OS components that use databases.

   Before you run the WebSphere Process Server for z/OS configuration script you need to create the databases and storage groups utilized by WebSphere Process Server for z/OS.
7. Stop the deployment manager server if need be.
8. Edit the properties in the deployment manager response file (DmgrDB2.rsp).

   WebSphere Process Server for z/OS includes sample response files for a network deployment configuration. You will need to edit some of the property values in the response file to tailor it to your environment.

   This response file is used as a keyword parameter in the configuration command. It contains the WebSphere Process Server for z/OS profile information that will be used to augment the default profile that was created when you filled out the customization dialogs and ran the resulting jobs to create the network deployment cell.
9. Create a DB2JccConfiguration.properties file.

10. Change directory to the deployment manager bin directory.
11. If you have not already done so, switch to the administrator's user id.
12. Run the WebSphere Process Server for z/OS configuration script.

    As a result of running the configuration script, configuration actions augment the network deployment default profile with WebSphere Process Server for z/OS configuration data from the deployment manager response file. Additionally, the DDL and SQL for the database tables is generated.

    Note: You can configure the response file to invoke actions that run the generated DDL and SQL, thereby creating the database definitions automatically. This can be done only for the WebSphere Process Server for z/OS database.
13. Create the database definitions manually by running the DDL and SQL that was generated by the WebSphere Process Server for z/OS configuration script.

    If you did not configure the response file to create the database definitions automatically, you need to run the generated DDL and SQL manually to create the database definitions.
14. Tune for performance. For best performance on any platform, see Tuning the application serving environment.

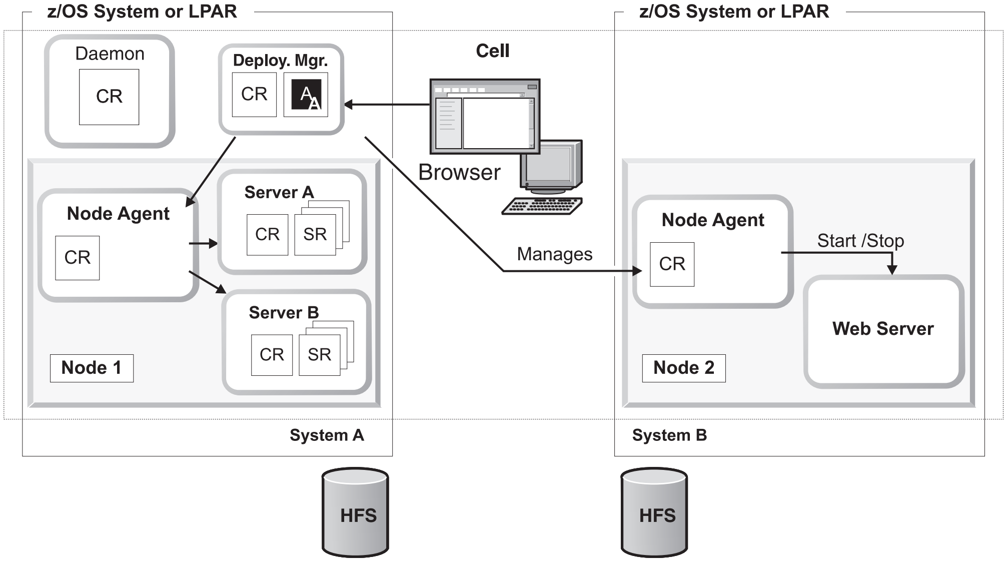
Managed node configuration

WebSphere Process Server for z/OS can be installed and configured to run in a WebSphere Application Server for z/OS *managed node* configuration.

Managed node

A *managed node* is a Web server that is being managed by the deployment manager. A managed node configuration provides the ability to start and stop the Web server and to push the plug-in configuration file to the Web server automatically. The Web server managed node requires an application server node to be created on the Web server system. It is commonly used when Web servers are installed behind a firewall where an application server node can be installed.

You can create an empty managed node and add it to an existing network deployment cell. The managed node will contain a node agent but no application servers. You can add an application server to the node with the administrative console since it can be federated it into an existing network deployment cell.

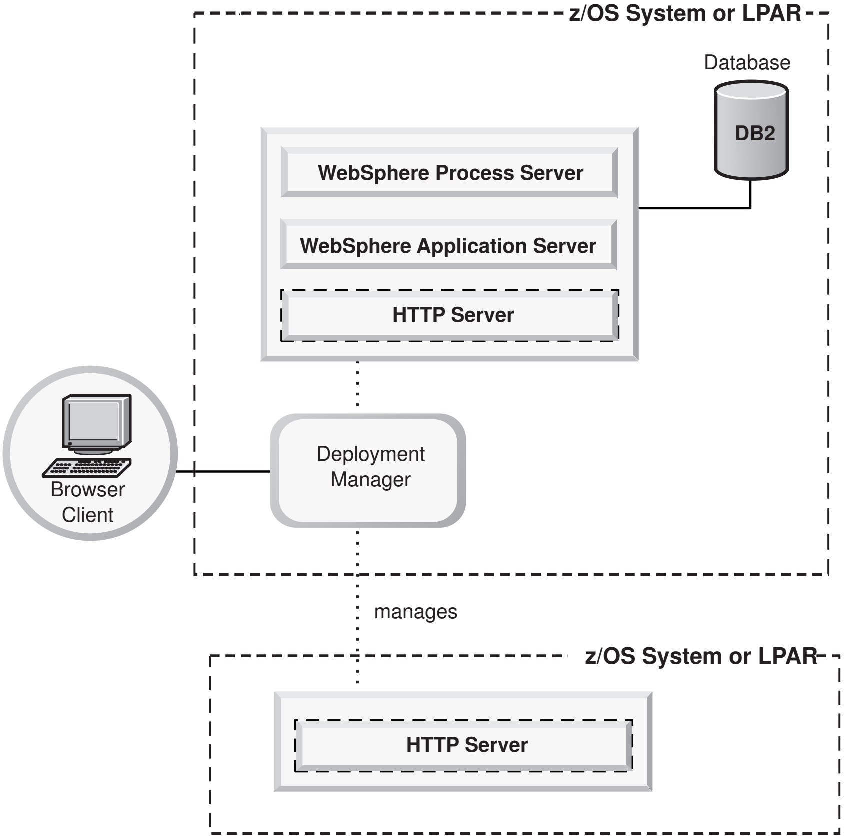


Unmanaged node

A Web server *unmanaged node* is a Web server in a configuration that is not being managed by any deployment manager. It is commonly used when Web servers are installed outside of a firewall where no application server node can be installed.

Task flow: Creating a managed node configuration:

This topic provides a task flow for installing WebSphere Process Server for z/OS and creating a managed node configuration.



Installing WebSphere Process Server for z/OS and creating a managed node configuration involves a series of steps that include WebSphere Application Server for z/OS installation and customization procedures.

The list of steps below represent a sequential task flow for installing WebSphere Process Server for z/OS and creating a managed node configuration. The details on how to perform these tasks are described in this information center and, where appropriate, in the WebSphere Application Server for z/OS information center.

The task flow described in this topic assumes that users have already loaded WebSphere Process Server for z/OS and WebSphere Application Server for z/OS onto the system using SMP/E and that you have configured the WebSphere Process Server for z/OS deployment manager.

Note: Use the program directory for WebSphere Application Server for z/OS and for WebSphere Process Server to guide you through the SMP/E process of unloading the modules from the media onto your z/OS system. The result of completing the SMP/E installation is an installation HFS.

1. Prepare and Plan your installation.

   Make sure you have access to the required documentation for both WebSphere Process Server for z/OS and WebSphere Application Server for z/OS.

   Prepare your z/OS target systems to run WebSphere Process Server, as described in “Preparing the base operating system” on page 56.
2. Create an empty managed node.

   Use the ISPF customization dialog to create an empty managed node, as described in the WebSphere Application Server for z/OS information center.

Note: Do not run the job BBOWMNAN to federate the node into the Deployment manager at this time. The process of adding the node to an existing Network Deployment cell is performed in a later step titled *Federate the managed node into the Deployment Manager*.

Several values specified on ISPF customization will need to be replicated in the managed node response file that is used by the WebSphere Process Server for z/OS configuration script. These values include:

- Cell name
- Node name
- Template path
- Profile path

3. Run the installation script against the WebSphere Process Server for z/OS installation HFS to create the symbolic links and product definitions.

   This step involves running the WebSphere Process Server for z/OS installation script (zSMPInstall.sh) from a command line.

   The installation command contains keyword parameters for the WebSphere Process Server for z/OS smpe_root, as well as the WebSphere Application Server for z/OS configuration HFS / app_server_root, and when it completes, it extends the WebSphere Process Server for z/OS features into the configuration HFS directory. For example: /WebSphere/V6R0/AppServer
4. Edit the properties in the managed node response file (ManagedDB2.rsp).

   WebSphere Process Server for z/OS includes sample response files for a managed node configuration. You will need to edit some of the property values in the response file to tailor it to your environment.

   This response file is used as a keyword parameter in the configuration command. It contains the WebSphere Process Server for z/OS profile information that will be used to augment the default profile that was created when you filled out the customization dialogs and ran the resulting jobs to create a WebSphere Application Server for z/OS managed node configuration.
5. If you have not already done so, create a **DB2JccConfiguration.properties** file.
6. Change directory to the deployment manager bin directory.
7. If you have not already done so, switch to the administrator's user id.
8. Run the WebSphere Process Server for z/OS configuration script.

   As a result of running the configuration script, configuration actions augment the managed node default profile with WebSphere Process Server for z/OS configuration data from the deployment manager response file. Additionally, the DDL and SQL for the database tables is generated.

   Note: You can configure the response file to invoke actions that run the generated DDL and SQL, thereby creating the database definitions automatically. Because the database administrator may not allow automatic updates to the database, you have the option to delay running the DDL and SQL scripts that create the database definitions. You will then need to run these scripts manually, after the configuration process.
9. Federate the managed node into the Deployment Manager.

   This is accomplished by running the job BBOWMNAN, that you did not run when creating an empty managed node.
10. Administer the managed node configuration as described in the topic *Creating a clustered environment* in the section Administering WebSphere Process Server for z/OS.

11. Tune for performance. For best performance on any platform, see Tuning the application serving environment.

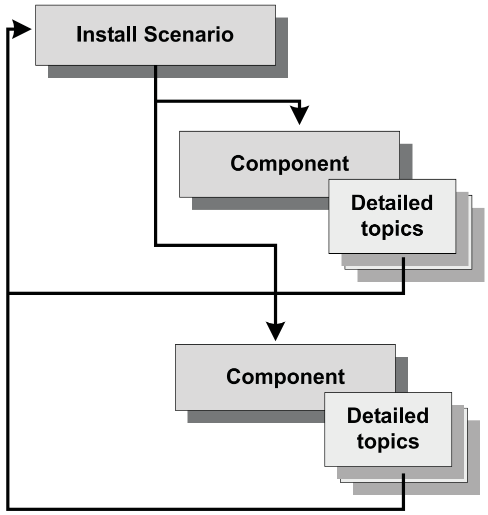
Scenarios for installation and configuration

WebSphere Process Server includes many products and components that you assemble to suit your specific business needs. The topics listed in this section provide information on planning, installing, configuring, and verifying WebSphere Process Server for various scenarios. The following information does not prescribe an installation and configuration, but allows you to build an environment that is right for you.

Before you begin

Plan your WebSphere Process Server environment. Although individual components have specific planning considerations, it is recommended that you plan your environment before starting installation.

How to use these scenarios



1. Choose an installation scenario. Within the scenario, follow the links for each component.
2. Within the component, follow the ordered links to access detailed topics that describe how to plan, install, configure, and verify the component. Each step includes a link to more detailed instructions.
3. After you are finished with a component, return to the scenario to proceed to the next component.

Installation scenarios

The following scenarios provide the recommended order to install and configure WebSphere Process Server and its components:

- Performing an installation and creating stand-alone configuration
- Performing an installation and creating a network deployment configuration
- Performing an installation and creating a managed node configuration
- Performing an uninstall

Planning for a stand-alone application server cell:

A stand-alone application server cell is the simplest WebSphere Process Server for z/OS configuration on which you can deploy and run WebSphere Process Server for z/OS 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.

A WebSphere Process Server for z/OS stand-alone application server configuration supports a Cloudscape or a DB2 for z/OS database.

A stand-alone WebSphere Process Server for z/OS configuration with a Cloudscape database can be created with minimal customization to the response file. For this configuration you should be able to achieve a working WebSphere Process Server for z/OS cell with the least amount of effort.

Note: A stand-alone WebSphere Process Server for z/OS configuration with a Cloudscape database is not appropriate for a production environment.

A stand-alone WebSphere Process Server for z/OS configuration with a DB2 for z/OS database requires prerequisite tasks for setting up DB2 for z/OS as well as more tailoring of the response file.

For more complicated or robust WebSphere Process Server for z/OS application-serving environments, the network deployment cell configuration is recommended.

Planning for a network deployment cell:

A network deployment cell is a full-function WebSphere Process 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:

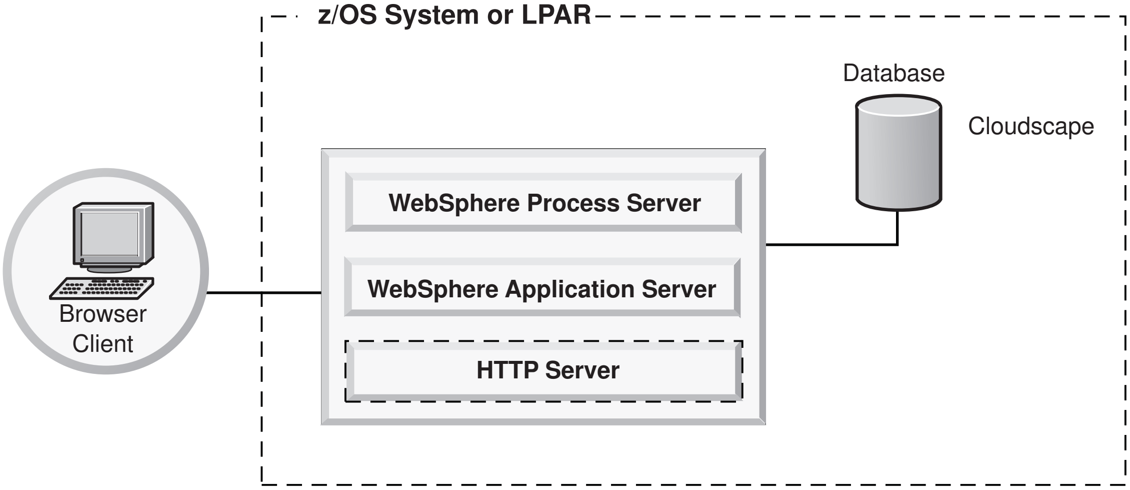
- When allocating target data sets for this option, it is possible, though not recommended, to use the same target data sets that you used for the stand-alone application server node. The jobnames for each configuration are very close to one another; and if you use the same target data sets, you might 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 HFSs such that the root HFS is shared among all processors and the deployment manager's configuration is in a configuration HFS on a system-generic mount point.

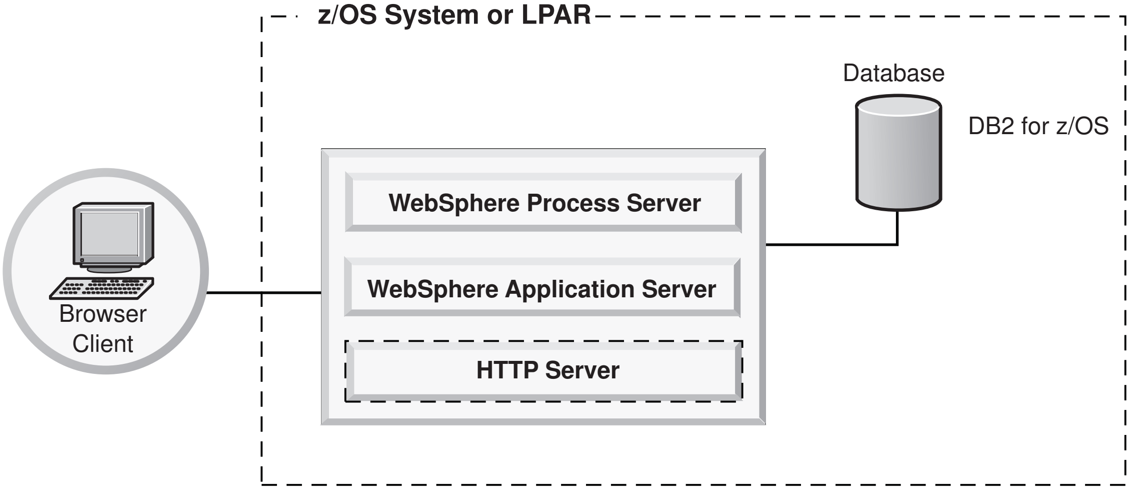
Scenario: Installing WebSphere Process Server for z/OS and creating a stand-alone configuration:

This scenario outlines how to install WebSphere Process Server for z/OS and create a stand-alone configuration.

Refer to the following outline and choose the options for a WebSphere Process Server installation in a stand-alone configuration:

A WebSphere Process Server for z/OS stand-alone configuration supports either a Cloudscape or a DB2 for z/OS database





1. WebSphere Application Server for z/OS
   - Installing your application serving environment
2. WebSphere Process Server for z/OS
   - Preparing to install on z/OS
   - Preparing the base z/OS environment

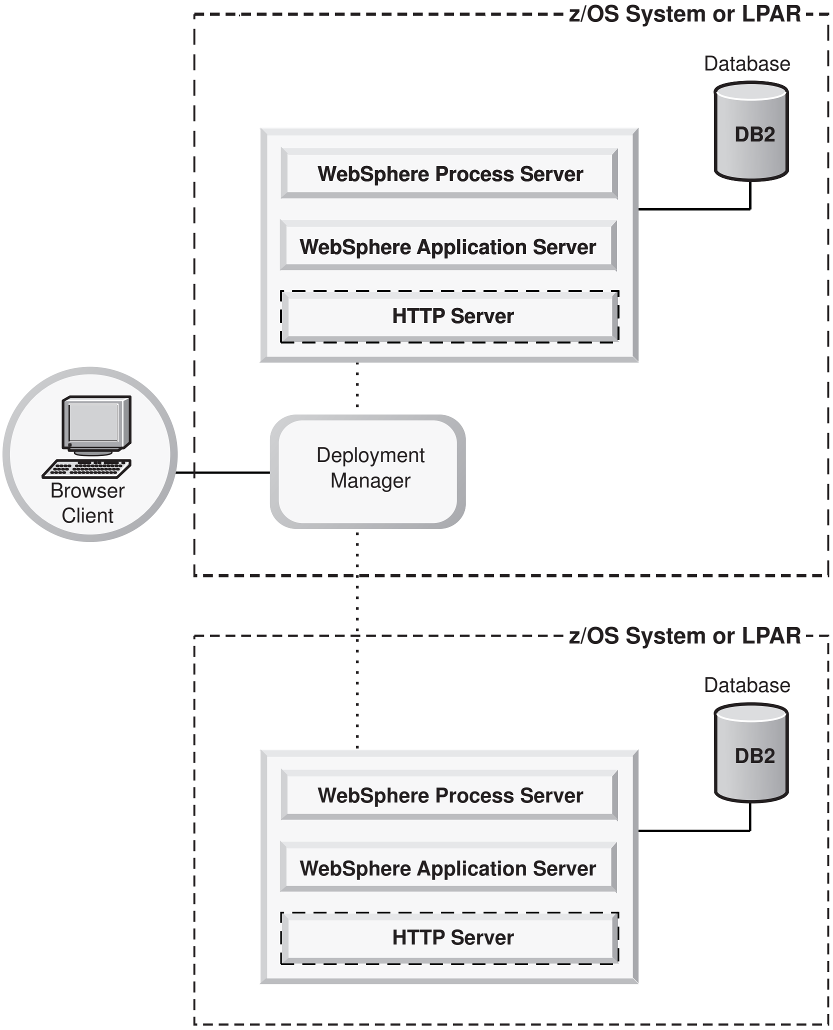
     Note: If the version of WebSphere Process Server for z/OS that resides on the system does not meet the prerequisite level required by the product you are installing, SMP/E will halt the installation. WebSphere Process Server for z/OS service may need to be applied to ensure that the base product is at the level to support the product being installed.
   - Planning to load the product code
   - Load the product code from the installation mediaWebSphere Process Server for z/OS
   - Verifying the product code was unloaded
   - Running the installation script
     - Log onto system with the appropriate credentials to install the product.
     - Run the installation script, **zSMPInstall.sh**, with the `-install` argument
   - Considerations for creating the database
   - Create the database and storage groups
   - Work with response files
   - Configuring
   - Verifying the installation and configuration.
3. Business Process Choreographer
   - Planning to use Business Process Choreographer
   - Configuring Business Process Choreographer.
4. Common Event Infrastructure
   - Administering theCommon Event Infrastructure.

You can now begin using your installation.

Scenario: Installing WebSphere Process Server for z/OS and creating a deployment manager configuration:

This scenario outlines how to create a WebSphere Process Server for z/OS Deployment Manager configuration.

Refer to the following outline and choose the options for a WebSphere Process Server for z/OS installation in a network deployment configuration:



1. WebSphere Application Server for z/OS
   - Installing your application serving environment
2. WebSphere Process Server for z/OS
   - Preparing to install on z/OS
   - Preparing the base z/OS environment

     Note: If the version of WebSphere Process Server for z/OS that resides on the system does not meet the prerequisite level required by the product you are installing, SMP/E will halt the installation. WebSphere Process Server for z/OS service may need to be applied to ensure that the base product is at the level to support the product being installed.
   - Planning to unload the product code
   - Loading the product code from the installation mediaWebSphere Process Server for z/OS

- Verifying the product code was unloaded
- Running the installation script
  - Log onto system with the appropriate credentials to install the product.
  - Run the installation script, **zSMPInstall.sh**, with the `-install` argument
- Considerations for creating the database
- Create the database and storage groups
- Work with response files
- Configuring
- Verifying the installation and configuration

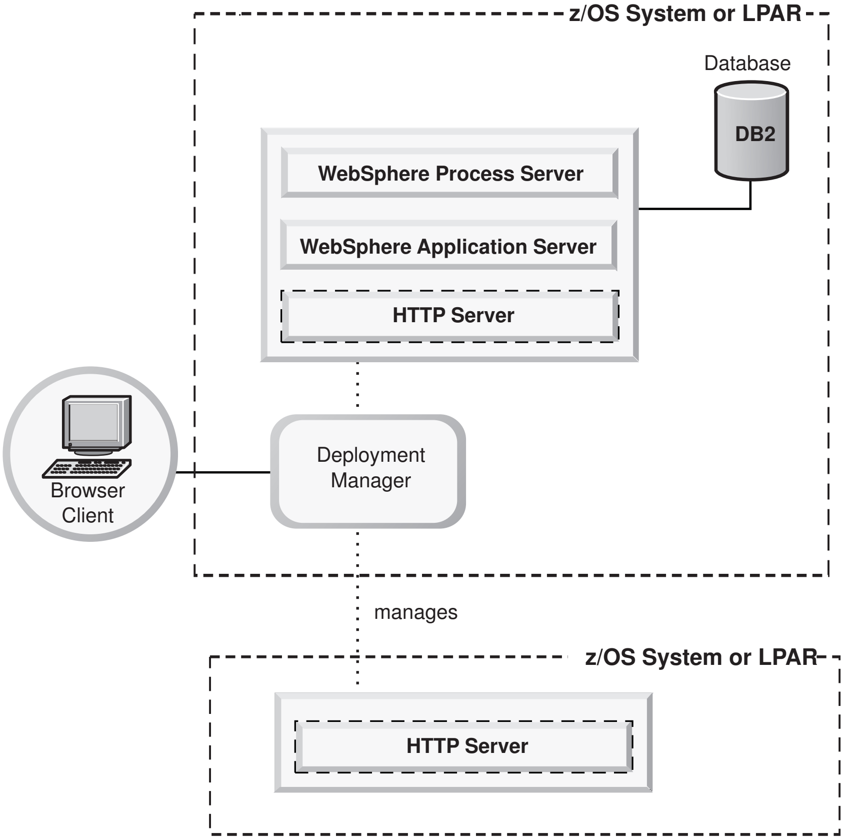
3. Business Process Choreographer
   - Planning to use Business Process Choreographer
   - Configuring Business Process Choreographer.
4. Common Event Infrastructure
   - Administering theCommon Event Infrastructure.

You can now begin using your installation.

Scenario: Installing WebSphere Process Server for z/OS and completing a network deployment configuration:

This scenario outlines how to install WebSphere Process Server for z/OS and complete a network deployment configuration.

Refer to the following outline and choose the options for a WebSphere Process Server for z/OS installation in a managed node configuration:



1. WebSphere Application Server for z/OS

- Installing your application serving environment

2. WebSphere Process Server for z/OS
   - Preparing to install on z/OS
   - Preparing the base z/OS environment

     Note: If the version of WebSphere Process Server for z/OS that resides on the system does not meet the prerequisite level required by the product you are installing, SMP/E will halt the installation. WebSphere Process Server for z/OS service may need to be applied to ensure that the base product is at the level to support the product being installed.
   - Planning to unload the product code
   - Load the product code from the installation media
   - Verifying the product code was unloaded.
   - Running the installation script.
     - Log onto system with the appropriate credentials to install the product.
     - Run the installation script, **zSMPInstall.sh**, with the `-install` argument
   - Considerations for creating the database
   - Create the database and storage groups
   - Work with response files
   - Configuring.
   - .Verifying the installation and configuration.
3. Business Process Choreographer
   - Planning to use Business Process Choreographer
   - Configuring Business Process Choreographer.
4. Common Event Infrastructure
   - Administering theCommon Event Infrastructure.

You can now begin using your installation.

Scenario: Uninstalling:

This scenario outlines how to uninstall WebSphere Process Server for z/OS.

The following scenario describes how to perform a complete uninstall of WebSphere Process Server for z/OS.

Note: There is no support for a partial or incremental uninstall.

WebSphere Process Server for z/OS

- Uninstalling WebSphere Process Server for z/OS
  - Log onto system with the appropriate credentials to install the product.
  - Run the installation script, **zSMPInstall.sh**, with the `-uninstall` argument. The script performs the following tasks:
    - Disables the WebSphere Process Server for z/OS by running Configuration Manager scripted actions. This removes any Administrative Console plug-in extensions.
    - Removes Profile Augmentation using the WSPROFILE scripted actions. This results in the unaugmentation of the WebSphere Process Server for z/OS default profile.

- Deletes the post install file.
- Removes code base permissions.

Note: If any shared common components are being shared by other applications the command line prompting warns that un-installation of the product may cause other applications to no longer function correctly.

Uninstall is blocked at this point unless the user specifically selects to continue the un-installation by affirming the wish to continue with the uninstall process.

The user is warned either that augmented default profile will be deleted (if underlying WebSphere Application Server for z/OS or WebSphere Application Server Network Deployment for z/OS is being uninstalled) or that augmented profiles will no longer be usable.

The WebSphere Process Server for z/OS product has been removed from your system.

Planning to use Business Process Choreographer

To plan your configuration of the business process container and the human task container, go to the WebSphere Process Server for z/OS, version 6.0.2, information center and review the topics under **Installing and configuring WebSphere Process Server > Planning to install WebSphere Process Server > Planning to use Business Process Choreographer**. You can also find this information in the *Business Process Choreographer PDF*.

Installing the software

Installing WebSphere Process Server involves loading the product from tape and running the installation script to create the product definitions. You can also optionally install the Message Service clients on the platforms that they support.

Loading the product code from the installation media onto z/OS

The product code for WebSphere Process Server for z/OS is installed using either an IBM ServerPac/SystemPac or an IBM Custom-Built Product Delivery Option (CBPDO). In a z/OS environment, loading product from media onto the system is usually the responsibility of a system programmer.

1. Make sure that the supported version of WebSphere Application Server for z/OS has been installed and customized.

   For information on how to install and customize WebSphere Application Server for z/OS, see Installing your application serving environment.
2. Complete the steps in *Planning to load the product code onto the system*.
3. Have access to the Program Directory for WebSphere Process Server for z/OS .

   The program directory is the primary user assistance for performing the SMP/E installation.

   You can download the program directory in PDF format from the WebSphere Process Server for z/OS download page, at http://www-306.ibm.com/software/integration/wps/library/infocenter/.

This section of the documentation provides guidance on using either CBPDO or an IBM ServerPac/SystemPac to install the WebSphere Process Server for z/OS program materials onto the z/OS system.

1. Order an IBM ServerPac/SystemPac or IBM CBPDO that contains the appropriate WebSphere Process Server for z/OS product.
2. Follow the instructions for the delivery vehicle that you choose:
   a. *Using an IBM SystemPac or ServerPac*

      An IBM SystemPac or ServerPac consists of loadable product libraries and corresponding SMP/E data sets.
   b. *Using an IBM Custom-Build Product Delivery Offering* .

      A CBPDO contains SMP/E relfiles and maintenance for one or more products.
3. File the installation materials for later use during product maintenance.

When you are finished unloading the product code from the install media onto the system, notify the product administrator so that he or she can run the installation script (**zSMPInstall.sh**) to create the WebSphere definitions that enable the product for use. See *Running the installation script* for more information.

Using an 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.

  See Planning for product data sets for more information.
- If you are installing from a driving system, make sure the maintenance level of the target system meets requirements for WebSphere Process Server for z/OS.
- When installation is complete, make sure the product data sets are available to your z/OS target system(s) and the product code HFS is mounted at `/usr/lpp/zWPS/V6R0` 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 web site at http://www-306.ibm.com/software/support/.

Using an IBM Custom-Build Product Delivery Option

An IBM Custom-Build Product Delivery Option (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 Process 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.

  See Planning for product data sets for more information.
- If you are installing from a driving system, make sure the maintenance level of the target system meets requirements for WebSphere Process Server for z/OS.
- When installation is complete, make sure the product data sets are available to your z/OS target system(s) and the product code HFS is mounted at `/usr/lpp/zWPS/V6R0` 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 web site at http://www-306.ibm.com/software/support/.

Verify the product code was unloaded

Verify that the WebSphere Process Server for z/OS SMP/E installation was successful.

- **Verify that the code was successfully unloaded onto the system**.

  The systems administrator relies on the *Program Directory for WebSphere Process Server for z/OS* for information concerning the material and procedures associated with the loading the contents of the product tapes onto the z/OS operating system.

  The verification of the SMP/E installation is an iterative process, where return codes indicate the success or failure of the various jobs run to load the contents of the product tape onto the system.

  There is no Installation Verification Test (IVT) for the SMP/E installation. It is assumed that the systems administrator has SMP/E expertise and access to SMP/E documentation to facilitate this portion of the installation.

  The following SMP/E documentation is recommended:
  - z/OS: SMP/E Commands, SA22-7771
  - z/OS: SMP/E Messages, Codes, and Diagnosis, GA22-7770
  - z/OS: SMP/E Reference, SA22-7772
  - z/OS: SMP/E User’s Guide, SA22-7773
  - z/OS: MVS Initialization and Tuning Reference, SA22-7592

Now that the product code has been loaded onto the system successfully, you can begin to plan to install the WebSphere definitions that enable the product for use.

About running the configuration script

Run the configuration script to augment the default profile with WebSphere Process Server for z/OS configuration data.

The configuration script relies on response files for specific configuration information. So, the response files provided represent the configurations supported by the product.

When you run the configuration script, the command includes a path to the response file for a particular configuration. The property values that you set in the response file are used to create your product configuration.

The following is a list of configurations supported by the product configuration script.

- Stand-alone server configuration supporting the following databases:
  - Cloudscape
  - DB2 for z/OS version 7
  - DB2 for z/OS version 8

  Default response files for these configurations reside on the install HFS, as follows:
  - /usr/lpp/zWPS/V6R0/usr/lpp/zWPS/V6R0/zos.config/**standAloneProfile.rsp**
  - /usr/lpp/zWPS/V6R0/usr/lpp/zWPS/V6R0/zos.config/**standAloneProfileDB2.rsp**

  Note: The information required to create the DB2 for z/OS database is in the response file named standAloneProfileDB2.rsp. This response file contains properties that can be set to support DB2 for z/OS version 7 or DB2 for z/OS version 8. The default values are set for DB2 for z/OS version 8.
- Network Deployment configuration supporting the following databases:
  - DB2 for z/OS version 7
  - DB2 for z/OS version 8

  The default response file for these configurations resides on the install HFS, as follows:
  - /usr/lpp/zWPS/V6R0/usr/lpp/zWPS/V6R0/zos.config/**dmgrProfile.rsp**

  Note: The information required to create the DB2 for z/OS database is in the response file named dmgrProfile.rsp. This response file contains properties that can be set to support DB2 for z/OS version 7 or DB2 for z/OS version 8. The default values are set for DB2 for z/OS version 8.
- Managed node configuration supporting the following databases:
  - DB2 for z/OS version 7
  - DB2 for z/OS version 8

  he default response file for these configurations resides on the install HFS, as follows:
  - /usr/lpp/zWPS/V6R0/usr/lpp/zWPS/V6R0/zos.config/**managedProfile.rsp**

  Note: The information required to create the DB2 for z/OS database is in the response file named managedProfile.rsp. This response file contains properties that can be set to support DB2 for z/OS version 7 or DB2 for z/OS version 8. The default values are set for DB2 for z/OS version 8.

Although the process of running the command is almost identical for each of the WebSphere Process Server for z/OS configurations (the difference being the response file that is referenced in the command), there are nuances with regard to each of the WebSphere Process Server for z/OS configurations. Before you run the configuration script from the command line, read about each of the WebSphere Process Server for z/OS types.

Running the installation script

Run the installation script (**zSMPInstall.sh**) to install and create the required WebSphere definitions for WebSphere Process Server for z/OS.

Running the installation script will result in the following actions:

- Create the symlinks from the WebSphere Application Server for z/OS configuration HFS to the WebSphere Process Server for z/OS smpe_root (installation HFS).
- Invoke Ant script actions that update the administrative console with the WebSphere Process Server for z/OS product definitions.
- Runs **applyPTF** processing as required.

For detailed description of the actions invoked by the installation script, see About the installation script.

Running the installation script for WebSphere Process Server for z/OS requires a working knowledge of z/OS UNIX System Services. You may want to have access to the following documentation:

- z/OS V1R7.0 UNIX System Services User's Guide
- z/OS V1R7.0 UNIX System Services Command Reference

Make sure of the following:

1. Review Task overview: installation and configuration for an understanding of the task flow of the installation and configuration process.
2. That the system programmer who is in charge of security and system access at your installation has set you up so that you can properly access the shell.

   This involves making modifications to your RACF® (security) profile and creating a home directory within the UNIX shell. The home directory is where you always start off when you begin a UNIX session. Within this directory, you keep any environmental variable files you may need to execute programs. These files contain information that's required in certain languages, like the location of the Java classes that are used in Java programs. You can also use the home directory as the root directory for keeping your work data. See Resource Access Control Facility Tools in the WebSphere Application Server for z/OS for information on tools and techniques to help you manage the security definitions used for WebSphere Application Server for z/OS.
3. The WebSphere Application Server for z/OS is installed and customized.

   The system programmer has completed the SMP/E installation for WebSphere Application Server for z/OS and has performed the required customization tasks and has run the resulting jobs to create a configuration HFS for the desired configuration type (stand-alone application server or deployment manager).
4. The WebSphere Process Server for z/OS product code has been loaded onto the system.

The system programmer has completed the SMP/E installation by successfully loading the contents of the WebSphere Process Server for z/OS installation media onto the system.

Perform this task when your goal is to create the required WebSphere Process Server for z/OS definitions.

1. Stop the server if it is still running
   a. For a WebSphere Application Server for z/OS network deployment configuration, stop the deployment manager server. You can use the **stopManager** command to stop the deployment manager.
   b. For a WebSphere Application Server for z/OS stand-alone server configuration, stop the stand-alone server. You can use the **stopServer** command to stop a single server.

   See Stopping an application server in the WebSphere Application Server for z/OS information center for detailed information on how to stop the server.
2. Back up your HFS. For information on backing up configuration and data for the WebSphere Application Server for z/OS system, see Backing up the WebSphere Application Server for z/OS system.
3. Get to where you will run the WebSphere Process Server for z/OS installation script.

   To run the installation script from an OS/390 UNIX command shell, perform the following:
   a. Access the OS/390 UNIX command shell.
   b. Enter the TSO command OMVS at the ISPF Command Shell or TSO OMVS from any other ISPF panel.
   c. Once you're in the UNIX shell, a command prompt (usually a dollar ($) or pound (#) sign) indicates that the system is ready to accept input.

   To run the installation script from telnet, perform the following:
   a. Return to the telnet session if need be.

      Type the following from the system prompt: **telnet TCPIP-Address port number**

      Here is an example of a Telnet command: `telnet 99.9.99.999 2023`, where 99.9.99.999 is the TCP IP-Address and 2023 is the port number.
   b. If you did not log on to the telnet session using the administrator's user id, you must switch to the administrator's user id. For example,
      ```
      su wsadmin
      ```
   c. Change to zos.config in the WebSphere Process Server for z/OS smpe_root directory:
      ```
      cd /usr/lpp/zWPS/V6R0/zos.config/bin
      ```
4. Add the current directory to the path: export PATH=.:$PATH.
5. Run the installation script Use the following syntax diagram as guide on how run the command.

Syntax diagram for running the WebSphere Process Server for z/OS installation script

```
►►──zSMPInstall.sh──-smproot──/usr/lpp/zWPS/V6R0──────────────────────────►
```

```
                                               (1)
►─-runtime─/WebSphere/V6R0M0/AppServer─────-install───────────────────────►

                                  (2)
►─-trace─┬─trace arguments─┬───────────────────────────────────────────────►◄
         └─*=all=enabled───┘
```

Notes:

1 Type the keyword parameter that applies to your installation environment. The keyword parameters for -smproot and -runtime are user defined and may vary from the default values shown in this example. The keyword parameter value for -smproot represents the smpe_root directory (installation HFS) for the WebSphere Process Server for z/OS product that resulted from the SMP/E installation. The keyword parameter value for -runtime represents the WebSphere Application Server for z/OS configuration HFS and app_server_root directory name combined (/WebSphere/V6R0/AppServer for example). 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 WebSphere Application Server for z/OS configuration process and contain symbolic links to files in the product directory. The -runtime path must match what was set when you created your WebSphere Application Server for z/OS application serving environment and will vary by configuration type. For example, the configuration root for a deployment manager may be `/WebSphere/V6R0/DeploymentManager`, while the configuration root for a stand-alone application server may be `/WebSphere/V6R0/AppServer`.

2 By including **-trace** the command writes to the appropriate trace file.

Review the messages from Standard Out. These are either displayed on the screen from which you ran **zSMPInstall.sh**, or in the file specified if the redirect **(**″>″**)** symbol was used on the command line.

There should be no error messages displayed.

An example of a successful execution of **zSMPInstall.sh** is as follows:

```
parsing command arguments...
parsing arguments complete
setting up configuration...
runtimeRootDirName is: /WebSphere/V6R0M0/AppServer
WAS_HOME is: /WebSphere/V6R0M0/AppServer
WBI_HOME is: /WebSphere/V6R0M0/AppServer
set up configuration complete
creating the symbolic links...
invokeSymLink
creation of symbolic links complete
doing post install file updates...
post install updates complete
running Configuration Manager update...
Configuration Manager update complete
```

```
CWPIZ0256I: set up configuration complete
CWPIZ0257I: creating the symbolic links...
CWPIZ0259I: creation of symbolic links complete
CWPIZ0260I: doing post install file updates...
CWPIZ0262I: post install updates complete
CWPIZ0263I: running Configuration Manager update...
Sep 27, 2006 10:47:57 PM com.ibm.ws390.installer.WPSInstaller
```

```
INFO: BBZWI218
Sep 27, 2006 10:53:35 PM com.ibm.ws390.installer.WPSInstaller
INFO: BBZWI219
CWPIZ0264I: Configuration Manager update complete
Sep 27, 2006 10:53:35 PM com.ibm.ws390.installer.WPSInstaller
INFO: BBZWI016
```

If the software did not install successfully, see “Troubleshooting the installation and configuration” on page 215 for information on how to assess installation problems.

After you run the installation script successfully, you are ready to configure the software. For more information, see “Configuring the software” on page 113.

Installing the WebSphere Process Server Client

WebSphere Process Server for z/OS is packaged with CD-ROMs that allow you to install a WebSphere Process Server Client on a Linux, UNIX, or Windows machine for use in a WebSphere Process Server for z/OS environment. A Client installation provides the WebSphere Process Server for z/OS configuration with SOACore and Business Process Choreographer functionality without the need for a full WebSphere Process Server installation.

Before installing the WebSphere Process Server Client on a Linux, UNIX, or Windows machine, ensure that:

- You have access to the WebSphere Process Server for Multiplatforms 6.0.2 information center.

  Because you are installing the WebSphere Process Server Client on a Linux, UNIX, or Windows machine, you might need to review content in this information center.
- You have decided whether to install WebSphere Application Server Network Deployment as part of your WebSphere Process Server Client installation or to instead use an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment, version 6.0.x.x.
- Your system meets all hardware requirements and that you have installed all required software corequisites and prerequisites. See WebSphere Process Server detailed system requirements at http://www.ibm.com/support/docview.wss?uid=swg27006205 and select the link to your version of WebSphere Process Server.
- You have prepared the operating system on the Client machine for installation. See Preparing the operating system for installation in the WebSphere Process Server for Multiplatforms 6.0.2 information center.
- You are installing the Client as the root user on a Linux or UNIX system, or as a member of the Administrator group on a Windows system.
- You are installing the Client from one of the following CD-ROMs:
  - WebSphere Process Server 6.0.2 Client for Windows
  - WebSphere Process Server 6.0.2 Client for AIX
  - WebSphere Process Server 6.0.2 Client for Solaris on SPARC
  - WebSphere Process Server 6.0.2 Client for Solaris on x86 64-bit
  - WebSphere Process Server 6.0.2 Client for HP-UX
  - WebSphere Process Server 6.0.2 Client for HP-UX on Itanium 64-bit
  - WebSphere Process Server 6.0.2 Client for Linux on x86 32-bit
  - WebSphere Process Server 6.0.2 Client for Linux on x86 64-bit
  - WebSphere Process Server 6.0.2 Client for Linux on iSeries and pSeries

- – WebSphere Process Server 6.0.2 Client for Linux on zSeries 31-bit
- – WebSphere Process Server 6.0.2 Client for Linux on zSeries 64-bit

This procedure assumes that you want to install interactively from a CD-ROM.

Use the following procedure to install the Client.

1. Log on as the root user on a Linux or UNIX system, or as a member of the Administrator group on a Windows system.
2. Place the CD-ROM into the CD-ROM drive of the machine you are installing on and mount the drive if necessary, as described in Mounting CD-ROMs on Linux and UNIX operating systems in the WebSphere Process Server for Multiplatforms 6.0.2 information center.
3. Start the Client Installation wizard by navigating to the WBI directory and doing the following:
   - **On Windows platforms**: Run the install.exe file.
   - **On Linux and UNIX platforms**: Run the command **install**.
4. From the Welcome panel of the installer, select **Next**.
5. In the Software License Agreement panel, review the IBM and non-IBM licensing terms, select **I accept both the IBM and the non-IBM terms**, and select **Next**.

   The Installation wizard checks for a supported operating system with prerequisite patches. At the end of the process, the Checking prerequisites panel indicates whether your system passed the check. This procedure assumes that your system passed.

   If your system did not pass, cancel the installation, make the required changes, and restart the installation.
6. In the Checking prerequisites panel, select **Next**

   The Installation wizard checks for existing installations of WebSphere Process Server, WebSphere Process Server Client, WebSphere Enterprise Service Bus, WebSphere Application Server, and WebSphere Application Server Network Deployment. This procedure assumes you do not have an existing WebSphere Process Server, Client, or WebSphere Enterprise Service Bus installation, but that you might have an existing WebSphere Application Server or WebSphere Application Server Network Deployment installation.

   If you *do not* have an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment, the Installation location panel is displayed. Proceed to Step 8.

   If you *do* have an existing installation of one of these products, an Existing installation detected panel is displayed. Proceed to step 7.

   Important: The installer will also detect unregistered instances of WebSphere Application Server or WebSphere Application Server Network Deployment if they have entries in the`.WASRegistry` file. This file is located in the `$USER_HOME` directory of the installation. Using an unregistered installation of one of these products with your WebSphere Process Server Client installation is neither recommended nor supported.
7. In the Existing installation detected panel, do one of the following:
   - If you want to install a new copy of WebSphere Application Server Network Deployment, select **Install a new copy of WebSphere Application Server Network Deployment, Version 6.0** and select **Next**. A warning panel outlines considerations you must make when installing the product on a

system with an existing installation. Select **Next** to close the warning panel. The Installation location panel is displayed. Proceed to step 8.

- If you want to install the Client over an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment, select **Use an existing installation of WebSphere Application Server Network Deployment, Version 6.0** or **Use an existing installation of WebSphere Application Server, Version 6.0** and select **Next**. (If you have multiple installations, be sure to select the one you want to use from the drop-down list.) A warning panel alerts you to stop servers before continuing if the application you chose has running servers. Select Next to close the warning panel. The Installation type panel is displayed. Proceed to step 9.

8. In the Installation location panel, accept the default installation root directory, or specify a different directory, and select **Next**.

   See the topic on Default installation paths in the WebSphere Process Server for Multiplatforms 6.0.2 information center for the default installation directories.

   Important:

   - You must provide a value for the installation root directory to continue.
   - Do not use non-ASCII special characters in directory names – they are not supported.
   - **On Linux and UNIX platforms**: Do not use symbolic links as the installation root directory – they are not supported. Also, do not use spaces in the directory path.
   - **On Linux platforms**: The installation root directory path must be no longer than 256 characters to successfully install the products.

9. The Installation type panel displays **Client installation** as the only type of installation allowed. Click **Next**.

10. In the Installation summary panel, review the components that will be installed, the amount of space they will consume, and where they will be located on the system, and select **Next** to install or **Back** to change your specifications.

    WebSphere Application Server Network Deployment and WebSphere Process Server are the only products installed.

    The Installation wizard creates the uninstaller program and shows a progress panel to indicate that components are being installed.

    If you elected to use an existing version of WebSphere Application Server or WebSphere Application Server Network Deployment, the Installation wizard examines it and does one of the following:

    - Does nothing if the installation is at the correct service level.
    - If the installation is at an earlier service level, applies the necessary fixes to bring the installation up to the appropriate level and also applies any necessary interim fixes.

    At the end of the installation, the Installation complete panel is displayed.

Attention: If errors are detected during installation, other panels might be displayed in place of the Installation complete panel. Examples include the following:

- Installation is complete with errors panel, which indicates that the installation completed but errors were generated.
- Installation failed panel, which indicates that the installation failed completely.

Each of these panels identifies the log file to reference in order to troubleshoot the problems. See the following topics in the WebSphere Process Server for Multiplatforms 6.0.2 information center:

- Troubleshooting the installation, for tips on troubleshooting your installation.
- Log files, for descriptions of relevant log files.
- Error messages: installation and profile creation and augmentation, for descriptions of error messages.

11. Click **Finish** to close the Installation wizard.

If the Installation complete panel is displayed, the WebSphere Process Server Client was installed successfully.

You are now ready to begin using the Client.

Installing the WebSphere Process Server Client silently

You can silently install the WebSphere Process Server Client on a Linux, UNIX or Windows machine that is part of a z/OS installation by using a file called a response file. In this case, instead of displaying a graphical user (″wizard″) interface, the silent installation causes the installation program to read all of your responses from a file that you provide. The response file `responsefile.wpsclient.txt` can be used to silently install the Client. The response file is packaged with default values on the Client CD-ROMs that are shipped with WebSphere Process Server for z/OS.

Before you begin

Before installing the WebSphere Process Server Client silently on a Linux, UNIX, or Windows machine, ensure that:

- You have access to the WebSphere Process Server for Multiplatforms 6.0.2 information center.

  Because you are installing the WebSphere Process Server Client on a Linux, UNIX, or Windows machine, you might need to review content in this information center.
- You want to install silently. If you would rather install the Client interactively, see Installing the WebSphere Process Server Client.
- You have decided whether to install WebSphere Application Server Network Deployment as part of your WebSphere Process Server Client installation or to instead use an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment, version 6.0.x.x.
- Your system meets all hardware requirements and that you have installed all required software corequisites and prerequisites. See WebSphere Process Server detailed system requirements at http://www.ibm.com/support/docview.wss?uid=swg27006205 and select the link to your version of WebSphere Process Server.

- You have prepared the operating system on the Client machine for installation. See Preparing the operating system for installation in the WebSphere Process Server for Multiplatforms 6.0.2 information center.
- You are installing the Client as the root user on a Linux or UNIX system, or as a member of the Administrator group on a Windows system.
- You are installing the Client silently by using a response file from one of the following CD-ROMs:
  - WebSphere Process Server 6.0.2 Client for Windows
  - WebSphere Process Server 6.0.2 Client for AIX
  - WebSphere Process Server 6.0.2 Client for Solaris on SPARC
  - WebSphere Process Server 6.0.2 Client for Solaris on x86 64-bit
  - WebSphere Process Server 6.0.2 Client for HP-UX
  - WebSphere Process Server 6.0.2 Client for HP-UX on Itanium 64-bit
  - WebSphere Process Server 6.0.2 Client for Linux on x86 32-bit
  - WebSphere Process Server 6.0.2 Client for Linux on x86 64-bit
  - WebSphere Process Server 6.0.2 Client for Linux on iSeries and pSeries
  - WebSphere Process Server 6.0.2 Client for Linux on zSeries 31-bit
  - WebSphere Process Server 6.0.2 Client for Linux on zSeries 64-bit
- Do not use the `-silent` option within the response file.

Response file location

The example `responsefile.wpsclient.txt` response file can be found in the `WBI` directory of the WebSphere Process Server Client CD-ROM you are using to install the software. It does not get installed.

Important: **On AIX platforms:** To prepare the file for a silent installation on AIX, use UNIX line-end characters (0x0D0A) to end each line of the options response file. The safest method of preparing the file is to edit the file on the target operating system.

Required disk space

For information on how much disk space will be required, see WebSphere Process Server detailed system requirements at http://www.ibm.com/support/docview.wss?uid=swg27006205 and select the link to your version of WebSphere Process Server.

Installing using the response file

To install silently, do the following:

1. From the CD-ROM associated with the platform onto which you are installing the Client, copy the file `responsefile.wpsclient.txt` from the `WBI` directory onto a place that you can easily identify on your machine and save it with a new name, such as `myoptionsfile`.

   Mount the drive if necessary, as described in Mounting CD-ROMs on Linux and UNIX operating systems in the WebSphere Process Server for Multiplatforms 6.0.2 information center.
2. Edit this file using a flat file editor of your choice, on the target operating system, to customize it with the parameters for your system. Read the directions within the response file to choose appropriate values. An example response file is shown in “Example responsefile.wpsclient.txt file” on page 102.

Important: Make sure that you change the license acceptance statement in the file to a value of ″true″. Leaving it with a value of ″false″ will cause the installation to fail.

3. Save your changes in your copy of responsefile.wpsclient.txt.
4. Use one of the following commands to install the WebSphere Process Server Client. The commands shown assume that you have copied the responsefile.wpsclient.txt file into a temporary directory and renamed it as myoptions.txt before customizing the file.
   - **On Linux and UNIX platforms:**

     ```
     install -options /tmp/WBI/myoptions.txt -silent
     ```
   - **On Windows platforms:**

     ```
     install.exe -options "C:\temp\WBI\myoptions.txt" -silent
     ```

Verify the success of the installation by examining the log.txt log file. If the last line of the file contains the word INSTCONFSUCCESS, the WebSphere Process Server Client and WebSphere Application Server Network Deployment (if selected) were installed successfully. The log file is located as follows, where *install_root* represents the location of the WebSphere Process Server Client installation:

- **On Linux and UNIX platforms:** *install_root*/logs/wbi/log.txt
- **On Windows platforms:** *install_root*\logs\wbi\log.txt

If this log file contains the string INSTCONFSUCCESS on the last line, then the installation was successful. Note that other terms such as INSTCONFPARTIALSUCCESS or INSTCONFFAILED can occur on other lines within the file, or even on the last line, but as long as INSTCONFSUCCESS is included in the last line, the installation was successful.

If the installation was not successful, examine other log files to determine why.

For tips on troubleshooting your installation, examine the section Troubleshooting below and see the Troubleshooting the installation section in the WebSphere Process Server for Multiplatforms 6.0.2 information center.

Troubleshooting

Be precise when supplying values in the file: Customize the response file precisely to let the installation program read the option values that the file contains. Incorrect specifications could cause the installation to fail. For example, always use the correct case within property names, which are case-sensitive. In addition, always enclose values in double quotation marks.

If the error is an invalid option value, InstallShield for Multiplatforms (ISMP) displays a warning message that you must confirm and stops the installation.

Compare your options response file to the responsefile.wpsclient.txt file that is shipped with the product to make the necessary corrections. After correcting the file, reinstall.

Certain events can prevent the InstallShield for Multiplatforms (ISMP) from starting the Installation wizard silently (for example, not having enough disk space to launch the Installation wizard). If your installation fails and there is no information in the installation logs, use the -log parameter to record entries for events that cause the ISMP program to fail to start the Installation wizard. The syntax of the **install** command for logging such events is as follows:

On AIX platforms:

```
install -options "/usr/IBM/WebSphere/silentFiles/myresponsefile.txt"
        -silent -log # !/usr/IBM/WebSphere/myOptionFiles/log.txt  @ALL
```

On HP-UX and Solaris platforms:

```
install -options "/opt/IBM/WebSphere/silentFiles/myresponsefile.txt"
        -silent -log # !/opt/IBM/WebSphere/myOptionFiles/log.txt  @ALL
```

On Linux platforms:

```
install -options "/opt/ibm/WebSphere/silentFiles/myresponsefile.txt"
        -silent -log # !/opt/ibm/WebSphere/myOptionFiles/log.txt  @ALL
```

On Windows platforms:

```
install.exe -options "C:\IBM\WebSphere\silentFiles\myresponsefile.txt"
            -silent -log # !C:\IBM\WebSphere\silentFiles\log.txt  @ALL
```

Example responsefile.wpsclient.txt file

The response file provided with WebSphere Process Server, before modification, is similar to the following:

```
################################################################################
#
# Licensed Material - Property of IBM
# 5724-L01, 5655-N53, 5655-FLW
# (C) Copyright IBM Corporation 2006. All Rights Reserved.
# US Government Users Restricted Rights- Use, duplication or disclosure
# restricted by GSA ADP Schedule Contract with IBM Corp.
#
################################################################################

################################################################################
#
# InstallShield Options File
#
# Wizard name: Install
# Wizard source: setup.jar
#
# This file can be used to configure Install with the options specified below
# when the wizard is run with the "-options" command line option. Read each
# setting's documentation for information on how to change its value.
#
# A common use of an options file is to run the wizard in silent mode. This lets
# the options file author specify wizard settings without having to run the
# wizard in graphical mode. To use this options file for silent mode
# execution, use the following command line arguments when running the wizard:
#
#    -options "D:\installImage\WBI\responsefile.wpsclient.txt" -silent
#
################################################################################

###################################################################################
#
# License Acceptance
#
# Valid Options : true   Accepts the license. Will install the product.
#    false  Declines the license. Install will not occur.
#
# If no install occurs, this will be logged to a temporary log file in the user's
# temporary directory.
#
# By changing the silentInstallLicenseAcceptance.value in this response file to
# "true", you agree that you have reviewed and agree to the terms of the IBM
# International Program License Agreement accompanying this program, which is
# located at CD_ROOT\WBI\LICENSES.  If you do not agree to these terms, do not
# change the value or otherwise download, install, copy, access, or use the
# program and promptly return the program and proof of entitlement to the party
# from whom you acquired it to obtain a refund of the amount you paid.
#
#
```

```
-G licenseAccepted="false"

################################################################################
#
# IBM WebSphere Process Server, V6.0 Install Location
#
# The install location of the product. Specify a valid directory into which the
# product should be installed. If the directory contains spaces, enclose it in
# double-quotes as shown in the Windows example below. Note that spaces in the
# install location is only supported on Windows operating systems.
#
# Below is the list of default install locations for each supported operating
# system.  By default, in this response file, the Windows install location is
# used.  If you want to use the default install location for another operating
# system, uncomment the appropriate default install location entry (by removing '#') and
# then comment out (by adding '#') the Windows operating system entry below.
#
#
# AIX Default Install Location:
#
#    -P wbiProductBean.installLocation=/usr/IBM/WebSphere/ProcServer
#
#
# HP-UX, Solaris or Linux Default Install Location:
#
#    -P wbiProductBean.installLocation=/opt/IBM/WebSphere/ProcServer
#
#     Note: On Linux, the default directory has a lowercase "ibm".
#
# Windows Default Install Location:
#
# NOTE: The Windows operating system limits the length of a fully qualified path
#       to 256 characters. A long pathname for the installation root directory makes
#       it more likely that this limit will be exceeded when files are created during
#       normal product use. IBM recommends that you keep the pathname of the installation
#       root directory as short as possible.

-P wbiProductBean.installLocation="C:\Program Files\IBM\WebSphere\ProcServer"


################################################################################
#
# Use Existing IBM WebSphere Application Server V6.0
#
# If you intend to use an existing installation of WebSphere Application Server V6.0 or
# WebSphere Application Server Network Deployment, V6.0, uncomment the following line
# (by removing '#').
#
# -W wasdetectionpanelInstallWizardBean.optionSelected="1"
#
# You must then set the above WebSphere Process Server install location to the install root
# of the existing WAS installation.
#
# Note that the install will fail if WebSphere Process Server has already been installed
# in the existing WAS install location.
#


################################################################################
#
# Setup Type
#
# This value is required for the installation. Do not change this!
#

-W setuptypepanelInstallWizardBean.selectedSetupTypeId="Client"
```

Coexisting

Coexistence is the ability of two or more entities to function in the same system or network.

Coexistence does not imply the interoperability of these entities. As defined here, coexistence refers to a single environment in which a number of different WebSphere server configurations, which may involve different versions of the same product, coexist.

Coexistence support

Supporting coexistence means that you should be able to install, configure, run and manage coexisting products within your system.

Coexisting WebSphere server configurations can include the following:

- WebSphere Application Server for z/OS
- WebSphere Process Server for z/OS
- WebSphere Process Server configured as an ESB-only server.

When setting up your system for coexistence, you need to be aware of the following points:

- **Dependency on WebSphere Application Server for z/OS**

  Because WebSphere Process Server for z/OS installs on top of WebSphere Application Server for z/OS, it is a prerequisite that each server initially be configured for WebSphere Application Server for z/OS and be compliant with WebSphere Application Server for z/OS coexistence prior to installing and configuring the WebSphere Process Server for z/OS. For more information, see Coexistence support as described in the WebSphere Application Server for z/OS Information Center.
- **Version coexistence**

  You can have previous versions of the product on your system at the same time. Previous versions of the product can exist in the same LPAR or in separate LPARs.

  WebSphere Process Server for z/OS can not be installed on top of a WebSphere Business Integration Server Foundation for z/OS server.
- **Installing for coexistence**

  The following installation scenarios for coexistence are supported:

  - WebSphere Application Server for z/OS, WebSphere Process Server and WebSphere ESB coexisting where none of the products are currently installed.
  - Coexisting WebSphere Process Server for z/OS and WebSphere Application Server for z/OS when WebSphere Application Server for z/OS already exists
  - Coexisting WebSphere Process Server for z/OS and WebSphere Process Server for z/OS configured as an ESB-only server in an existing WebSphere Application Server for z/OS

Avoiding port conflicts

A major consideration in coexistence is the avoidance of port conflicts.

Review the port number settings, especially when you are planning to coexist.

You must prevent port conflicts from occurring when you have WebSphere Process Server for z/OS coexisting on the same server with another installation of itself or of WebSphere Enterprise Service Bus, WebSphere Business Integration Server Foundation, andWebSphere Application Server for z/OS.

Use this procedure to avoid port conflicts when adding a WebSphere Process Server for z/OS node to a WebSphere Process Server for z/OS deployment manager cell when a WebSphere Enterprise Service Bus, WebSphere Business Integration Server Foundation for z/OS, and WebSphere Application Server for z/OS node exists on the same system.

Perform the steps in this task as part of your strategy to avoid port conflicts for coexisting products.

1. Determine which products are coexisting. For example, if you plan to use a single z/OS system to contain more than one WebSphere Process Server for z/OS instance, you must configure port allocations of the second and later WebSphere Process Server for z/OS instances to avoid port conflicts with the first instance of WebSphere Process Server for z/OS already installed.
2. Configure port allocations To configure the port allocation, on the Administrative console of WebSphere Application Server for z/OS, navigate to Servers > Application Servers > YourProcessServerName >Ports.

Port number settings in versions of WebSphere Process Server, WebSphere Enterprise Service Bus, WebSphere Application Server, WebSphere Application Server Network Deployment, and WebSphere Business Integration Server Foundation

Prevent port conflicts from occurring when you want an installation of WebSphere Process Server to coexist with another installation of WebSphere Process Server, or with an installation of WebSphere Enterprise Service Bus, WebSphere Application Server, WebSphere Application Server Network Deployment, or WebSphere Business Integration Server Foundation. This topic provides reference information about identifying port numbers for these products.

Port numbers for WebSphere Process Server, WebSphere Enterprise Service Bus, and WebSphere Application Server and WebSphere Application Server Network Deployment, versions 6.0.0.x, 6.0.1.x, and 6.0.2.x

Table 15. Port definitions for WebSphere Process Server, WebSphere Enterprise Service Bus, and WebSphere Application Server and WebSphere Application Server Network Deployment, versions 6.0.0.x, 6.0.1.x, and 6.0.2.x

Port	WebSphere Process Server or WebSphere Enterprise Service Bus (with underlying WebSphere Application Server Network Deployment)	WebSphere Application Server	File
	Value	Value	
HTTP_TRANSPORT	9080	9080	serverindex.xml and virtualhosts.xml
HTTP Admin Console Port (HTTP_TRANSPORT_ADMIN)	9060	9060	serverindex.xml and virtualhosts.xml
HTTPS Transport Port (HTTPS_TRANSPORT)	9443	9443	serverindex.xml and virtualhosts.xml
HTTPS Admin Console Secure Port (HTTPS_TRANSPORT_ADMIN)	9043	9043	serverindex.xml and virtualhosts.xml
BOOTSTRAP_ADDRESS	9809	2809	serverindex.xml
SOAP_CONNECTOR_ADDRESS	8879	8880	serverindex.xml

Table 15. Port definitions for WebSphere Process Server, WebSphere Enterprise Service Bus, and WebSphere Application Server and WebSphere Application Server Network Deployment, versions 6.0.0.x, 6.0.1.x, and 6.0.2.x (continued)

Port	WebSphere Process Server or WebSphere Enterprise Service Bus (with underlying WebSphere Application Server Network Deployment)	WebSphere Application Server	File
	Value	**Value**	
SAS_SSL_SERVERAUTH_LISTENER_ ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	9401	9401	serverindex.xml
CSIV2_SSL_SERVERAUTH_ LISTENER_ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	9403	9403	serverindex.xml
CSIV2_SSL_MULTIAUTH_LISTENER_ ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	9402	9402	serverindex.xml
ORB_LISTENER_ADDRESS	9100	9100	serverindex.xml
DCS_UNICAST_ADDRESS	9352	9353	serverindex.xml
SIB_ENDPOINT_ADDRESS	7276	7276	serverindex.xml
SIB_ENDPOINT_SECURE_ADDRESS	7286	7286	serverindex.xml
SIB_MQ_ENDPOINT_ADDRESS	5558	5558	serverindex.xml
SIB_MQ_ENDPOINT_SECURE_ ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	5578	5578	serverindex.xml
Internal JMS Server (JMSSERVER_SECURITY_PORT)	Not applicable	5557	serverindex.xml
DRS_CLIENT_ADDRESS **Deprecation:** This port is deprecated and is no longer used in the current version of WebSphere Application Server.	7989	7873	serverindex.xml

Table 15. Port definitions for WebSphere Process Server, WebSphere Enterprise Service Bus, and WebSphere Application Server and WebSphere Application Server Network Deployment, versions 6.0.0.x, 6.0.1.x, and 6.0.2.x (continued)

Port	WebSphere Process Server or WebSphere Enterprise Service Bus (with underlying WebSphere Application Server Network Deployment)	WebSphere Application Server	File
	Value	Value	
IBM HTTP Server Port	Not applicable	80	virtualhosts.xml, plugin-cfg.xml, and *IHSinstall_root*/conf/httpd.conf
IBM HTTPS Server Admin Port	Not applicable	8008	*IHSinstall_root*/conf/admin.conf
CELL_DISCOVERY_ADDRESS	7277	Not applicable	serverindex.xml
CELL_MULTICAST_DISCOVERY_ ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	7272	Not applicable	serverindex.xml
NODE_MULTICAST_IPV6_ DISCOVERY_ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	5001	5001	serverindex.xml

When you federate a node into a deployment manager cell, the deployment manager instantiates the nodeagent server process on the process server node. The nodeagent server uses these port assignments by default:

Table 16. Port definitions for the WebSphere Process Server nodeagent server process

Port	Value	File
BOOTSTRAP_ADDRESS	2089	serverindex.xml
ORB_LISTENER_ADDRESS	9900	serverindex.xml
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS	9901	serverindex.xml
CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS	9202	serverindex.xml
CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS	9201	serverindex.xml
NODE_DISCOVERY_ADDRESS	7272	serverindex.xml
NODE_MULTICAST_DISCOVERY_ADDRESS	5000	serverindex.xml
NODE_IPV6_MULTICAST_DISCOVERY_ADDRESS	5001	serverindex.xml
DCS_UNICAST_ADDRESS	9353	serverindex.xml
DRS_CLIENT_ADDRESS	7888	serverindex.xml
SOAP_CONNECTOR_ADDRESS	8878	serverindex.xml

Port numbers for WebSphere Application Server and WebSphere Application Server Network Deployment, version 5.1.x, and WebSphere Business Integration Server Foundation, version 5.1.x

WebSphere Business Integration Server Foundation uses the same ports as the product it extends. Therefore, if it extends WebSphere Application Server Network Deployment, use the values under that column in Table 17. If it extends WebSphere Application Server, use the values under that column.

Table 17. Port definitions for WebSphere Application Server and WebSphere Application Server Network Deployment, version 5.1.x, and WebSphere Business Integration Server Foundation, version 5.1.x

Port	**WebSphere Application Server Network Deployment**	**WebSphere Application Server**	**File**
	Value	**Value**	
HTTP_TRANSPORT	Not applicable	9080	serverindex.xml and virtualhosts.xml
HTTPS Transport Port (HTTPS_TRANSPORT)	Not applicable	9443	serverindex.xml and virtualhosts.xml
HTTP Admin Console Port (HTTP_TRANSPORT_ADMIN)	9090	9090	serverindex.xml and virtualhosts.xml
HTTPS Admin Console Secure Port (HTTPS_TRANSPORT_ADMIN)	9043	9043	serverindex.xml and virtualhosts.xml
Internal JMS Server (JMSSERVER_SECURITY_PORT)	Not applicable	5557	server.xml
JMSSERVER_QUEUED_ADDRESS	Not applicable	5558	serverindex.xml
JMSSERVER_DIRECT_ADDRESS	Not applicable	5559	serverindex.xml
BOOTSTRAP_ADDRESS	9809	2809	serverindex.xml
SOAP_CONNECTOR_ADDRESS	8879	8880	serverindex.xml
DRS_CLIENT_ADDRESS	7989	7873	serverindex.xml
SAS_SSL_SERVERAUTH_LISTENER_ ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	9401	0	serverindex.xml
CSIV2_SSL_SERVERAUTH_ LISTENER_ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	9403	0	serverindex.xml
CSIV2_SSL_MULTIAUTH_LISTENER_ ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	9402	0	serverindex.xml

Table 17. Port definitions for WebSphere Application Server and WebSphere Application Server Network Deployment, version 5.1.x, and WebSphere Business Integration Server Foundation, version 5.1.x (continued)

Port	**WebSphere Application Server Network Deployment**	**WebSphere Application Server**	**File**
	Value	**Value**	
IBM HTTP Server Port	Not applicable	80	virtualhosts.xml, plugin-cfg.xml, and *IHSinstall_root*`/conf/httpd.conf`
IBM HTTPS Server Admin Port	Not applicable	8008	*IHSinstall_root*`/conf/admin.conf`
CELL_DISCOVERY_ADDRESS	7277	Not applicable	serverindex.xml
ORB_LISTENER_ADDRESS	9100	9100	serverindex.xml
CELL_MULTICAST_DISCOVERY_ ADDRESS A character space was added to this entry to enable it to fit in the table cell. The actual entry does not include a character space.	7272	Not applicable	serverindex.xml

When you federate a WebSphere Application Server or WebSphere Application Server Network Deployment, version 5.1.x application server node into a deployment manager cell, the deployment manager instantiates the nodeagent server process on the application server node. The nodeagent server uses these port assignments by default:

Table 18. Port definitions for the nodeagent server process of WebSphere Application Server and WebSphere Application Server Network Deployment, version 5.1.x, and WebSphere Business Integration Server Foundation, version 5.1.x

Port	**Value**	**File**
BOOTSTRAP_ADDRESS	2089	serverindex.xml
ORB_LISTENER_ADDRESS	9900	serverindex.xml
SAS_SSL_SERVERAUTH_LISTENER_ADDRESS	9901	serverindex.xml
CSIV2_SSL_MUTUALAUTH_LISTENER_ADDRESS	9101	serverindex.xml
CSIV2_SSL_SERVERAUTH_LISTENER_ADDRESS	9201	serverindex.xml
NODE_DISCOVERY_ADDRESS	7272	serverindex.xml
NODE_MULTICAST_DISCOVERY_ADDRESS	5000	serverindex.xml
DRS_CLIENT_ADDRESS	7888	serverindex.xml
SOAP_CONNECTOR_ADDRESS	8878	serverindex.xml

When a messaging engine is created on a server, the port assignments are as defined in Table 19 on page 110. For a stand-alone server profile, the action of creating the messaging engine is a part of installation. For a network deployment environment, the messaging engine is created as a part of the administration actions to create a bus member or add a messaging engine to a bus member.

Table 19. Port definitions for the WebSphere Process Server jmsserver process

Port	Value	File
JMSSERVER_DIRECT_ADDRESS	5559	serverindex.xml
JMSSERVER_QUEUED_ADDRESS	5558	serverindex.xml
SOAP_CONNECTOR_ADDRESS	8876	serverindex.xml
JMSSERVER_SECURITY_PORT	5557	server.xml

Installing WebSphere Process Server Client to coexist with existing installations of various WebSphere products

Use this procedure to install a WebSphere Process Server Client on a Linux, UNIX, or Windows machine for use in a WebSphere Process Server for z/OS environment with an existing installation of WebSphere Process Server, WebSphere Process Server Client, WebSphere Enterprise Service Bus, or a supported version of WebSphere Application Server or WebSphere Application Server Network Deployment. This procedure uses the Installation wizard graphical user interface (GUI) from the WebSphere Process Server Client CD.

Before installing the WebSphere Process Server Client on a Linux, UNIX, or Windows machine, ensure that:

- You have access to the WebSphere Process Server for Multiplatforms information center.

  Because you are installing the WebSphere Process Server Client on a Linux, UNIX, or Windows machine, you need to review content in this information center that pertains to installing on a non-z/OS machine.
- You have decided whether to install WebSphere Application Server Network Deployment as part of your WebSphere Process Server Client installation or to instead use an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment, version 6.0.x.x.
- Your system meets all hardware requirements and that you have installed all required software corequisites and prerequisites. See Hardware and software requirements for more information.
- You have prepared your operating system on the Client machine for installation. See Preparing the operating system for installation in the WebSphere Process Server for Multiplatforms 6.0.2 information center.
- You are installing the Client as the root user on a Linux or UNIX system, or as a member of the Administrator group on a Windows system.
- You are installing the Client from one of the following CD-ROMs:
  - WebSphere Process Server 6.0.2 Client for Windows
  - WebSphere Process Server 6.0.2 Client for AIX
  - WebSphere Process Server 6.0.2 Client for Solaris on SPARC
  - WebSphere Process Server 6.0.2 Client for Solaris on x86 64-bit
  - WebSphere Process Server 6.0.2 Client for HP-UX
  - WebSphere Process Server 6.0.2 Client for HP-UX on Itanium 64-bit
  - WebSphere Process Server 6.0.2 Client for Linux on x86 32-bit
  - WebSphere Process Server 6.0.2 Client for Linux on x86 64-bit
  - WebSphere Process Server 6.0.2 Client for Linux on iSeries and pSeries
  - WebSphere Process Server 6.0.2 Client for Linux on zSeries 31-bit
  - WebSphere Process Server 6.0.2 Client for Linux on zSeries 64-bit

This procedure assumes you have an installation of WebSphere Process Server, WebSphere Enterprise Service Bus, WebSphere Application Server, version 6.0.x.x, or WebSphere Application Server for z/OS 6.0.x.x on your system. You do not have to have existing profiles. It also assumes you want to install the Client interactively from a CD-ROM.

1. Log on as the root user on a Linux or UNIX system, or as a member of the Administrator group on a Windows system.
2. Place the CD-ROM into the CD-ROM drive of the machine you are installing on and do the following:

   Mount the drive if necessary, as described in Mounting CD-ROMs on Linux and UNIX operating systems in the WebSphere Process Server for Multiplatforms 6.0.2 information center.
3. Start the Client Installation wizard by navigating to the WBI directory and doing the following:
   - **On Windows platforms**: Run the install.exe file
   - **On Linux and UNIX platforms**: Run the command **install**.
4. From the Welcome panel of the installer, select **Next**.
5. In the Software License Agreement panel, review the IBM and non-IBM licensing terms, select **I accept both the IBM and the non-IBM terms**, and select **Next**.

   The Installation wizard checks for a supported operating system with prerequisite patches. At the end of the process, the Checking prerequisites panel indicates whether your system passed the check. This procedure assumes that your system passed.

   If your system did not pass, cancel the installation, make the required changes, and restart the installation.
6. In the Checking prerequisites panel, select **Next**

   The Installation wizard checks for existing installations of WebSphere Process Server, WebSphere Process Server Client, WebSphere Enterprise Service Bus, WebSphere Application Server, and WebSphere Application Server Network Deployment. This procedure assumes you do not have an existing WebSphere Process Server, Client, or WebSphere Enterprise Service Bus installation, but that you might have an existing WebSphere Application Server or WebSphere Application Server Network Deployment installation.

   If you *do not* have an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment, the Installation location panel is displayed. Proceed to Step 8.

   If you *do* have an existing installation of one of these products, an Existing installation detected panel is displayed. Proceed to step 7.

   Important: The installer will also detect unregistered instances of WebSphere Application Server or WebSphere Application Server Network Deployment if they have entries in the `.WASRegistry` file. This file is located in the `$USER_HOME` directory of the installation. Using an unregistered installation of one of these products with your WebSphere Process Server installation is neither recommended nor supported.
7. In the Existing installation detected panel, do one of the following:
   - If you want to install a new copy of WebSphere Application Server Network Deployment, select **Install a new copy of WebSphere Application Server Network Deployment, Version 6.0** and select **Next**. A warning panel outlines considerations you must make when installing the product on a

system with an existing installation. Select **Next** to close the warning panel. The Installation location panel is displayed. Proceed to step 8.

- If you want to install the Client over an existing installation of WebSphere Application Server or WebSphere Application Server Network Deployment, select **Use an existing installation of WebSphere Application Server Network Deployment, Version 6.0** or **Use an existing installation of WebSphere Application Server, Version 6.0** and select **Next**. (If you have multiple installations, be sure to select the one you want to use from the drop-down list.) A warning panel alerts you to stop servers before continuing if the application you chose has running servers. Select Next to close the warning panel. The Installation type panel is displayed. Proceed to step 9.

8. In the Installation location panel, accept the default installation root directory, or specify a different directory, and select **Next**.

   See the topic on Default installation paths in the WebSphere Process Server for Multiplatforms 602 information center for the default installation directories.

   Important:

   - You must provide a value for the installation root directory to continue.
   - Do not use non-ASCII special characters in directory names – they are not supported.
   - **On Linux and UNIX platforms**: Do not use symbolic links as the installation root directory – they are not supported. Also, do not use spaces in the directory path.
   - **On Linux platforms**: The installation root directory path must be no longer than 256 characters to successfully install the products.

9. The Installation type panel displays the **Client installation** as the only type of installation allowed. Click **Next**.

10. In the Installation summary panel, review the components that will be installed, the amount of space they will consume, and where they will be located on the system, and select **Next** to install or **Back** to change your specifications.

    WebSphere Application Server Network Deployment and WebSphere Process Server are the only products installed.

    The Installation wizard creates the uninstaller program and shows a progress panel to indicate that components are being installed.

    If you elected to use an existing version of WebSphere Application Server or WebSphere Application Server Network Deployment, the Installation wizard examines it and does one of the following:

    - Does nothing if the installation is at the correct service level.
    - If the installation is at an earlier service level, applies the necessary fixes to bring the installation up to the appropriate level and also applies any necessary interim fixes.

    At the end of the installation, the Installation complete panel is displayed.

Attention: If errors are detected during installation, other panels might be displayed in place of the Installation complete panel. Examples include the following:

- Installation is complete with errors panel, which indicates that the installation completed but errors were generated.
- Installation failed panel, which indicates that the installation failed completely.

Each of these panels identifies the log file to reference in order to troubleshoot the problems. See the following topics in the WebSphere Process Server for Multiplatforms 6.0.2 information center for tips on troubleshooting your installation:

- Log files, for descriptions of relevant log files.
- Error messages: installation and profile creation and augmentation, for descriptions of error messages.

the WebSphere Process Server for Multiplatform 6.0.2 information center for descriptions of relevant log files listed in the *Log files* topic, error messages in *Error messages: installation and profile creation and augmentation* topic, and refer to *Troubleshooting installation* topic for tips on troubleshooting your installation.

11. Click **Finish** to close the Installation wizard.

Configuring the software

After you have installed WebSphere Process Server for z/OS and have run the installation script (zSMPInstall.sh), you must create the desired configuration by augmenting an existing WebSphere Application Server for z/OS default profile.

In WebSphere Application Server for z/OS, all runtime environments are created with the ISPF Customization Dialog using a profile name of *default*.

Note: The manageprofiles command, and the -profile option on other administrative commands, are not used with WebSphere Application Server for z/OS.

The WebSphere Application Server for z/OS server configuration is kept in a directory structure under a *profiles* directory. So, for the base stand-alone application server, in the configuration root, there will be a *profiles* directory, and in that directory, there will be a “default” directory that contains all of the directories and files specific to your server. The profiles> default directory structure is the same for a deployment manager / empty managed node.

Important: This is important because you will frequently navigate to this profiles> default directory to find server and FFDC logs.

When you create a WebSphere Process Server for z/OS configuration, you do so by augmenting the WebSphere Application Server for z/OS default profile with WebSphere Process Server configuration data. You accomplish this by running the configuration script with a response file that contains parameters specific to the configuration that you are creating.

The Service Integration Bus (SIBus) component of the WebSphere version 6 runtime is based entirely on default platform messaging, which is a Java implementation of JMS contained completely within WebSphere Application Server.

Use the subtopics in this section to find more detailed information on tasks you might have to perform after you install WebSphere Enterprise. Pointers to conceptual topics supporting the tasks are provided as well.

You can also review topics on creating servers within a network deployed environment in the *Administering WebSphere Process Server for z/OS, version 6.0.2* PDF or in the WebSphere Process Server for z/OS, version 6.0.2, online information center at http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp by navigating to **Administering WebSphere Process Server**.

Considerations for configuring queue resources for the business process container using WebSphere MQ

WebSphere Process Server for z/OS product configurations include features that utilize a Java Message Service (JMS) provider.

If you plan on using WebSphere MQ as your JMS provider for the business process container, you need to consider the following:

- You must have WebSphere MQ installed prior to creating the queues and queue manager.
- On z/OS it is likely that the business process container will use a preexisting queue manager and listener.
- Shell script commands for creating the queue manager and listeners are not used on z/OS.
- You cannot create the queues and queue manager automatically by using the installation wizard in the administrative console *or by running bpeconfig.jacl*

  The business process container installation wizard in the administrative console and bpeconfig.jacl configure *WebSphere resources only*.

  If you choose to configure the business process container by using either the installation wizard or bpeconfig.jacl, you will still need to run the corresponding manual step to create the WebSphere MQ queues.
- Running bpeconfig.jacl or running createQueues.sh will result in a file named **tmp_crt_ques.mqs**.

  The file **tmp_crt_ques.mqs** contains sample queue definitions.

  These sample queue definitions are based on parameters that you set before running bpeconfig.jacl or createQueues.sh.

  After running either bpeconfig.jacl or createQueues.sh, you should give the **tmp_crt_ques.mqs** file to your WebSphere MQ administrator. The contents of this file can be used by WebSphere MQ z/OS commands in JCL scripts to configure the WebSphere MQ resources.

For information on how to run bpeconfig.jacl, see information on creating the queue manager and queues in the *Business Process Choreographer* documentation.

For information on how to run createQueues.sh, see information on creating the queue manager and queues in the *Business Process Choreographer* documentation.

Database specifications

WebSphere Process Server uses a number of databases to hold, store and track information.

During normal operation of WebSphere Process Server a large amount of data is being accessed, moved and supplemented. This data is housed in a number of databases that you create prior to running the configuration script.

You can choose one of two ways to configure these databases, as follows:

- Configure the databases programmatically

  In the response file set **dbDelayConfig=false** to allow the product configuration script to programmatically configure the databases. When you specify dbDelayConfig=false the product configuration process will connect to the database and create the definitions for the data sources and database tables automatically.
- Configure the databases manually

  In the response file set **dbDelayConfig=true** to prevent the product configuration script from programmatically configuring the databases.

  When you specify **dbDelayConfig=true**, you will need to run shell scripts and ANT tasks separately to create the data source and database table definitions manually. You can do this after you have run the configuration script successfully. See *Configuring the databases manually* for details.

WebSphere Process Server uses a JDBC data source, and corresponding relational database repositories (in Cloudscape or DB2 for z/OS) for the following features:

- AppScheduler, Recovery, Mediation, Customization, and Relationship
- Business Process Choreographer and Human Task Manager
- Service Integration Bus (SIBus)
  - Messaging engines (MEs) for Business Process Choreographer (BPC), Common Event Infrastructure (CEI), Service Component Architecture (SCA) Application and System Buses.
- Common Event Infrastructure (CEI)
- WebSphere ESB (Enterprise Service Bus) for mediation event logging.

For a stand-alone configuration using a DB2 for z/OS database, or for a deployment manager configuration, you must create the databases and storage groups for the following features prior to running the WebSphere Process Server for z/OS configuration script:

- AppScheduler, Recovery, Mediation, Customization, and Relationship
- Business Process Choreographer and Human Task Manager
- Common Event Infrastructure (CEI)
- WebSphere ESB (Enterprise Service Bus) for mediation event logging.

When you run the WebSphere Process Server for z/OS configuration script, make sure that the database and storage group names in the response file match the names that you set when you created the database and storage groups. For information on database and storage groups, see Creating databases and storage groups.

As part of the WebSphere Process Server configuration, which is driven by the configuration script(s), data source definitions are created in your server, and SQL for database tables and tablespaces are generated. The SQL can be used as sample DDL (Data Definition Language) statements by your data base administrator (DBA).

The manner by which you configure databases (as part of product configuration or by running shell scripts ANT tasks separately at a later time), depends on your installation and configuration scenario. For example, if you are installing and configuring WebSphere Process Server into a stand-alone application server using the Cloudscape database, you can run the product configuration script using the default values in the response file to configure WebSphere Process Server and its various databases. If your installation and configuration scenario is for a stand-alone server that will use DB2 for z/OS as its database, then you will need to meet some additional configuration requirements such as configuring the DB2 for z/OS subsystem to support the WebSphere Process Server configuration.

A WebSphere Process Server network deployment configuration using DB2 for z/OS is significantly more complex in terms of WebSphere Application Server profiles, resources, containers, applications and other infrastructure. However, the network deployment configuration steps for configuring the DB2 for z/OS is fairly similar to the stand-alone configuration in terms of the database setup.

Table 20 lists the WebSphere Process Server features that require a database and the default names of the databases associated with these features.

Note: You can edit the default values assigned to the database names if you choose.

Table 20. WebSphere Process Server components and the datasbases they require.

Feature	Database (default name)
Business Process Choreographer	BPEDB
Common Event Infrastructure	CEIDB **Note:** The configuration process defines two databases named EVENT and EVENTCAT respectively. These database names should also be set when you create the database and storage groups. These databases will hold the definitions for the Common Event Infrastructure database tables and data source.
Relationships	WPRCSDB (the common database)
Mediation	WPRCSDB (the common database)
Recovery	WPRCSDB (the common database)
Application scheduler	WPRCSDB (the common database)
Selectors/Business rules	WPRCSDB (the common database)/Repository DB (Cloudscape)
SIBus	User created
ESB	ESBDB

Details of the supported databases including versions, are provided in the WebSphere Process Server system requirements.. Each database is represented by a parameter *dbType* which is a character string. The *dbType* parameter is used in file naming and file paths. The values of *dbType* for the supported databases are shown in Table 21 on page 117.

Table 21. The supported databases and their associated dbType values.

Database	**dbType**
Cloudscape	Cloudscape
DB2 Universal OS/390 V7.1	DB2zOSV7
DB2 Universal OS/390 V8.1	DB2zOSV8
DB2 CLI	DB2

A second parameter used in file path and file naming conventions is feature, which indicates which of the various databases (see Table 20 on page 116) is under consideration. Table 22 lists the databases and the associated feature parameter.

Table 22. Databases and their associated feature name.

Database	**Feature**
Business Process Choreographer	ProcessChoreographer
Common event infrastructure	CEI
Common database	CommonDB
ESB	EsbLoggerMediation

When you install WebSphere Process Server, database scripts are created in the following locations:

Note: The *configuration_root/app_server_root* variable represents the WebSphere Application Server for z/OS product configuration root and application server root combined. This is sometimes referred to as the WAS_HOME directory. For example, the WebSphere Application Server for z/OS product configuration root for a stand-alone server would be */WebSphere/V6R0M0/AppServer*, while the WebSphere Application Server for z/OS product configuration root for a deployment manager would be */WebSphere/V6R0M0/DeploymentManager*

```
configuration_root/app_server_root/dbscripts/feature/dbType
```

giving a unique location for the SQL scripts for each database. This holds the scripts for both the common database and the Business Process Choreographer database. The Common Event Infrastructure database and the ESB Logger Mediation database use the locations:

```
configuration_root/app_server_root/profiles/profileName/event/dbscripts/dbType
```

and

```
configuration_root/app_server_root/util/EsbLoggerMediation/dbType
```

respectively.

You can delay configuring the database by setting the response file parameter: dbDelayConfig to *true*. If you set this value to *true* the product configuration script does not invoke the tasks to configure the databases, and you must use the provided scripts to configure the databases manually.

Important: Business Process Choreographer and Common Event Infrastructure databases must be configured manually in a network deployment configuration, regardless of how you set the value for dbDelayConfig.

If you choose to defer creation of the databases until after product configuration, then default scripts, which include database parameters entered in the response file, are created in the following location:

*/configuration_root/app_server_root/*profiles/*profileName*/dbscripts/*feature*/*dbType*/*dbName*

For example, *WebSphere/V6R0/AppServer*/profiles/*profileName*/dbscripts/*feature*/*dbType*/*dbName*

Note: If you are creating the databases as part of the product configuration any mistakes in the database parameters that your provide will yield errors in the profile being augmented. However, if you are delaying the database creation, the configuration will be created without errors, but the generated database scripts will contain errors and therefore must be edited before the databases can be created.

The scripts are ready to execute, but you can edit them to include any specific requirements. If you try to start WebSphere Process Server before creating the databases you will receive an error message.

Since the path of the scripts contains the type of database (*dbType*) there is no need for the database type to be included in the script name. The script naming convention is shown in Table 23.

Table 23. Database script naming convention

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

The creation of JDBC providers is still performed during product configuration, based upon the database parameters that you provide in the response file. However, if a suitable JDBC provider is located on the server, the existing JDBC provider is used and the creation of a new one is omitted.

Common database specifications

A common database is optionally created when you configure WebSphere Process Server. This database acts as a repository by various components. If you prefer, you can defer the creation of this database until after product configuration.

The common database (CommonDB) is used by the following WebSphere Process Server components:

- Recovery
- Relationship service
- Mediation
- Application Scheduler
- Customization (selector and business rule group)

You can either create the common database when you run the product configuration script, or after you have configured WebSphere Process Server by running the SQL that is generated as part of the configuration process. The SQL includes the database information that you provided in the response file.

Note: You can use the **ws_ant.sh** script to run the generated SQL.

For a stand-alone or deployment manager configuration using a DB2 for z/OS database, you must create the common database and storage groups *prior to running the WebSphere Process Server for z/OS configuration script*.

When you run the WebSphere Process Server for z/OS configuration script, make sure that the common database and storage group names in the response file match the names that you set when you created the database and storage groups. For information on database and storage groups see the topic **Creating databases and storage groups**.

Supported database types

The following databases are supported for use as the common database:

- Cloudscape(TM) – Cloudscape is the default database type for a stand-alone profile. Cloudscape is disabled for network deployment environments. The selector and business rule group components do not use the Common database when Cloudscape is the database that is being used.
- DB2 Universal Database™ –
  - DB2 Universal Database for OS/390(R) V7.1
  - DB2 Universal Database for OS/390 V8.1

    DB2 Universal Database version 8.1 is the default database type for network deployment environments.

Database Management Service (DBMS) instances

There is one DBMS instance per cell.

Database actions invoked by the product configuration script

After the SMP/E installation of WebSphere Process Server for z/OS, the SQL is located in *smpe_root*/dbscripts/CommonDB/*dbType*.

Running the WebSphere Process Server for z/OS installation script, creates symbolic links from the *smpe_root* to the WebSphere Application Server for z/OS */configuration_root/app_server_root*.

For example, for a stand-alone configuration the symbolic link would be to *WebSphere/V6R0/AppServer*/dbscripts/CommonDB/*dbType*.

For a deployment manager configuration the symbolic link would be to *WebSphere/V6R0/DeploymentManager*/dbscripts/CommonDB/*dbType*.

Stand-alone profile and deployment manager profile augmentation

The Common database is optionally created when you run the product configuration script.

If you defer creation of the database until after you have run the configuration script, the customized scripts will be found in:

configuration_root/*app_server_root*/profiles/*profilename*/dbscripts/CommonDB/*dbType*/*dbName*.

For example, a stand-alone profile would be found in /*WebSphere*/*V6R0*/*AppServer*/profiles/*profilename*/dbscripts/CommonDB/*dbType*/*dbName*. While a deployment manager profile would be found in */WebSphere/V6R0/DeploymentManager/*profiles/*profilename*/dbscripts/CommonDB/*dbType*/*dbName*

These scripts contain symbolic placeholders for which response file values are substituted by the configuration script.

The scripts perform the following steps:

- Create new tables and table spaces in the common database if asked (valid only for a local database), depending on your choices in the response file.

  Note: These scripts create only the table and table spaces. The creation of the common database is a prerequisite for running these scripts, regardless of whether you run the scripts automatically (when you run the configuration script), or manually, after you have run the configuration script. See Creating databases and storage groups for more information.

  Note: Even though you can defer creation of the database until after you have run the configuration script, it is necessary to enter valid information in the Database parameters within the response file. This information is used to create the data source for the WebSphere Process Server.
- If **delayConfig=false** was specified in the response file, each component runs the "createTable" SQL. For example,

  ```
  dbscripts/CommonDB/DB2zOSV8/createTable_customization.sql
  dbscripts/CommonDB/DB2zOSV8/createTable_lockmanager.sql
  dbscripts/CommonDB/DB2zOSV8/createTable_mediation.sql
  dbscripts/CommonDB/DB2zOSV8/createTable_AppScheduler.sql
  dbscripts/CommonDB/DB2zOSV8/createTable_CommonDB.sql
  dbscripts/CommonDB/DB2zOSV8/createTable_Recovery.sql
  dbscripts/CommonDB/DB2zOSV8/createTable_Relationship.sql
  ```

SQL scripts

SQL scripts allow you to configure your database after profile augmentation (after running the configuration script). SQL scripts for each common database client are located at the following locations. The Common database is the central database in a network deployment environment. Tables are created with a deployment manager default profile and so there are no SQL scripts executed as part of managed node creation.

If you defer creation of the database until after you have run the configuration script, the customized scripts will be found in:

configuration_root/*app_server_root*/profiles/*profilename*/dbscripts/CommonDB/*dbType*/*dbName*. For example, */WebSphere/V6R0/DeploymentManager/*profiles/*profilename*/dbscripts/CommonDB/*dbType*/*dbName*

The names of the SQL scripts have been made database-independent since the database is implicit in the file path. The naming convention is:

- For a component-specific script: createTable_*componentName*.sql. For example createTable_Recovery.sql.

Restrictions

There are several restrictions on the database commands that are available to the user during profile augmentation.

Create new database is disabled for the following database types:

- DB2 Universal Database for OS/390 V7.1
- DB2 Universal Database for OS/390 V8.1

This means that the **dbCreateNew** property in the response file must have a value of "false".

Databases and storage groups must be created in accordance with instructions provided in Creating databases and storage groups.

Tables

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

Table 24. Tables created by WebSphere Process Server components

Component	**Table names**	**Scripts**
Recovery	FAILEDEVENTS FAILEDEVENTBOTYPES FAILEDEVENTMESSAGE	`createTable_Recovery.sql`
Mediation	MEDIATION_TICKETS	`createTable_mediation.sql`
Relationship	Dynamic table, be created at runtime	`createTable_Relationship.sql` (create tablespace, for DB2 z/OS only)
AppScheduler	WSCH_LMGR WSCH_LMPR WSCH_TASK WSCH_TREG	`createTable_AppScheduler.sql`
Customization (Selection/BR)	BYTESTORE BYTESTOREOVERFLOW APPTIMESTAMP	`createTable_customization.sql`
CommonDB	Schema Version Information	`createTable_commonDb.sql`
Lock manager	PERSISTENTLOCK	`createTable_lockmanager.sql`

All the above SQL scripts are executed by the commonDBUtility.ant from each component script, such as configRecovery > commonDBUtility > execute createTable_Recovery.sql.

When the **delayConfig=true** in the response file, the SQL files are created, but are not run. When this is the case, you will need to run the SQL manually after the configuration. For more information, see Configuring the database manually.

Exported scripts

Database scripts are exported to the following folder:

configuration_root/*app_server_root*/profiles/*profilename*/dbscripts/CommonDB/*dbType*/*dbName*

Scripts are created regardless of the value (true or false) that you assign to the **delayConfig** property in the response file. The scripts contain only basic creation statements for database tables and table spaces. The database administrator can use database tool of to run these scripts.

User ID privileges

Generally the user credentials that you provide in the response file should be able to create table spaces, tables, indexes, and stored procedures. For Create new database option, the user identity should have the necessary privileges to create a new database.

Common Event Infrastructure database specifications

In order to monitor WebSphere Process Server events, it is necessary to capture events as they occur in a database. The Common Event Infrastructure (CEI) database is used for this purpose.

The WebSphere Process Server Common Event Infrastructure database is created for a stand-alone configuration and for each instance of Common Event Infrastructure server in a network deployment environment. The Common Event Infrastructure database is an internal device and you do not interact directly with it. All interactions with the Common Event Infrastructure database must be performed using the supported Common Event Infrastructure APIs. See: Configuring the event database for more information.

For a stand-alone configuration using a DB2 for z/OS database, or for a managed node or deployment manager configuration, you must create the Common Event Infrastructure database and storage groups *prior to running the WebSphere Process Server for z/OS configuration script*.

When you run the WebSphere Process Server for z/OS configuration script, make sure that the Common Event Infrastructure database and storage group names in the response file match the names that you set when you created the database and storage groups. For information on database and storage groups, see Creating databases and storage groups.

Supported database types

The following databases are supported for use as the Common Event Infrastructure database:

- Cloudscape V5.1
- DB2 Universal Database for OS/390 V7.1
- DB2 Universal Database for OS/390 V8.1

Database Management Service (DBMS) instances

The installation script creates the Common Event Infrastructure database automatically only for a stand-alone server configuration, so there is one instance of the DBMS for each server. All other instances of the Common Event Infrastructure database need to be created manually.

Database actions invoked by the product configuration script

Configuration of the Common Event Infrastructure database is done by the profile template ANT script `configCei.ant` invoking the Common Event Infrastructure API with a response file containing all the necessary properties for the desired configuration.

SQL scripts

Database configuration is accomplished by creating a response file invoking the Common Event Infrastructure API with it as a parameter. As Common Event Infrastructure processes this response file and creates scripts, they are placed under the following location: `${profilePath}/event/dbscripts/${dbtype}`

JDBC provider

The scripts that are used for creating the Common Event Infrastructure database are stored in the following location: *configuration_root*/*app_server_root*/profiles/*profilename*/event/dsscripts/${dbtype}

The JNDI names of the components are:

- jndiName="jdbc/cei"
- jndiName="jdbc/eventcatalog"

Tables

Many tables are created. Look under the generated scripts to see which tables are generated for the given database product.

Exported scripts

Shell scripts are created in the following directory to be used to run the generated SQL scripts.

configuration_root/*app_server_root*/profiles/*profilename*/event/dbscripts/*dbtype*

User ID privileges

Generally the user credentials that you provide in the response file should be able to create tablespaces, tables, indexes, and stored procedures.

Additional information

More information about configuring the Common Event Infrastructure can be found in the topic *Configuring the event database* in the *WebSphere Process Server for z/OS, version 6.0.2 Common Event Infrastructure* PDF. Or view the topic in the WebSphere Process Server for z/OS, version 6.0.2 online information center at http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp by navigating to **Administering WebSphere Process Server > Administering the Common Event Infrastructure > Post-installation configuration > Configuring the event database**.

Business Process Choreographer database specifications

The Business Process Choreographer requires a database. The Business Process Choreographer database (BPEDB) is used for this purpose.

The Business Process Choreographer database is only used by Business Process Choreographer.

Supported database types

The following databases are supported for use as the Business Process Choreographer database:

- Cloudscape V5,1
  - Cloudscape JDBC Provider (XA)

    Note: Cloudscape is not supported as the Business Process Choreographer database in a network deployment environment. If you are creating a network deployment configuration for WebSphere Process Server for z/OS, you need to create and configure DB2 for z/OS as the database.
- DB2 Universal Database for OS/390 V7 and V8
  - DB2 Legacy CLI-based Type 2 JDBC Driver (XA)
  - DB2 Universal JDBC Driver Provider (XA)

For a stand-alone configuration using a DB2 for z/OS database, or for a managed node or deployment manager configuration, you must create the Business Process Choreographer database and storage groups prior to running the WebSphere Process Server for z/OS configuration script.

When you run the WebSphere Process Server for z/OS configuration script, make sure that the Business Process Choreographer database and storage group names in the response file match the names that you set when you created the database and storage groups. For information on database and storage groups, see Creating databases and storage groups.

Database Management Service (DBMS) instances

Each deployment target (either server or cluster) can be configured to support Business Process Choreographer applications. Each Business Process Choreographer deployment target must have its own database. Databases cannot be shared between deployment targets.

Database actions invoked by the product configuration script

When you run the WebSphere Process Server configuration script for a stand-alone server, you can set parameters in either of the response files (standAloneProfile.rsp for Cloudscape or standAloneProfileDB2.rsp for DB2 for z/OS), to create a Business Process Choreographer *sample configuration*. This sample configuration is not supported in network deployment environments.

As a result of running the configuration script for a stand-alone server with the Business Process Choreographer sample configuration parameters set in the response file, bpeconfig.jacl is invoked and the database is created using the database settings that you supplied in the response file.

Important: The database for the Business Process Choreographer sample configuration should only be created for use with Cloudscape, and will only be configured if the **delayConfig=false**.

Because configuring the Business Process Choreographer database for a network deployment environment can be a complicated process requiring many parameters, it is done separately from the WebSphere Process Server configuration by running the Business Process Choreographer configuration script named *bpeconfig.jacl* manually.

Note: For a stand-alone configuration using DB2 for z/OS database, or for a managed node or deployment manager configuration, you must first create the database and storage group prior to running the configuration script.

You must make sure that the database name(s) and storage group name(s) that you set when you created the database and storage group match the database name(s) and storage group name(s) in the response file.

SQL scripts

All the scripts to create and manage the Business Process Choreographer database can be found in the directory

All the scripts to create and manage the Business Process Choreographer database can be found in the directory `*configuration_root*/*app_server_root*/dbscripts/ProcessChoreographer/*dbType*`. These scripts are described in detail in *Using scripts to administer Business Process Choreographer* in the *WebSphere Process Server for z/OS, version 6.0.2 Business Process Choreographer* PDF. Or view the topics in the WebSphere Process Server for z/OS, version 6.0.2 online information center at http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp by navigating to **Administering WebSphere Process Server > Administering Business Process Choreographer > Using scripts to administer Business Process Choreographer**.

There are scripts to create the database (for simple setups), to create tablespaces, dbspaces and the schema (for advanced setups), to clear the schema, to drop the schema, and to drop the tablespaces or dbspaces. These scripts can either be run manually by a database administrator or you can run the bpeconfig.jacl script which will invoke the scripts for you.

JDBC provider

If an appropriate JDBC provider does not exist one is created.

The data source name is:

BPEDataSource*dbType*

The data source JNDI name is:

jdbc/BPEDB

Restrictions

Create new database is disabled for the following database types.
- DB2 Universal Database for OS/390 V7.1 (DB2UDBOS390_V7_1)
- DB2 Universal Database for OS/390 V8.1 (DB2UDBOS390_V8_1)

Cloudscape is not supported in network deployment environments.

Tables

Many tables are created. See the generated scripts to see which tables are generated for the given database product.

Exported scripts

Business Process Choreographer does not export any scripts. All scripts needed to configure the Business Process Choreographer database are deployed as part of the product installation to the

```
configuration_root/app_server_root/dbscripts/ProcessChoreographer/dbType
```

directory.

User ID privileges

Generally the user credentials that you provide in the response file should be able to create tablespaces, tables, indexes, and stored procedures. For the Create new database option, the user identity should have the necessary privileges to create a new database.

Messaging engine database specifications

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

The messaging engine component provides the core messaging functionality inside the application server. The messaging engine runs in the CRA. Messaging engines can communicate with each other. Once the messaging engines are defined in a bus, the intercommunication is implicit without explicit configuration.

Each message engine has its own data store with a set of tables in a DB2 for z/OS database . These tables are used to hold durable data by a messaging engine. For example, a persistent message is stored in the data store.

This section covers the database as used by the message engines for Service Component Architecture (SCA), Business Process Choreographer, and Common Event Infrastructure. The default database name for the SCA messaging engine is SCADB, for the other messaging engines it is MEDB. For the embedded Cloudscape database, each messaging engine will have its own database or schema. The default schema name is IBMWSSIB.

Configure your SCA messaging engine using the administrative console via the Service Component Architecture Configuration panel.

You have a lot of control over the messaging engine databases, for example, you can create a database for each messaging engine or you can use a single database for all the messaging engines. Similarly you can use many database schemas, or a single one.

Supported database types

The following list is the databases and versions that are supported as messaging engine databases.

- Cloudscape
- DB2 z/OS 7 & 8 (DB2 Legacy CLI-based Type 2 JDBC Driver (XA))
- DB2 z/OS 7 & 8 (DB2 Universal JDBC Driver Provider (XA))

Database Management Service (DBMS) instances

Each messaging engine has its own database or schema to support the message engine:

- One is used to host each messaging engine for the system bus.
- Another is used to host each messaging engine for the application bus.

- Another is used to host each messaging engine for the Common Event Infrastructure bus.
- Another is used to host each messaging engine for the Business Process Choreographer bus.

The naming convention for the JDBC data source that the messaging engine uses to interact with the database is:

- System bus: <*node*><*server*>-SCA.SYSTEM.<*cell*>..Bus
- Application bus: <*node*><*server*>-SCA.APPLICATION.<*cell*>.Bus
- Common Event Infrastructure bus: <*node*><*server*> | <*cluster*>-CommonEventInfrastructure_Bus
- Business Process Choreographer bus: <*node*><*server*>.-BPC.<*cell*>.Bus

Cloudscape database naming convention is shown below:

- System bus: *configuration_root/app_server_root*/profiles/*profilename*/databases/com.ibm.ws.sib/(<*node*>.<*server*> | <*cluster*>)-SCA.SYSTEM.<*cell*>.Bus
- Application bus: *configuration_root/app_server_root*/profiles/*profilename*/databases/com.ibm.ws.sib/(<*node*>.<*server*> | <*cluster*>)-SCA.APPLICATION.<*cell*>.Bus
- Common Event Infrastructure bus: *configuration_root/app_server_root*/profiles/*profilename*/event/CloudScapeEventBusDB/(<*node*>.<*server*> | <*cluster*>)-CommonEventInfrastructure_Bus
- Business Process Choreographer bus: *configuration_root/app_server_root*/profiles/*profilename*/databases/com.ibm.ws.sib/(<*node*>.<*server*> | <*cluster*>)-BPC.<*cell*>.Bus

The default for <*cell*> is the cell name. You can override this with your own bus identifier name. You cannot use the administrative console to create customized names, but you can use administrative tasks to do so.

Database actions invoked by the product configuration script

Stand-alone profile

The message engine database is created during the installation and configuration.

Network deployment

You can configure the message engines for Service Component Architecture using the Application servers > *servername* > Service Component Architecture panel of the administrative console.

The following administrative tasks are performed during profile creation:

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

For both tasks, the following parameters are handed over:

- busDataSource
- meAuthAlias
- busSchemaName
- createTables
- systemBusId (the default is the cell name)

For Business Process Choreographer the messaging engine configuration is performed using the bpeconfig.jacl script. For Common Event Infrastructure the provided scripts can be used to configure the messaging engine.

SQL scripts

No SQL scripts are created as part of the WebSphere Process Server product, existing base WebSphere Application Server for z/OS scripts can be used to create database and tables if necessary. The MEDB needs to be created manually before it is configured using the Application servers > *servername* > Service Component Architecture panel of the administrative console.

JDBC provider

Service Component Architecture

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

Business Process Choreographer

The JDBC provider creation for the messaging engine database is similar to the approach followed in the creation of the BPEDB see “Business Process Choreographer database specifications” on page 123 for more details.

Common Event infrastructure

The JDBC provider creation for messaging engine database is similar to the approach followed in the creation of the CEIDB see “Common Event Infrastructure database specifications” on page 122 for more details.

Restrictions

There are no known restrictions.

Tables

For information on the tables see the WebSphere Application Server for z/OS topic on data stores.

Exported scripts

No scripts are exported.

User ID privileges

Generally the user credentials that you provide in the response file should be able to create tablespaces, tables, indexes, and stored procedures.

Enterprise service bus logger mediation database specifications

Messages processed by mediation modules can be logged in the enterprise service bus logger mediation database (EsbLogMedDB).

The enterprise service bus logger mediation (EsbLogMedDB) database is used by the MessageLogger mediation primitive in WebSphere Process Server. It is created automatically for a stand-alone configuration. A set of DDL files are provided to allow you to use additional databases either in a stand-alone server configuration or for a network deployment environment.

For a stand-alone configuration using a DB2 for z/OS database, or for a managed node or deployment manager configuration, you must create the Enterprise Service Bus database and storage groups prior to running the WebSphere Process Server for z/OS configuration script.

When you run the WebSphere Process Server for z/OS configuration script, make sure that the Enterprise Service Bus database and storage group names in the response file match the names that you set when you created the database and storage groups. For information on database and storage groups, see Creating databases and storage groups.

Supported database types

A Cloudscape 5.1 database is created automatically when you run the configuration script with the stand-alone response file (standAloneProfile.rsp). Users can manually create databases for all the other supported database types, using the appropriate DDL file. The DDL file is called Table.ddl and is found in the *configuration_root/app_server_root*/util/EsbLoggerMediation/*dbType* directory. The same file can be used to create the EsbLogMedDB for a network deployment environment.

Database Management Service (DBMS) instances

Only one DBMS instance is automatically created for a stand-alone configuration; however, users can manually create as many other instances as they require (each message logger mediation primitive can be configured to use a different data source and hence a different database).

Database actions invoked by the product configuration script

For a stand-alone sever configuration:

- When you configure WebSphere Process Server for a stand-alone server a Cloudscape database is created in `*configuration_root*/*app_server_root*/profile/*profileName*/databases` directory if one does not already exist.
- At the server scope, a suitable database provider is found or created and an accompanying data source is created.
- This action is performed for all WebSphere Process Server stand-alone profile augmentations – you cannot override this augmentation action.

Deployment manager configuration:

Manually create the database using the file Table.ddl.

SQL scripts

The file createMessageLoggerResource.jacl is located in `*configuration_root*/*app_server_root*/bin` and is used during the product configuration. DDL files for all database types are located in the `*configuration_root*/*app_server_root*/util/EsbLoggerMediation/<*dbType*>` directories.

JDBC provider

If a suitable JDBC provider is obtained at server scope, then this is used. The name of the data source is “ESB Logger Mediation Data Source”. You can create your own data source to use if you wish to configure a Message Logger mediation to use a different data source.

Restrictions

Cloudscape 5.1 is created in a stand-alone profile. No automatic creation is performed for network deployment. The Message Logger mediation runtime code assumes a fixed table name of MSGLOG and a fixed schema qualifier of ESBLOG.

You can supply the schema qualifier to be used by the runtime. If this value is not supplied, the default value of ESBLOG will be used.

You can override the schema qualifier when the DB2 database resides on z/OS, by providing a server scope **NamespaceBinding**, as follows:

```
Binding Identifier:   ESBMessageLogger
Name In Namespace:    esb/messageLogger/qualifier
String Value:         <schema qualifier>
```

Tables

Only one table is created called MSGLOG.

Exported scripts

No scripts are exported.

User ID privileges

Generally the user credentials that you provide in the response file should be able to create tablespaces, tables, indexes, and stored procedures.

Selector and Business Rules Group database specifications

When you install an application containing business rule or selector artifacts, the server stores these artifacts in database tables so that you can dynamically update them without changing the application.

The selector and business rules group components use a database to hold the selector and business rule artifacts that are created in WebSphere Integration Developer and installed on the server. If you make any changes to a selector through the administrative console or to business rules through the business rule manager, the database is updated with the latest information. The original artifacts in the EAR are not synchronized with any updates made after the application is installed.

Supported database types

The following databases are supported:

- Cloudscape
- DB2 Universal Database for z/OS V8.1
- DB2 Universal Database for z/OS and OS/390 V7.1

When running the configuration script with the response file standAloneProfile.rsp, Cloudscape is the database and a separate RepositoryDB is used.

Database Management Service (DBMS) instances

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

Database actions invoked by the product configuration script

Stand-alone configuration

- If you are configuring WebSphere Process Server for a stand-alone application server using the response file standAloneProfile.rsp with Cloudscape, the configDynamicArtifactRepository.ant script is invoked by the configuration script (zWPSConfigsh.sh or zWESBConfig.sh). This script calls the configDynamicArtifactRepository.jacl command to create the Cloudscape JDBC provider at the node level for the RepositoryDB.
- The RepositoryDB database is created in `/*configuration_root*/*app_server_root*/cloudscape/databases/RepositoryDB`. The default profile for the server installation will use the same database. Multiple servers cannot be started and use rules, as Cloudscape allows only one JVM to access the database at a time.

Deployment manager configuration

If you are configuring WebSphere Process Server for network deployment configuration using the response file DmgrDB2.rsp, the CommonDB for the cell is expected to be set up with the appropriate tables for selector and business rule group components. All selectors and business rules for the server in a deployment manager or managed node will use this database.

SQL scripts

The SQL scripts for table creation are available at `/*configuration_root*/*app_server_root*/dbscripts/CommonDB/*dbType*`. For example, `/*WebSphere*/*V6R0*/*AppServer*/dbscripts/CommonDB/*dbType*`

JDBC provider

The Common database data source is used.

Restrictions

DB2 v7.0 for z/OS has a limitation on the length of the primary key to not exceed 256 bytes. The primary key for selectors and business rules is composed of the Name, Namespace, and Type. If DB2 v7.0 for zOS is being used, ensure the Name and Namespace of the selectors and business rules do not exceed 245 bytes. These values can be modified in WebSphere Integration Developer. The Type value is not changeable and is approximately 8 bytes. There are technotes on this limitation:

- Technote 1226352 – http://www-1.ibm.com/support/docview.wss?rs=0&q1=1226352&uid=swg21226352&loc=en_US&cs=utf-8&cc=us&lang=en
- Technote 1226351 – http://www-1.ibm.com/support/docview.wss?rs=0&q1=1226352&uid=swg21226352&loc=en_US&cs=utf-8&cc=us&lang=en

When applications that contain selectors or business rules are uninstalled from the server or cell, the artifacts stored in the Common database or RepositoryDB databases are not removed. These must be removed manually following the instructions in the topic *Removing business rule and selector data* in the *Administering WebSphere Process Server for z/OS, version 6.0.2* PDF. Or view the topics in the WebSphere Process Server for z/OS, version 6.0.2 online information center at http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/index.jsp by navigating to **Administering WebSphere Process Server > Administering WebSphere Process Server resources > Overview of business rules and selectors > Considerations for modules containing business rules and selectors > Removing business rule and selector data from the repository**.

Tables

For selectors and business rules, there are three tables which are used to hold the artifacts:

- ByteStore
- ByteStoreOverflow
- AppTimestamp

Exported scripts

The scripts are exported to the same location as CommonDB, except when using Cloudscape where no scripts are exported.

User ID privileges

Generally the user credentials that you provide should be able to create tablespaces, tables, indexes, and stored procedures.

JDBC providers

JDBC providers allow applications to interact with relational databases.

Applications use JDBC providers to interact with relational databases. The JDBC provider supplies the specific JDBC driver implementation class for access to a specific vendor database. To create a pool of connections to that database, you associate a data source with the JDBC provider. Together, the JDBC provider and the data source objects are functionally equivalent to the J2EE Connector Architecture (JCA) connection factory, which provides connectivity with a non-relational database.

The general recommendation is to use the DB2 Universal JDBC Driver Type 2 implementation for local DB2 connections and Type 4 implementation for remote DB2 connections. ″Local″ means that the DB2 Subsystem locates in the same LPAR as WebSphere Process Server. Whereas the ″remote″ refers to the DB2 Subsystem located on a different LPAR from the WebSphere Process Server.

For more information on JDBC providers, see JDBC Providers in the WebSphere Application Server for z/OS, version 6.0 information center.

Data sources

Data sources provide the link between applications and relational databases.

Applications use a data source to obtain connections to a relational database. A data source is analogous to the J2EE Connector Architecture (JCA) connection factory, which provides connectivity to other types of enterprise information systems (EIS).

A data source is associated with a JDBC provider, which supplies the driver implementation classes that are required for JDBC connectivity with your specific vendor database. Application components transact directly with the data source to obtain connection instances to your database. The connection pool that corresponds to each data source provides connection management.

You can create multiple data sources with different settings, and associate them with the same JDBC provider. For example, you might use multiple data sources to access different databases within the same vendor database application. WebSphere Process Server requires JDBC providers to implement one or both of the following data source interfaces, which are defined by Sun Microsystems. These interfaces enable the application to run in a single-phase or two-phase transaction protocol.

- ConnectionPoolDataSource - a data source that supports application participation in local and global transactions, excepting two-phase commit transactions. When a connection pool data source is involved in a global transaction, transaction recovery is not provided by the transaction manager. The application is responsible for providing the backup recovery process if multiple resource managers are involved.
- XADataSource - a data source that supports application participation in any single-phase or two-phase transaction environment. When this data source is involved in a global transaction, the WebSphere Application Server for z/OS transaction manager provides transaction recovery.

For more information on data sources, see Data sources in the WebSphere Application Server for z/OS, version 6.0 information center.

Considerations for creating the database

WebSphere Process Server for z/OS product configurations include features that utilize databases.

As part of your WebSphere Process Server for z/OS configuration planning, you create the database and the database storage groups required by WebSphere Process Server for z/OS components that utilize databases.

The WebSphere Process Server for z/OS component features that utilize databases include the following:

- Common Event Infrastructure

  The Common Event Infrastructure utilizes two databases. An event database and an event catalog database

  This means that you will be required to create the database and storage groups for both of these databases.
- Business Process Choreographer

- Components that share a common database. These common components include the following:
  - Relationship services
  - Mediation
  - Recovery
  - Business rules
  - Selector
  - Scheduler

The WebSphere Process Server for z/OS product package includes a sample file with default values for the database and database storage groups that you need to create prior to running the WebSphere Process Server for z/OS configuration script.

You can tailor values in the sample and use database creation utility to create the database and storage groups.

For detailed information, see Creating databases and database storage groups.

Database naming conventions

If you only have one server configured with WebSphere Process Server for z/OS using a DB2 subsystem, you can use the default database names as provided in the sample response files.

For multiple WebSphere Process Server for z/OS-configured servers (in the same cell, or different cells) using the same DB2 subsystem or same data-sharing group, you must plan for a naming convention to isolate the following DB2 components:

- Database names
- Storage Group names
- Schema-qualifiers for your tables
- VCATs (hi-level qualifiers for the backing VSAM data sets)
- Volumes or SMS storage group for these data sets.

The following example is a database naming convention (for nine databases) in which the cell name is B6CELL:

```
Data-   Dbase     Storage    Schema    VCAT
 base    Name      Group      -Owner    DSN-hlq
-----   --------  --------   -------   -------
WPS:    B6WPSDB   B6WPSSTO   B6CELL    B6WPS
BPE:    B6BPEDB   B6BPESTO   B6CELL    B6WPS
SIBs:   B6SIBAPP  B6SIBSTO   B6CELLA   B6WPS
        B6SIBSCA  B6SIBSTO   B6CELLS   B6WPS
        B6SIBBPC  B6SIBSTO   B6CELLB   B6WPS
        B6SIBCEI  B6SIBSTO   B6CELLC   B6WPS
CEI:    B6EVTDB   B6EVTSTO   B6CELL    B6WPS
        B6EVCTDB  B6EVTSTO   B6CELL    B6WPS
ESB:    B6ESBDB   B6ESBSTO   B6CELL    B6WPS
```

Schema Names and SQLIDs

The DB2 tables must have uniquely-qualified schema names to coexist with other WebSphere Process Server for z/OS or WebSphere Business Integration Server cells in the same DB2 subsystem. You can set the current schema or SQL ID value using the administrative console, in the Data source > Custom properties definitions.

The DB2 table and index names can be prefixed with these names in the DDL definitions, or you can insert a SET CURRENT SQLID and COMMIT statements in front of the DDL used to create the tables. For example:

```
SET CURRENT SQLID = 'B6CELL';
COMMIT;
```

Another way to uniquely qualify the database tables, is to use the WebSphere Application Server for z/OS Servant's User ID as the qualifier. Each of the SI Bus databases requires unique schema names for their tables, because the same table names are used for all Buses. These same schema names must also be set in the SI Bus Messaging Engine Data Store properties. See Messaging engines in the WebSphere Application Server for z/OS information center.

Grants and permissions

We use a userid with 'SYSADM' authority ('sysadm1' in our case) to run all the DB2 scripts and jobs to create databases, storage groups, tables, etc. This userid is also used to define the authentication aliases in the Data sources.

The user IDs used by the WebSphere Process Server for z/OS servant started task, and Deployment Manager started task may need authorization to access and create tables in the WebSphere Process Server for z/OS runtime.

Work with your DB2 administrator to determine what authorizations are required in your environment to create and access these databases and tables.

DASD space requirements for DB2 table spaces and tables

If you use the generated sample DDL statements for all nine databases, they may require over 3500 cylinders of 3390 DASD storage, depending on your DB2 system default allocations.

For a simple test environment, you can reduce the space by altering or adding a 'PRIQTY nnn' parameter to the USING STOGROUP <storage_group_name> clauses in the sample DDL.

The PRIQTY value units are the number of 1K bytes. If not specified, the default value depends on your DB2 system defaults. In a test environment, you might reduce these values.

Creating databases and storage groups

Create the databases and storage groups used by WebSphere Process Server for z/OS components. WebSphere Process Server for z/OS provides sample scripts that are pre-configured with default values to create the databases and the associated storage groups.

Attention: The information in this topic applies only when your configuration uses DB2 for z/OS.

You must have loaded WebSphere Process Server for z/OS onto the system using SMP/E and have run the installation script zSMPInstall.sh successfully in order to access the sample file.

You must create the databases and storage groups utilized by WebSphere Process Server for z/OS prior to running the configuration script. When you run the configuration script, the DDL and SQL for the database tables is created. The databases to hold those tables must exist as a context for creating the DDL and SQL for the tables.

Note: WebSphere Process Server for z/OS packages a utility named **DBUtility.sh**. This utility is packaged in the installation HFS directory, for example, /usr/lpp/zWPS/V6R0/bin/DBUtility.sh.. You can use this utility to create the database and storage groups, as well as to run the SQL to create the database tables. You should consult with your database administrator before using **DBUtility.sh**. For information on using **DBUtility.sh** to run the SQL to create the database tables, see Configuring the database manually.

1. Access the sample file(s) from the following directory paths.

   Note: The WebSphere Application Server for z/OS *configuration_root/ app_server_root* will vary depending on the configuration type. For a stand-alone configuration, the variable */configuration_root/app_server_root* = /WebSphere/V6R0M0/AppServer. For a deployment manager configuration, the variable */configuration_root/app_server_root* = /WebSphere/V6R0M0/DeploymentManager

   - For Business Process Choreographer, the sample file of database and associated storage groups is located in the following directory: /*configuration_root/app_server_root*/dbscripts/ProcessChoreographer/DB2/createBpcDb2Zos.sql.
   - For Common Event Infrastructure, the sample file of the event and eventcat databases and associated storage groups is located in the following directory: /*configuration_root/app_server_root*/dbscripts/CEI/DB2/createCeiDb2Zos.sql.
   - For WebSphere Process Server for z/OS features using the common database, the sample file is located in the following directory: /*configuration_root/ app_server_root*/dbscripts/CommonDB/DB2/createCommonDb2Zos.sql.
   - For Enterprise Service Bus, the sample file of database and associated storage groups is located in the following directory:/*configuration_root/ app_server_root*/dbscripts/EsbLoggerMediation/DB2/createEsbDb2Zos.sql

   The files apply to DB2 version 7 and DB2 version 8.
2. Copy the sample file(s) from the directory listed above to your working directory. For example, to copy the common database sample file, perform the following: cp /*configuration_root/app_server_root*/dbscripts/CommonDB/DB2/createCommonDb2Zos.sql/*myworkingdirectory*
3. Edit the values in the file to suit your needs. Change the database names and the storage group names to meet your naming requirements.

   Note: The sample files are delivered in ASCII format. Depending on the capabilities of the tool you use to view, edit and run this file, you may need to convert the file to a readable format, EBCDIC for example. You can use iconv to convert the file to EBCDIC, and then use vi as shown in the following example:

   ```
   iconv –t IBM-1047 –f ISO8859-1 createCommonDb2Zos.sql >
   createCommonDb2Zos_EBCDIC.sql
   ```

   ```
   vi createCommonDb2Zos_EBCDIC.sql
   ```

   Note: If you convert the file from ASCII format to EBCDIC, you must *reconvert* the file back to ASCII format in order for the file to run correctly. Use iconv to convert the file back to ASCII, for example:

   ```
   iconv -t ISO8859-1 -f IBM-1047 createCommonDb2Zos_EBCDIC.sql >
   createCommonDb2Zos_ASCII.sql
   ```

Note: The names that you specify in the sample file need to be carried forward as values in the response file that is referenced in the product configuration script.

4. Create the database and storage groups. The sample files are not executed programmatically during the configuration process. Use the standard database definition tools and procedures to submit the SQL to create the database and storage groups.
5. Verify that each one executes successfully with no errors by inspecting the output.

The following are examples of the sample files of default database and storage group names packaged with WebSphere Process Server for z/OS

Business Process Choreographer

```
-- ########################################################################
-- # Licensed Materials - Property of IBM
-- # 5724-L01, 5724-I82, 5655-N53, 5655-R15
-- # (C) Copyright IBM Corporation 2006. All Rights Reserved.
-- # US Government Users Restricted Rights - Use, duplication or disclosure
-- # restricted by GSA ADP Schedule Contract with IBM Corp.
-- ########################################################################
SET CURRENT SQLID = 'WSADMIN';

DROP   DATABASE BPEDB ;
DROP   STOGROUP BPEDBSTO;
CREATE STOGROUP BPEDBSTO VOLUMES (WBIUS3,WBIUS4,WBIUS5) VCAT WSDB2;
CREATE DATABASE BPEDB
  STOGROUP BPEDBSTO
  BUFFERPOOL BP0
  CCSID UNICODE
  INDEXBP BP0;
COMMIT;
```

WebSphere Process Server for z/OS features using the common database:

```
-- ########################################################################
-- # Licensed Materials - Property of IBM
-- # 5724-L01, 5724-I82, 5655-N53, 5655-R15
-- # (C) Copyright IBM Corporation 2006. All Rights Reserved.
-- # US Government Users Restricted Rights - Use, duplication or disclosure
-- # restricted by GSA ADP Schedule Contract with IBM Corp.
-- ########################################################################
SET CURRENT SQLID = 'WSADMIN';

DROP   DATABASE WPRCSDB ;
DROP   STOGROUP WPRCSSTO;
CREATE STOGROUP WPRCSSTO VOLUMES (WBIUS3,WBIUSR,WBIUS5) VCAT WSDB2;
CREATE DATABASE WPRCSDB
  STOGROUP WPRCSSTO
  BUFFERPOOL BP0
  CCSID UNICODE
  INDEXBP BP0;
COMMIT;
```

The WebSphere Process Server for z/OS features that use the common database include the following:

- Recovery
- Relationship service
- Mediation
- Application Scheduler
- Customization (selector and business rule group)

Common Event Infrastructure

```
-- ########################################################################
-- # Licensed Materials - Property of IBM
-- # 5724-L01, 5724-I82, 5655-N53, 5655-R15
-- # (C) Copyright IBM Corporation 2006. All Rights Reserved.
-- # US Government Users Restricted Rights - Use, duplication or disclosure
-- # restricted by GSA ADP Schedule Contract with IBM Corp.
-- ########################################################################
SET CURRENT SQLID = 'WSADMIN';

DROP   DATABASE EVENT ;
DROP   DATABASE EVENTCAT ;
DROP   STOGROUP EVTSTO;
CREATE STOGROUP EVTSTO VOLUMES (WBIUS3,WBIUS4,WBIUS5) VCAT WSDB2;
CREATE DATABASE EVENT
  STOGROUP EVTSTO
  BUFFERPOOL BP0
  CCSID UNICODE
  INDEXBP BP0;
CREATE DATABASE EVENTCAT
  STOGROUP EVTSTO
  BUFFERPOOL BP0
  CCSID UNICODE
  INDEXBP BP0;
COMMIT;
```

Enterprise Service Bus

```
-- ########################################################################
-- # Licensed Materials - Property of IBM
-- # 5724-L01, 5724-I82, 5655-N53, 5655-R15
-- # (C) Copyright IBM Corporation 2006. All Rights Reserved.
-- # US Government Users Restricted Rights - Use, duplication or disclosure
-- # restricted by GSA ADP Schedule Contract with IBM Corp.
-- ########################################################################
SET CURRENT SQLID = 'WSADMIN';

DROP   DATABASE ESBDB ;
DROP   STOGROUP ESBDBSTO;
CREATE STOGROUP ESBDBSTO VOLUMES (WBIUS3,WBIUS4,WBIUS5) VCAT WSDB2;
CREATE DATABASE ESBDB
  STOGROUP ESBDBSTO
  BUFFERPOOL BP0
  CCSID UNICODE
  INDEXBP BP0;
COMMIT;
```

Now you are ready to edit the response files. See Working with response files for more information.

Working with response files

Response files loaded onto the z/OS system during the SMP/E portion of the installation contain default properties for supported WebSphere Process Server for z/OS configurations.

The content of the response file includes properties for the resources that define the runtime environment. You provide the path to a response file as part of the command that drives the WebSphere Process Server for z/OS configuration.

For a listing of the sample response files that are included with the product, see Choosing a response file.

The following syntax diagram indicates where in the command to specify the response file.

Configuring WebSphere Process Server for a stand-alone configuration using DB2 for z/OS

```
►►──zWPSConfig.sh──-augment────────────────────────────────────────────────────►

  (1)
►──────┬-response───────────────────────────────────────────────┬──────────►◄
       └──/usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp──┘
```

Notes:

1 Type the absolute path of the response file that contains the properties required for your product configuration. The path shown above is the path to the default response file packaged with the product. The path name that you would type must represent the path location of the response file that you modified with property values specific to your environment.

Things to know before you use response files

Before you run the configuration script, you should know the following characteristics of response files:

- The response files shipped with the product contain default values that you will need to edit to suit your environment.
- Before you edit any of the response files, you will need to create databases and the associated storage groups.

  Some of the database properties in the response file must match names of the databases and associated storage groups. These instances are notated in the sample response files.
- The properties that you set in a response file may result in you having to perform additional configuration steps.

  Read the comment sections of the chosen response file before setting the values. The comment sections contain important information on setting properties, including descriptions of what happens as a result of setting certain values for properties.
- Be careful when you add comments to a response file.

  If you put comments in the response file, and that comment resides on the same line as a property value, even if you precede the comment with a cross hatch character (#), the Ant script attempts to read the comment as part of the parameter value, causing unpredictable results. For example, do not enter comments as shown in this example:

  ```
  ################################################################################
  # DB2 Properties
  ################################################################################
  dbJDBCClasspath=/shared/db2810/jcc/classes    # DB2 ClassPath Location
  dbJDBCProperties=/u/hutch/wpswork/            # DB2JccConfiguration.properties
  ```

  The correct way to add comments shown in the sample above would be as follows:

  ```
  ################################################################################
  # DB2 Properties
  ################################################################################
  ```

```
# DB2 ClassPath Location
dbJDBCClasspath=/shared/db2810/jcc/classes
# DB2JccConfiguration.properties
dbJDBCProperties=/u/hutch/wpswork/
```

- Be careful of ″empty spaces″ at the end of a value for a property.

  Embedded spaces at the end of the property in your response file can cause problems. Make sure that you do not have any additional spaces at the end of a property value in your response file.

Reading a response file

The response file contains a lot of helpful information in the form of comments.

You should print out the response file and read the commented text prior to running the configuration script. The response file comments help you understand how to set values for your particular environment.

The top of the response file contains a section named **GLOBAL Properties**. Here is an example:

```
################################################################################
# GLOBAL Properties
################################################################################
JMSUSER=ibmuser
JMSPASS=ibmuser
DBUSER=wsadmin
DBPASS=wsadmin
CONFIGSERVER=server1
DBLOCATION=LOC1

################################################################################
```

The **GLOBAL Properties** section provides a central location for ″common values″. Common values are values that display in multiple places (because they are used by multiple components) throughout the response file. Providing a central location for these common values facilitates the editing process. For example, if all server properties need to be set to **server1**, the use of a global property allows you to see and, if necessary, change all instances by changing the value of the global property only. You would not be required to search the response file for all instances of the property that you want to change.

You use the GLOBAL property by specifying the global keyword prefixed with the $ symbol as the value for a subsequent property. For example, serverName=$CONFIGSERVER.

Global properties must physically appear in the response file before they are referenced by the $ symbol.

When a global property is used, it must constitute the whole value of a property. In other words, the following is NOT allowed:

```
templatePath=/usr/$USERPATH/dir
```

Global properties can also simplify the use of the override argument (-Z) in the product configuration command by reducing command line clutter.

The comments for a property display above the property. The property and the default value display in bold text. As shown in the following example:

```
################################################################################
#
# Profile name
#
# On z/OS, there is always one and only one profile and that profile is named
# default in each of the configurations.
#
# The profile referred to here is the default profile installed and
# and configured during the WebSphere Application Server for z/OS install.
#
profileName=default
```

Choosing the response file

Choose the correct response file for the type of configuration that you are creating.

Many of the response file parameters affect the JDBC provider data source definitions in WebSphere, and corresponding database definitions in DB2. Make sure to use the appropriate response file template, which you can copy from the ../zWPS/V6R0/zos.config directory. There are many similarities between these templates, but there are also many differences. Be sure to read the comment sections of the response file templates.

Choosing a response file requires an understanding of the different WebSphere Process Server for z/OS configurations.

WebSphere Process Server for z/OS includes sample response files for each of the supported configuration types.

The sample response files were loaded onto the system and are part of the HFS that was created during the SMP/E phase of the installation.

When you run **zWPSConfig.sh** or **zWESBConfig.sh**, the keyword -response is followed by the user defined absolute path of the response file.

As part of the configuration actions, the script passes the absolute path name of the response file into the configuration manager, which in turn runs a sequence of Ant scripts to update the administrative console with WebSphere Process Server for z/OS definitions and augment the default profile.

Response files used for creating a stand-alone configuration

WebSphere Process Server for z/OS includes the following response files for creating a stand-alone configuration:

- standAloneProfile.rsp
- standAloneProfileDB2.rsp

standAloneProfile.rsp

The sample standAloneProfile.rsp is located in /usr/lpp/zWPS/V6R0/zos.config/**standAloneProfile.rsp** .

Note: The directory specified assumes an smpe_root (installation HFS) of `/usr/lpp/zWPS/V6R0/zos.config`.

The standAloneProfile.rsp includes configuration parameters for the Cloudscape database.

You should use this response file for your first configuration, because it provides you with a fully functional WebSphere Process Server for z/OS server for testing applications, with a minimum amount of effort. A stand-alone configuration created with the standAloneProfile.rsp is not meant for use in a production environment.

standAloneProfileDB2.rsp

The sample standAloneProfileDB2.rsp is located in /usr/lpp/zWPS/V6R0/zos.config/**standAloneProfileDB2.rsp** .

Note: The directory specified assumes an smpe_root (installation HFS) of `/usr/lpp/zWPS/V6R0/zos.config`.

The standAloneProfileDB2.rsp includes configuration parameters for the DB2 for z/OS database.

The use of this response file requires some additional configuration requirements to set up the DB2 for z/OS databases.

This response file supports DB2 for z/OS version 7 and DB2 for z/OS version 8. The default properties assume a DB2 for z/OS version 8. If you have DB2 for z/OS version 7 installed, you will need to modify the default values in this response file to support DB2 for z/OS version 7.

The response file contains comments on those properties that apply to DB2 for z/OS version 7 specifically.

Response files used for creating a deployment manager in a network deployment configuration

WebSphere Process Server for z/OS includes response file for creating a deployment manager in a network deployment configuration. The sample response file is named DmgrDB2.rsp and is located in /usr/lpp/zWPS/V6R0/zos.config/**DmgrDB2.rsp**

Note: The directory specified assumes an smpe_root (installation HFS) of `/usr/lpp/zWPS/V6R0/zos.config`.

The DmgrDB2.rsp file includes configuration parameters for the DB2 for z/OS database.

The use of this response file requires some additional configuration requirements to set up the DB2 for z/OS databases.

This response file supports DB2 for z/OS version 7 and DB2 for z/OS version 8. The default properties assume a DB2 for z/OS version 8. If you have DB2 for z/OS version 7 installed, you will need to modify the default values in this response file to support DB2 for z/OS version 7.

The response file contains comments on those properties that apply to DB2 for z/OS version 7 specifically.

Creating a network deployment cell requires manual configuration of resources, containers, applications and other infrastructure, but it is similar to the standalone server with DB2 in terms of the database setup.

Response files used for creating a managed node configuration

WebSphere Process Server for z/OS includes response file for creating a managed node configuration. The sample response file is named ManagedDB2.rsp and is located in /usr/lpp/zWPS/V6R0/zos.config/**ManagedDB2.rsp**

Note: The directory specified assumes an smpe_root (installation HFS) of `/usr/lpp/zWPS/V6R0/zos.config`.

The ManagedDB2.rsp file includes configuration parameters for the DB2 for z/OS database.

The use of this response file requires some additional configuration requirements to set up the DB2 for z/OS databases.

This response file supports DB2 for z/OS version 7 and DB2 for z/OS version 8. The default properties assume a DB2 for z/OS version 8. If you have DB2 for z/OS version 7 installed, you will need to modify the default values in this response file to support DB2 for z/OS version 7.

The response file contains comments on those properties that apply to DB2 for z/OS version 7 specifically.

Sample response files

WebSphere Process Server for z/OS comes with sample response files that are installed into the read-only HFS.

Purpose

As part of the configuration command, you pass in a response file associated with the configuration you are creating.

The response file contains the properties for the WebSphere Process Server for z/OS product configuration.

The following sections contain notated samples of the response files.

Important: The samples documented in this section are for documentation purposes only. Do not use the samples in this section as the actual response file, as they contain documentation-specific information such as links to topics in the information center. Copy and tailor the response files from the read-only HFS when creating your WebSphere Process Server for z/OS product configuration.

Response file for standalone configuration using Cloudscape - standAloneProfile.rsp

The following is a sample of the default response file used to create a WebSphere Process Server for z/OS stand-alone configuration that uses the Cloudscape database. Although a stand-alone configuration that uses a Cloudscape database is not appropriate for use in a production environment, it is a good configuration to create your first time, as it provides you with a fully functional server for z/OS server for testing applications, with a minimum amount of effort.

Note: The sample response file below contains properties for Business Process Choreographer and WebSphere Business Integration Server. Business Process

Choreographer and WebSphere Business Integration Server are not configured for an ESB-only server configuration of WebSphere Process Server for z/OS. These properties are ignored by the configuration processing.

```
################################################################################
# Licensed Materials - Property of IBM
# 5655-N53, 5655-R15
# (C) Copyright IBM Corporation 2006, 2007. All Rights Reserved.
# US Government Users Restricted Rights - Use, duplication or disclosure
# restricted by GSA ADP Schedule Contract with IBM Corp.
################################################################################
################################################################################
#
# Sample Response file for creating a WebSphere Process Server for z/OS
# stand-alone configuration using a Cloudscape database
#
################################################################################
################################################################################
#
# Common Properties
#
################################################################################
#
# Augment
#
# The "augment" keyword is required in order for the configuration to
# invoke the profile augmentation actions.
#
# There is no value associated with this keyword. The "augment" keyword
# may be replaced with "unaugment" in order to undo the results from a
# previous augment action. However, "augment" must be specified for the
# configuration to complete successfully.
#
#  Note: The uninstall command also references the response file.
#  However, you do not need to change the value from "augment" to
#  "unaugment" when you run the zSMPInstall.sh to uninstall the
#  product. For more information on uninstalling the product, see
#  the topic "Run the install script to uninstall the product" in the
#  information center.

#
augment

################################################################################
#
# Profile name
#
# The profile referred to here is the default profile installed and
# and configured during the WebSphere Application Server for z/OS
# installation.
#
# As "default" is the standard name for the profile, accept the default
# value for this property.
#
profileName=default

################################################################################
#
# Profile path
#
# This value should reflect the default profile root directory for
# WebSphere Application Server for z/OS
#
# The server configuration for WebSphere Application Server for z/OS
# resides in a directory structure under a "profiles" directory. So, for
# a base stand-alone application server in the config_root, there will
# be a "profiles" directory, and in that directory there will be a
# "default" directory that contains all of the directories and files
# (such as commands, configuration files, and log files) that define
# the runtime environment for your server configuration.
#
# In most cases the WebSphere Application Server for z/OS installation
# uses the default value shown below.
#
# The /WebSphere/V6R0M0/AppServer portion of the profile path
# shown below should be the configuration HFS / app_server_root for your base appserver.
#
```

```
# If the default value was not used to install and configure
# WebSphere Application Server for z/OS, make sure you edit the
# value below to represent the profile path that was used for the
# WebSphere Application Server for z/OS installation.
#
#
profilePath=/WebSphere/V6R0M0/AppServer/profiles/default
########################################################################
#
# Template path
#
# This value determines which component augment actions are performed as
# part of the configuration process. Valid values are as follows:
#
# /WebSphere/V6R0M0/AppServer/profileTemplates/default.wbicore
# /WebSphere/V6R0M0/AppServer/profileTemplates/default.bfm
# /WebSphere/V6R0M0/AppServer/profileTemplates/default.wbiserver
#
# /WebSphere/V6R0M0/AppServer/profileTemplates/default.*
#
# If one of the first three values is specified, only the augmentation
# actions within that directory are performed. The fourth value is the
# default (.../default.*),which when specified invokes augment actions
# for all of the template paths.
#
# Caution: Changing the default value causes an incomplete configuration
# of WebSphere Process Server for z/OS and should only be done if there
# is a valid reason to do so.
#
# The /WebSphere/V6R0M0/AppServer portion of the template path
# shown below should be the configuration HFS / app_server_root of your AppServer.
#
templatePath=/WebSphere/V6R0M0/AppServer/profileTemplates/default.*

################################################################################
#
# Cell name
#
# This value determines the cell name within the WebSphere Application
# Server for which this Process Server configuration applies. The
# default value is SY1.

#
#
cellName=SY1

################################################################################
#
# Node name
#
# This value determines the node name within the WebSphere Application
# Server for which this Process Server configuration applies. The
# default value is SY1.

#
# Node name under one cell has to be unique. The default value is SY1.
#
# The value assigned should be the LONGNAME of the node that you
# configured for your stand-alone appserver.
#
nodeName=SY1

################################################################################
#
# Server name
#
# This value determines the server name within the WebSphere Application
# Server for which this Process Server configuration applies. The
# default value is SY1.

#
# The value assigned should be the LONGNAME of the server that you
# configured for your stand-alone appserver.
#
serverName=server1

################################################################################
#
# WBI Core Properties
```

```
#
################################################################################
#
# Common Event Infrastruction Configuration
#
# The following entries are used to create a sample Common Event Infrastructure
# (CEI) configuration. ceiSampleJmsUser, ceiSampleJmsPwd, and ceiSampleServer
# values are required.
#
################################################################################
#
# Java messaging service (JMS) userid for the CEI sample configuration
#
# Specifies the value of the Java Messaging Service (JMS) userid for a Common
# Event Infrastructure (CEI) configuration. Default value is ibmuser.
#
ceiSampleJmsUser=ibmuser

################################################################################
#
# Java messaging service (JMS) password for the CEI sample configuration
#
# Specifies the value of the Java Messaging Service (JMS) password for a Common
# Event Infrastructure (CEI) configuration. Default value is ibmuser.
#
ceiSampleJmsPwd=ibmuser

################################################################################
#
# Server name for the CEI sample configuration
#
# Specifies the value of the server name for the Common Event Infrastructure (CEI)
# sample configuration. The default value is server1.
#
ceiSampleServerName=server1

################################################################################
#
# Database product name for Common Event Infrastructure
#
# Specifies the default value of the database product name for Common Event
# Infrastructure (CEI) processing. The default is CLOUDSCAPE_V51_1.
#
ceiDbProduct=CLOUDSCAPE_V51_1

################################################################################
#
# Configure SCA
#
################################################################################
#
# The following entries are used to create a Service Component Architecture
# sample configuration.
#
################################################################################
#
# Configure SCA security
#
# If you do not want to create a Service Component Architecture
# sample configuration, change the default value from true to false and
# leave the values for the properties specified below blank
#
configureScaSecurity=true

################################################################################
#
# SCA security user id
#
# If you chose to create a Service Component Architecture configuration above
# then specify the userid for that configuration. Default value is ibmuser.
#
scaSecurityUserId=ibmuser

################################################################################
#
# SCA security password
#
# If you chose to create a Service Component Architecture configuration above
# then specify the password for SCA security user id for that configuration.
```

```
#
scaSecurityPassword=ibmuser

################################################################################
#
# Business Process Choreographer Properties
#
# Set the properties below to create a sample Business Process
# Choreographer (BPC) configuration.
#
# If you do not wish to create a sample Business Process Choreographer
# configuration, remove the default values for bpcmqUser, bpcmqPwd and
# bpcadminGroups.
#
# The Business Process Choreographer sample using Cloudscape is valid
# for a stand-alone server configuration in a test environment only
# (it is not a valid configuration for a production environment).
#
# The values specified here for the Business Process Choreographer
# sample configuration are used to run the Business Process
# Choreographer configuration script (bpeconfig.jacl)
#
# If you choose not to create a sample Business Process Choreographer
# configuration at this time, you can run bpeconfig.jacl after running the
# product configuration script to create all of the Business Process
# Choreographer resources.
#
# Running bpeconfig.jacl is the recommended method for configuring the
# Business Process Choreographer resources used in WebSphere Process
# server for z/OS. For more information on running bpeconfig.jacl, see
# the topic "Using the bpeconfig.jacl script to configure Business
# Process Choreographer" in the information center.#

#
################################################################################
#
# MQ User Name
#
# Specify the user name for the configuration of resources for the Business
# Process Container using WebSphere MQ. The default is ibmuser.
#
bpcmqUser=ibmuser

################################################################################
#
# MQ Password
#
# Specify the password for the MQ User Name for the configuration of resources
# for the Business Process Container using WebSphere MQ. The default is ibmuser.
#
bpcmqPwd=ibmuser

################################################################################
#
# Admin BFM Users
#
# Specify the security role for the business process system administrator. This
# is the security role used by the Business Process Choreographer to run a
# business process. The default is ibmuser.
#
bpcadminUsers=ibmuser################################################################################
#
# WBI Server Properties
#
########################################################################
#
# Application Scheduler configuration
#
# Application Scheduler is used to schedule migration application group
# events. If you choose to configure Application Scheduler, set the
# property to "true". Otherwise, specify false.
#
# You must specify the server (into which the Application Scheduler
# application is installed) in the appSchedulerServer property below.
#
# For information on the Application Scheduler, see the topic
# "Application Scheduler" in the information center.

#
```

```
configureAppScheduler=true

########################################################################
#
# Indicate the application server name on this node if you chose to configure
# it for Application Scheduler use. Usually, for a stand-alone profile the
# server name is "server1". The default is server1.
#
# The value assigned should be the LONGNAME of the server that you
# configured for your stand-alone appserver.
#
appSchedulerServer=server1

################################################################################
#
# Configure Database Authentication for process server
#
# Various WebSphere Process Server for z/OS components use a database connection.
#
# Choose a database type and enter the database configuration information
# based on that type.
#
################################################################################
#
# Database name
#
# Specify the name of the WebSphere Process Server database. The default is
# WBIDB.
#
dbName=WBIDB

################################################################################
#
# Database type
#
# Specify the type of the WebSphere Process Server database. The default is
# Cloudscape.
#
dbType=Cloudscape

################################################################################
#
# Create new or use existing database
#
# You can choose to use an existing database for WebSphere Process Server or
# you can create a new database. To use an existing database, set the following
# property to "false". To create a new database set the following property
# to "true". The default is "true".
#
dbCreateNew=true

################################################################################
#
# Location of database server (database product installation root)
#
# Specify the database location. The database server location is the
# installation root for the database product. The default location for db type
# of cloudscape is /WebSphere/V6R0M0/AppServer/cloudscape/databases/WBIDB
#
dbLocation=/WebSphere/V6R0M0/AppServer/cloudscape/databases/WBIDB
```

Response file for standalone configuration using DB2 for z/OS - standAloneProfileDB2.rsp

The following is a sample of the default response file used to create a stand-alone configuration for WebSphere Process Server for z/OS that uses a DB2 for z/OS database. WebSphere Process Server for z/OS configurations that use DB2 for z/OS have prerequisite tasks associated with configuring the database and the DB2 subsystem. For more information on how to prepare for a WebSphere Process Server for z/OS configuration that uses DB2 for z/OS, see Considerations for creating the database and Creating database and storage groups.

This response file supports DB2 for z/OS version 7 and DB2 for z/OS version 8. The default properties assume a DB2 for z/OS version 8. If you have DB2 for z/OS version 7 installed, you will need to modify the default values in this response file to support DB2 for z/OS version 7.

The response file contains comments on those properties that apply to DB2 for z/OS version 7 specifically.

Note: The sample response file below contains properties for Business Process Choreographer and WebSphere Business Integration Server. Business Process Choreographer and WebSphere Business Integration Server are not configured for and ESB-only server configuration of WebSphere Process Server for z/OS. These properties are ignored by the configuration processing.

```
########################################################################
# Licensed Materials - Property of IBM
# 5655-N53, 5655-R15
# (C) Copyright IBM Corporation 2006, 2007. All Rights Reserved.
# US Government Users Restricted Rights - Use, duplication or disclosure
# restricted by GSA ADP Schedule Contract with IBM Corp.
########################################################################
########################################################################
#
# Sample Response file for creating a WebSphere Process Server for z/OS
# stand-alone configuration using a DB2 for z/OS database.
#
########################################################################
########################################################################
# GLOBAL PROPERTIES
#
# Global properties prepended by the symbol $ may used as substitution
# for other values in this file. The global property value must be the
# complete value to be substituted.
#
# See examples of $JMSUSER used in this file.
#
#  Note: Global Properties must display in the response file before they
#  are referenced.
#
# For a description of how to use Global properties, see the section on
# "Reading the response file" in Working with response files topic in the
# information center.#

#
########################################################################
JMSUSER=ibmuser
JMSPASS=ibmuser
CONFIGSERVER=server1
########################################################################
# The following Global Properties are used to set values for databases
# used by WebSphere Process Server for z/OS components.
#
# The default values assigned are used to set the corresponding value
# for all the component databases.
#
# If the same database product and version is used for all components,
# then most component values can be set by simply using the value of
# the Global Property.
#
# Note that some properties, such as dbName, must be unique for each
# component and therefore cannot be set with a Global Property.
#
# If database products or versions differ for components, then
# component-specific values must be set individually, and the Global
# Properties can not be used.
#
# See the individual components for property descriptions. Note that
# some components do not use all the database Global Properties.
#
# For information on databases used by WebSphere Process Server
# for z/OS components, see "Database specifications" in the information
# center.
# For information on databases used by WebSphere Process Server
# for z/OS components, see Database specifications in the information
# center.
#
# The default values of the Global Properties in this response file
# are for DB2 for z/OS version 8.
#
# For DB2 version 7, change the value of Global Property DBPRODUCT to
# DB2UDBOS390_V7_1 and set the other the other Global Properties as
# appropriate.
#
########################################################################
DBPRODUCT=DB2UDBOS390_V8_1
DBLOCATION=LOC1
DBHOME=/db2810/jcc
DBPROPERTIES=/db2810/jcc/properties
DBJDBCCLASSPATH=/db2810/jcc/classes
DBUSER=wsadmin
DBPASS=wsadmin
DBHOSTNAME=localhost
DBSERVERPORT=446

########################################################################
#
# Common Properties
#
########################################################################
#
# Augment
#
# The "augment" keyword is required in order for the configuration to
# invoke the profile augmentation actions.
#
# There is no value associated with this keyword. The "augment" keyword
# may be replaced with "unaugment" in order to undo the results from a
# previous augment action. However, "augment" must be specified for the
# configuration to complete successfully.
#
#  Note: The uninstall command also references the response file.
#  However, you do not need to change the value from "augment" to
#  "unaugment" when you run the zSMPInstall.sh to uninstall the
#  product. For more information on uninstalling the product, see
#  the topic "Run the install script to uninstall the product" in the
#  information center.

#
augment

########################################################################
#
# Profile name
#
# On z/OS, there is always one and only one profile and that profile is
# named "default" in each of the configurations.
#
# The profile referred to here is the default profile installed and
# and configured during the WebSphere Application Server for z/OS
# installation.
#
# As "default" is the standard name for the profile, accept the default
# value for this property.
#
profileName=default

########################################################################
#
# Profile path
#
#
# This value should reflect the default profile root directory for
# WebSphere Application Server for z/OS
#
# The server configuration for WebSphere Application Server for z/OS
# resides in a directory structure under a "profiles" directory. So, for
# a base stand-alone application server in the config_root, there will
# be a "profiles" directory, and in that directory there will be a
# "default" directory that contains all of the directories and files
# (such as commands, configuration files, and log files) that define
# the runtime environment for your server configuration.
#
```

```
# In most cases the WebSphere Application Server for z/OS installation
# uses the default value shown below.
#
# The /WebSphere/V6R0M0/AppServer portion of the profile path
# shown below should be the configuration HFS / app_server_root for your base
# appserver.
#
# If the default value was not used to install and configure
# WebSphere Application Server for z/OS, make sure you edit the
# value below to represent the profile path that was used for the
# WebSphere Application Server for z/OS installation.
#
#
profilePath=/WebSphere/V6R0M0/AppServer/profiles/default

########################################################################
#
# Template path
#
# This value determines which component augment actions are performed as
# part of the configuration process. Valid values are as follows:
#
# /WebSphere/V6R0M0/AppServer/profileTemplates/default.wbicore
# /WebSphere/V6R0M0/AppServer/profileTemplates/default.bfm
# /WebSphere/V6R0M0/AppServer/profileTemplates/default.wbiserver
#
# /WebSphere/V6R0M0/AppServer/profileTemplates/default.*
#
# If one of the first three values is specified, only the augmentation
# actions within that directory are performed. The fourth value is the
# default (.../default.*),which when specified invokes augment actions
# for all of the template paths.
#
# Caution: Changing the default value causes an incomplete configuration
# of WebSphere Process Server for z/OS and should only be done if there
# is a valid reason to do so.
#
# The /WebSphere/V6R0M0/AppServer portion of the template path
# shown below should be the configuration HFS / app_server_root for your base
# appserver.
#
templatePath=/WebSphere/V6R0M0/AppServer/profileTemplates/default.*

#
# Cell name
#
# This value determines the cell name within the WebSphere Application
# Server for which this Process Server configuration applies. The
# default value is SY1.

#
# The value assigned should be the LONGNAME of the cell that you
# configured for your stand-alone appserver.
#
cellName=SY1

########################################################################
#
# Node name
#
# This value determines the node name within the WebSphere Application
# Server for which this Process Server configuration applies. The
# default value is SY1.

#
# Node name under one cell has to be unique. The default value is SY1.
#
# The value assigned should be the LONGNAME of the node that you
# configured for your stand-alone appserver.
#
nodeName=SY1

########################################################################
#
# Server name
#
# This value determines the server name within the WebSphere Application
# Server for which this Process Server configuration applies. The
# default value is server1.
```

```
#
# The value assigned should be the LONGNAME of the server that you
# configured for your stand-alone appserver.
#
serverName=$CONFIGSERVER

########################################################################
#
# Create new or use existing database
#
# For a configuration using DB2 for z/OS, the database will always be
# an existing database. The option to create a new database as part of
# the configuration process does not apply when the database is DB2
# for z/OS.
#
# As a prerequisite to creating a stand-alone configuration that uses
# DB2 for z/OS, you must create the databases and their associated
# storage groups.
#
# Because of this prerequisite, the value of this property must indicate
# that you are using an existing database.
#
# Accept the default value of false to indicate that you are not
# creating a new database.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# Accept the default value.
#
dbCreateNew=false

########################################################################
#
# Delay Configuration of the database table and data source definitions
#
# If the database and storage groups were defined as described in "Creating
# the databases and storage groups", the configuration process can connect
# to the database and define the tables and tablespaces automatically.
#
# Specify "true" to prevent the configuration process from running
# the scripts that will create the data source and database table
# definitions automatically.
#
#  Note: Delaying automatic database configuration will require you to
#  run the scripts to create the data source and database table
#  definitions manually. You can do this after you have run the
#  configuration script successfully. For more information, see the
#  topic "Configuring the databases manually" in the information
#  center

#
# Specify "false" to allow the configuration process to run the scripts
# that will create the data source and database tables definitions
# automatically.
#
#
# The default value is "false".
#
#  Note: Accepting the default value allows the configuration process
#  to connect to the database and create the definitions for the data
#  sources and database tables automatically. Always consult with your
#  Database administrator with regard to updating the database
#  automatically.
#
dbDelayConfig=false

########################################################################
#
# WBI Core Properties
#
########################################################################
#
# Common Event Infrastructure Configuration
#
# The following entries are used to create a sample Common Event
# Infrastructure(CEI) configuration. ceiSampleJmsUser, ceiSampleJmsPwd
# and ceiSampleServerName are required properties.
#
# The userids/passwords in this response file are used to create J2EE
# authentication aliases to access various resources that will be
# created as part of the augmentation actions invoked by the
# configuration script.

#########################################################################
#
# Java messaging service (JMS) userid for the CEI sample configuration
#
# Specifies the value of the Java Messaging Service (JMS) userid for a
# Common Event Infrastructure (CEI) configuration. Default value is
# ibmuser.
#
ceiSampleJmsUser=$JMSUSER

#########################################################################
#
# Java messaging service (JMS) password for the CEI sample configuration
#
# Specifies the value of the Java Messaging Service (JMS) password for a
# Common Event Infrastructure (CEI) configuration. Default value is
# ibmuser.
#
ceiSampleJmsPwd=$JMSPASS

#########################################################################
#
# Server name for the CEI sample configuration
#
# Specifies the value of the server name for the Common Event
# Infrastructure (CEI) sample configuration. The default value is
# server1.
#
# The value assigned should be the LONGNAME of the server that you
# configured for your stand-alone appserver. The Common Event
# Infrastructure component will be configured on the server.
#
ceiSampleServerName=$CONFIGSERVER

#########################################################################
#
# Database name for Common Event Infrastructure
#
# The property for the Common Event Infrastructure database name
# should NOT be changed from the default value CEIDB.
#
# The configuration process defines two databases named EVENT and
# EVENTCAT respectively. These database names should also be set
# when you create the database and storage groups. These databases
# will hold the definitions for the CEI database tables and data
# sources.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

ceiDbName=CEIDB

#########################################################################
#
# Database Storage Group for Common Event Infrastructure
#
# Specify the Database Storage Group name for the configuration of the
# Common Event Infrastructure database. This value will be substituted in
# all CEI generated DDL/SQL definitions.
#
# This value must match the storage group name set when you created the
# database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is EVTSTO.
#
```

```
ceiDbStorageGroup=EVTSTO

########################################################################
#
# Database SQL ID for Common Event Infrastructure
#
# The following Common Event Infrastructure SQL ID is for future use
# only and should not be changed from the default value WSADMIN.
#
ceiDbSqlId=WSADMIN

########################################################################
#
# Database product name for Common Event Infrastructure
#
# Specifies the default value of the database product name for Common
# Event Infrastructure (CEI) processing. The default value is
# DB2UDBOS390_V8_1.
#
# Valid Values are:
#
#  DB2UDBOS390_V8_1  for DB2 Version 8
#  DB2UDBOS390_V7_1  for DB2 Version 7
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
ceiDbProduct=$DBPRODUCT

########################################################################
#
# Database Connection Location for Common Event Infrastructure
#
# Specifies the database location for Common Event Infrastructure
# database. This value will be used in place of the database name on the
# WebSphere data source definition.
#
# The default value is LOC1.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
ceiDbConnectionLocation=$DBLOCATION

########################################################################
#
# Database Home Directory for Common Event Infrastructure
#
# Specify the Database Home JCC Directory. This value will vary
# depending on the  version of DB2 being used and where the database
# is installed.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
# The default value is /db2810/jcc.
#
ceiDbHome=$DBHOME

########################################################################
#
# Database DB2 JCC Properties Directory for Common Event Infrastructure
#
# Specify the directory that contains the DB2JccConfiguration.properties
# file. This file will be required by systems containing multiple DB2
# instances or using DB2 DataSharing.
#
# This value will vary depending on the version of DB2 being used and
# where it is installed. The default value is /db2810/jcc/properties.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
ceiDbJDBCProperties=$DBPROPERTIES

########################################################################
#
# Database JDBC Driver Path for Common Event Infrastructure
#
```

```
# Specify the JDBC Driver Path for Common Event Infrastructure database.
# This will vary depending on the version of DB2 being used and where it
# is installed. The default value is /db2810/jcc/classes.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
#
ceiDbJDBCClasspath=$DBJDBCCLASSPATH

########################################################################
#
# Database User ID for Common Event Infrastructure
#
# Specify the User ID for the configuration of the Common Event
# Infrastructure database. The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
ceiDbUser=$DBUSER

########################################################################
#
# Database Password for Common Event Infrastructure
#
# Specify the Password for the configuration of the Common Event
# Infrastructure database. The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
ceiDbPwd=$DBPASS

########################################################################
#
# Database Host Name for Common Event Infrastructure
#
# This property is used for defining a type 4 DB2 data source and will
# only be used for annotating the WebSphere Data source. The
# configuration process creates a type2 provider only.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
# The default value is localhost.
#
ceiDbHostName=$DBHOSTNAME

#######################################################################
#
# Database Port Number for Common Event Infrastructure
#
# This property is used for defining a type 4 DB2 Data source and is
# used for annotating the WebSphere Data source only. This value is not
# required.
#
# Configuration processing will only create a type2 provider. The
# default value is 446.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
ceiDbServerPort=$DBSERVERPORT

########################################################################
#
# Configure SCA
#
########################################################################
#
# The following entries are used to create a Service Component
# Architecture sample configuration.
#
########################################################################
#
# Configure SCA security
#
# If you do not want to create a Service Component Architecture sample
# configuration, change the default value from true to false and
# remove the default values assigned to the properties below.
#
# Accepting the default value of true will set up an authentication
# alias for your SIBUS if you turn on security at some point.
#
configureScaSecurity=true

########################################################################
#
# SCA security user id
#
# If you chose to create a Service Component Architecture sample
# configuration, then specify the userid for that configuration.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
# The default value is ibmuser.
#
scaSecurityUserId=$JMSUSER

########################################################################
#
# SCA security password
#
# If you chose to create a Service Component Architecture sample
# configuration, then specify the password for SCA security user id for
# that configuration.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
# Default value is ibmuser.
#
scaSecurityPassword=$JMSPASS

########################################################################
#
# Enterprise Service Bus Properties
#
# The following entries are used to create an Enterprise Service Bus
# Repository sample configuration.
#
########################################################################
#
# Database name for Enterprise Server Bus (ESB)
#
# Specify the name of the Enterprise Service Bus database. This value
# will be substituted in the generated DDL/SQL produced during the
# configuration of ESB.
#
# This value must match the database name set when you created the
# database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is ESBDB.
#
esbDbName=ESBDB

########################################################################
#
# Database Storage Group for Enterprise Service Bus
#
# Specify the Database Storage Group name for the configuration of the
# Enterprise Service Bus database. This value will be substituted in the
# generated DDL/SQL produced during the configuration for ESB.
#
# This value must match the storage group name set when you created the
# database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.
```

```
#
# The default value is ESBDBSTO.
#
esbDbStorageGroup=ESBDBSTO

########################################################################
#
# Database SQL ID for Enterprise Service Bus
#
# Specify the Database SQL ID for the configuration of the Enterprise
# Service Bus database. This value will be substituted in the generated
# DDL/SQL produced during the configuration for ESB.
#
# The default value is ESBLOG.
#
esbDbSqlId=ESBLOG

########################################################################
#
# Database product name for Enterprise Service Bus
#
# Specify the value of the database product name for Enterprise Service
# Bus (ESB) processing.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
# The default value is DB2UDBOS390_V8_1.
#
# Valid Values are:
#
#   DB2UDBOS390_V8_1  for DB2 Version 8
#   DB2UDBOS390_V7_1  for DB2 Version 7
#
esbDbProduct=$DBPRODUCT

########################################################################
#
# Database Connection Location for Enterprise Service Bus
#
# Specifies the Database Connection Location for the Enterprise
# Service Bus database.
#
# This value will be used in place of the database name on the
# Websphere data source definition.
#
# The default value is LOC1.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.

esbDbConnectionLocation=$DBLOCATION

########################################################################
#
# Database DB2 JCC Properties Directory for Enterprise Service Bus
#
# Specify the directory that contains the DB2JccConfiguration.properties
# file. This file will be required by systems containing multiple DB2
# instances or using DB2 DataSharing. This value will vary depending on
# the version of DB2 being used and where it is installed.
#
# The default value is /db2810/jcc/properties.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
esbDbJDBCProperties=$DBPROPERTIES

########################################################################
#
# Database Home Classes Directory for Enterprise Service Bus
#
# Specify the Database Home Classes Directory. This value will vary
# depending on the version of DB2 being used and where it is installed.
#
# The default value is /db2810/jcc/classes.
#
#
```

```
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.

esbDbJDBCClasspath=$DBJDBCCLASSPATH

########################################################################
#
# Database User ID for Enterprise Service Bus
#
# Specify the User ID for the WebSphere Process Server database.
# The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
esbDbUser=$DBUSER

########################################################################
#
# Database Password for Enterprise Service Bus
#
# Specify the Password for the WebSphere Process Server database.
# The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
esbDbPwd=$DBPASS

########################################################################
# Database Host Name for Enterprise Service Bus
#
# This property is used for defining a type 4 DB2 Data source and will
# only be used for annotating the WebSphere Data source.
#
# The configuration process creates a type2 provider only.
#
# The default value is localhost.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
esbDbHostName=$DBHOSTNAME

########################################################################
#
# Database Port Number for Enterprise Service Bus
#
# This property is used for defining a type 4 DB2 Data source and will
# only be used for annotating the WebSphere data source. This value is
# not required.
#
# The configuration process creates a type2 provider only.
#
# The default value is 446.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
esbDbServerPort=$DBSERVERPORT

########################################################################
#
# Business Process Choreographer Properties
#
########################################################################
#
# Set the properties below to create a sample Business Process
# Choreographer (BPC) configuration.
#
#
# If you do not wish to create a sample Business Process Choreographer
# configuration, remove the default values for bpcmqUser, bpcmqPwd and
# bpcadminGroups.
#
# The Business Process Choreographer sample is valid for a stand-alone
# server configuration only.  The values specified here for the Business
# Process Choreographer sample configuration are used to run the Business
# Process Choreographer configuration script (bpeconfig.jacl)
```

```
#
#
# If you choose not to create a sample Business Process Choreographer
# configuration at this time, you can run bpeconfig.jacl after running the
# product configuration script to create all of the Business Process
# Choreographer resources.
#
# Running bpeconfig.jacl is the recommended method for configuring the
# Business Process Choreographer resources used in WebSphere Process
# server for z/OS. For more information on running bpeconfig.jacl, see
# the topic "Using the bpeconfig.jacl script to configure Business
# Process Choreographer" in the information center.#

#
########################################################################
#
# MQ User Name for Process Choreographer
#
# Specify the user name for the configuration of resources for the
# Business Process Container using WebSphere MQ. The default value is
# ibmuser.
#
# This property is used in setting up an authentication alias for your
# Process Choreographer JMS resources that are created.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
bpcmqUser=$JMSUSER

#######################################################################
#
# MQ Password for Process Choreographer
#
# Specify the password for the MQ User Name for the configuration of
# resources for the Business Process Container using WebSphere MQ. The
# default value is ibmuser.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
bpcmqPwd=$JMSPASS

########################################################################
#
# Admin Users for Process Choreographer
#
# Specify the security role for the business process system
# administrator. This is the security role used by the Business Process
# Choreographer component to run a business process. The default value
# is ibmuser.
#
# This property should be a user from your SAF registry that you are
# going to associate with the business process administrator role.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
bpcadminUsers=$JMSUSER

########################################################################
#
# Database Name for Process Choreographer
#
# Specify the DB2 Database name used by the Business Process Container.
# This value will be substituted in the DDL/SQL produced by the
# configuration process for Business Process Choreographer.
#
# This value must match the database name set when you created the
# database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default is BPEDB.
#
bpcDbName=BPEDB
```

```
########################################################################
#
# Database Storage Group for Process Choreographer
#
# Specify the Database Storage Group name for the configuration of the
# Business Process Container. This value will be substituted in the
# DDL/SQL produced by the configuration of the Business Process
# Choreographer component.
#
# This value must match the storage group name set when you created the
# database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is BPEDBSTO.
#
bpcDbStorageGroup=BPEDBSTO

########################################################################
#
# Database SQL ID for Business Process Choreographer
#
# Specify the DB2 SQL ID used by the Business Process Container. This
# value will be substituted in the DDL/SQL produced by the configuration
# of the Business Process Choreographer component.
#
# The default is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
bpcDbSqlId=$DBUSER

########################################################################
#
# Database product name for Process Choreographer
#
# Specify the value of the database product name for Business Process
# Container.
#
# The default value is DB2UDBOS390_V8_1.
#
# Valid Values are:
#
#    DB2UDBOS390_V8_1  for DB2 Version 8
#    DB2UDBOS390_V7_1  for DB2 Version 7
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
bpcDbProduct=$DBPRODUCT

########################################################################
#
# Database Connection Location for Process Choreographer
#
# Specify the Database Location for the configuration of the Business
# Process Container. This value will used in place of the database name
# on the WebSphere data source definition. The default value is LOC1.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
#
bpcDbConnectionLocation=$DBLOCATION

########################################################################
#
# Database DB2 JCC Properties Directory for Process Choreographer
#
# Specify the directory that contains the DB2JccConfiguration.properties
# file. This file will be required by systems containing multiple DB2
# instances or using DB2 DataSharing. This value will vary depending on
# the version of DB2 being used and where DB2 is installed. The default
# value is /db2810/jcc/properties.
```

```
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
bpcDbJDBCProperties=$DBPROPERTIES

#########################################################################
#
# Database Home Classes Directory for Process Choreographer
#
# Specify the Database Home Classes Directory for Process Choreographer.
# This value will vary depending on version of DB2 being used and where
# DB2 is installed.
#
# The default value is /db2810/jcc/classes.
#
bpcDbJDBCClasspath=$DBJDBCCLASSPATH

########################################################################
#
# Database User ID for Process Choreographer
#
# Specify the User name for authenticating the JDBC resources for the
# Business Process Container. The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
bpcDbUser=$DBUSER

########################################################################
#
# Database Password for Process Choreographer
#
# Specify the Password for authenticating the JDBC resources for the
# Business Process Container. The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
bpcDbPwd=$DBPASS########################################################################
#
# WBI Server Properties
#
########################################################################
#
# Application Scheduler configuration
#
#
# Application Scheduler is used to schedule migration application group
# events. If you choose to configure Application Scheduler, set the
# property to "true". Otherwise, specify false.
#
# You must specify the server (into which the Application Scheduler
# application is installed) in the appSchedulerServer property below.
#
# For information on the Application Scheduler, see the topic
# "Application Scheduler" in the information center.

#
configureAppScheduler=true

########################################################################
#
# Indicate the application server name on this node if you chose to
# configure it for Application Scheduler use. Usually, for a stand-alone
# profile the server name is "server1". The default is value server1.
#
# The value assigned should be the LONGNAME of the server that you
# configured for your stand-alone appserver.
#
appSchedulerServer=$CONFIGSERVER

########################################################################
#
# Configure the Common Database for the process server
#
# Various WebSphere Process Server components use a common database
# connection. The following properties define the values for the Common
# database.
#
# For information on the components that share a common database, see
# the topic "Common database specifications" in the information center
#

#
#
########################################################################
#
# Database name
#
# Specify the name of the WebSphere Process Server for z/OS database. This value
# will be substituted in the generated DDL/SQL produced during the
# configuration for components sharing the common database.
#
# This value must match the database name set when you created the
# database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is WPRCSDB.
#
dbName=WPRCSDB

########################################################################
#
# Database Storage Group
#
# Specify the Storage Group name for the WebSphere Process Server
# database. This value will be substituted in the generated DDL/SQL
# produced during the configuration for components sharing the common
# database.
#
# This value must match the storage group name set when you created
# the database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is WPRCSSTO.
#
dbStorageGroup=WPRCSSTO

#########################################################################
#
# Database type
#
# Specify the type of the WebSphere Process Server database.
# The default value is DB2UDBOS390_V8_1.
#
# Valid values are:
#
#       DB2UDBOS390_V8_1  for DB2 Version 8
#       DB2UDBOS390_V7_1  for DB2 Version 7
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbType=$DBPRODUCT

########################################################################
#
# Database Connection Location
#
# Specify the Database Connection Location WebSphere Process Server.
# Specifies the default database location for WBI Server. This value
# will be used in place of the database name on the Websphere data
# source definition.
#
# The default value is LOC1.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
```

```
#
dbConnectionLocation=$DBLOCATION

########################################################################
#
# Database DB2 JCC Properties Directory
#
# Specify the directory that contains the DB2JccConfiguration.properties
# file.
#
# This file will be required by systems containing multiple DB2
# instances or using DB2 DataSharing. This value will vary depending
# on the version of DB2 being used and where DB2 is installed.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
# The default value is /db2810/jcc/properties.
#
dbJDBCProperties=$DBPROPERTIES

########################################################################
#
# Database Home Classes Directory
#
# Specify the Database Home Classes Directory. This value will vary
# depending on the version of DB2 being used and where DB2 is installed.
#
# The default value is /db2810/jcc/classes.
#
dbJDBCClasspath=$DBJDBCCLASSPATH

########################################################################
#
# Database User ID
#
# Specify the User ID for the WebSphere Process Server database.
# The default value is wsadmin.
#
dbUserId=$DBUSER

########################################################################
#
# Database Password
#
# Specify the Password for the WebSphere Process Server database.
# The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbPassword=$DBPASS

########################################################################
#
# Database Host Name
#
# This property is used for defining a type 4 DB2 data source and will
# only be used for annotating the WebSphere Data source.
#
# The configuration creates a type2 provider only.
#
# The default value is localhost.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbHostName=$DBHOSTNAME

########################################################################
#
# Database Port Number
#
# This property is used for defining a type 4 DB2 data source and will
# only be used for annotating the WebSphere data source. This value is
# not required.
#
# The configuration creates a type2 provider only.
#
```

```
# The default value is 446.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.

dbServerPort=$DBSERVERPORT
```

Response file for network deployment configuration - DmgrDB2.rsp

The following is a sample of the default response file used to create a deployment manager configuration for WebSphere Process Server for z/OS that uses a DB2 for z/OS database. WebSphere Process Server for z/OS configurations that use DB2 for z/OS have prerequisite tasks associated with configuring the database and the DB2 subsystem. For more information on how to prepare for a WebSphere Process Server for z/OS configuration that uses DB2 for z/OS, see Considerations for creating the database and Creating database and storage groups.

```
################################################################################
# Licensed Materials - Property of IBM
# 5655-N53, 5655-R15
# (C) Copyright IBM Corporation 2006, 2007. All Rights Reserved.
# US Government Users Restricted Rights - Use, duplication or disclosure
# restricted by GSA ADP Schedule Contract with IBM Corp.
################################################################################
################################################################################
#
# Sample Response file for creating a WebSphere Process Server for z/OS
# deployment manager configuration using DB2 for z/OS database.
#
################################################################################
# GLOBAL PROPERTIES
# Global properties prepended by the symbol $ may used as substitution for
# other values in this file. The global property value must be the complete
# value to be substituted.
#
# See examples of $JMSUSER used in this file.
#
# Note: Global Properties must physically appear in the response file before
#       they are referenced.
#
# For a description of how to use Global properties, see the section on
# "Reading the response file" in Working with response files topic in the
# information center.#

################################################################################
JMSUSER=ibmuser
JMSPASS=ibmuser
CONFIGSERVER=dmgr
################################################################################
# The following Global Properties are used to set values for databases
# used by WebSphere Process Server for z/OS components.
#
# The default values assigned are used to set the corresponding value
# for all the component databases.
#
# If the same database product and version is used for all components,
# then most component values can be set by simply using the value of
# the Global Property.
#
# Note that some properties, such as dbName, must be unique for each
# component and therefore cannot be set with a Global Property.
#
# If database products or versions differ for components, then
# component-specific values must be set individually, and the Global
# Properties can not be used.
#
# See the individual components for property descriptions. Note that
# some components do not use all the database Global Properties.
#
# For information on databases used by WebSphere Process Server
# for z/OS components, see "Database specifications" in the information
# center.
# For information on databases used by WebSphere Process Server
# for z/OS components, see Database specifications in the information
# center.
```

```
#
# The default values of the Global Properties in this response file
# are for DB2 for z/OS version 8.
#
# For DB2 version 7, change the value of Global Property DBPRODUCT to
# DB2UDBOS390_V7_1 and set the other the other Global Properties as
# appropriate.
#
################################################################################
DBPRODUCT=DB2UDBOS390_V8_1
DBLOCATION=LOC1
DBPROPERTIES=/db2810/jcc/properties
DBJDBCCLASSPATH=/db2810/jcc/classes
DBUSER=wsadmin
DBPASS=wsadmin
DBHOSTNAME=localhost
DBSERVERPORT=446
################################################################################
#
# Common Properties
#
################################################################################
# The "augment" keyword is required in order for the configuration to
# invoke the profile augmentation actions.
#
# There is no value associated with this keyword. The "augment" keyword
# may be replaced with "unaugment" in order to undo the results from a
# previous augment action. However, "augment" must be specified for the
# configuration to complete successfully.
#
#  Note: The uninstall command also references the response file.
#  However, you do not need to change the value from "augment" to
#  "unaugment" when you run the zSMPInstall.sh to uninstall the
#  product. For more information on uninstalling the product, see
#  the topic "Run the install script to uninstall the product" in the
#  information center.

#
augment

################################################################################
#
# Profile name
#
# On z/OS, there is always one and only one profile and that profile is
# named "default" in each of the configurations.
#
# The profile referred to here is the default profile installed and
# and configured during the WebSphere Application Server for z/OS
# installation.
#
# As "default" is the standard name for the profile, accept the default
# value for this property.
#
profileName=default

################################################################################
#
# Profile path
#
# This value should reflect the default profile root directory for
# WebSphere Application Server for z/OS
#
# The server configuration for WebSphere Application Server for z/OS
# resides in a directory structure under a "profiles" directory. So, for
# a base stand-alone application server in the config_root, there will
# be a "profiles" directory, and in that directory there will be a
# "default" directory that contains all of the directories and files
# (such as commands, configuration files, and log files) that define
# the runtime environment for your server configuration.
#
# In most cases the WebSphere Application Server for z/OS installation
# uses the default value shown below.
#
# The /WebSphere/V6R0M0/DeploymentManager portion of the profile path
# shown below should be the configuration HFS / app_server_root for your
# deployment manager.
#
profilePath=/WebSphere/V6R0M0/DeploymentManager/profiles/default
```

```
################################################################################
#
# Template path
#
# This value determines which component augment actions are performed as
# part of the configuration process. Valid values are as follows:
#
# /WebSphere/V6R0M0/DeploymentManager/profileTemplates/dmgr.wbicore
# /WebSphere/V6R0M0/DeploymentManager/profileTemplates/dmgr.bfm
# /WebSphere/V6R0M0/DeploymentManager/profileTemplates/dmgr.wbiserver
#
# /WebSphere/V6R0M0/DeploymentManager/profileTemplates/dmgr.*
#
# If one of the first three values is specified, only the actions within that
# directory are performed. The fourth value is the default (.../dmgr.*),
# which causes all of the actions in the first three template paths to be
# performed.
#
# Caution: Changing the default value causes an incomplete configuration of
# WebSphere Process Server for z/OS and should only be done if there is
# a valid reason to do so.
#
# The /WebSphere/V6R0M0/DeploymentManager portion of the template path
# shown below should be the configuration HFS / app_server_root for your
# base appserver.
#
templatePath=/WebSphere/V6R0M0/DeploymentManager/profileTemplates/dmgr.*
################################################################################
#
# Cell name
#
# This value determines the cell name within the WebSphere Application
# Server for which this Process Server configuration applies. The default value
# is v1cell.

#
# The value assigned should be the LONGNAME of the cell that you configured
# for your DeploymentManager appserver.
#
cellName=v1cell

################################################################################
#
# Node name
#
# This value determines the node name within the WebSphere Application
# Server for which this Process Server configuration applies.
#

#
# Node name under one cell has to be unique. The default value is v1dmnode.
#
# The value assigned should be the LONGNAME of the node that you configured
# for your DeploymentManager appserver.
#
nodeName=v1dmnode

################################################################################
#
# Server name
#
# This value determines the server name within the WebSphere Application
# Server for which this Process Server configuration applies.

# Server name under one node has to be unique. The default value is dmgr.
#
# The value assigned should be the LONGNAME of the server that you
# configured for your DeploymentManager appserver.
#
serverName=$CONFIGSERVER

################################################################################
#
# Create new or use existing database
#
# For a configuration using DB2 for z/OS, the database will always be
# an existing database. The option to create a new database as part of the
# configuration process does not apply when the database is DB2 for z/OS.
```

```
#
# As a prerequisite to creating a deployment manager configuration that
# uses DB2 for z/OS, you must create the databases and their associated
# storage groups.
#
# Because of this prerequisite, the value of this property must indicate
# that you are using an existing database.
#
# Accept the default value of false to indicate that you are not
# creating a new database.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# Accept the default value.
#
dbCreateNew=false

################################################################################
#
# Delay Configuration of the database table and datasource definitions
#
# If the database and storage groups were defined as described in "Creating
# the databases and storage groups, the configuration process can connect
# to the database and define the tables and tablespaces automatically.
#
# Specify "true" to prevent the configuration process from running
# the scripts that will create the data source and database table
# definitions automatically.
#
#  Note: Delaying automatic database configuration will require you to
#  run the scripts to create the data source and database table
#  definitions manually. You can do this after you have run the
#  configuration script successfully. For more information, see the
#  topic "Configuring the databases manually" in the information
#  center

#
# Specify "false" to allow the configuration process to run the scripts
# that will create the data source and database tables definitions
# automatically.
#
# Always consult your Database administrator with regard to updating DB2
# automatically.
#
# The default value is "true"
#
dbDelayConfig=true

################################################################################
#
# WBI Core Properties
#
################################################################################
#
# Configure SCA
#
################################################################################
#
# The following entries are used to create a Service Component Architecture
# sample configuration.
#
################################################################################
#
# Configure SCA security
#
# If you do not want to create a Service Component Architecture
# sample configuration, change the default value from true to false and
# leave the values for the properties specified below blank.
#
# Accepting the default will set up an authentication alias for your SIBUS
# if you turn on security at some point.
#
configureScaSecurity=true

################################################################################
#
```

```
# SCA security user id
#
# If you chose to create a Service Component Architecture configuration above
# then specify the userid for that configuration. Default value is ibmuser.
#
scaSecurityUserId=$JMSUSER

#################################################################################
#
# SCA security password
#
# If you chose to create a Service Component Architecture configuration above
# then specify the password for SCA security user id for that configuration.
# Default value is ibmuser.
#
scaSecurityPassword=$JMSPASS

#################################################################################
#
# WBI Server Properties
#
#########################################################################
#
# Application Scheduler configuration
#
# Application Scheduler is used to schedule migration application group
# events. If you choose to configure Application Scheduler, set the
# property to "true". Otherwise, specify false.
#
# You must specify the server (into which # the Application Scheduler
# application is installed) in the appSchedulerServer property below.
#
# For information on the Application Scheduler, see the topic
# "Application Scheduler" in the information center.

#
configureAppScheduler=false

#########################################################################
#
# Indicate the application server name on this node if you chose to
# configure it for Application Scheduler use. Usually, for a
# DeploymentManager profile the server name is "dmgr".
#
# The default is value dmgr.
#
# The value assigned should be the LONGNAME of the server that you
# configured for your DeploymentManager appserver.
#
appSchedulerServer=$CONFIGSERVER#################################################################################
#
# Configure the Common Database for the process server
# Various WebSphere Process Server components use a common database
# connection. The following properties define the values for the Common
# database.
#
# For information on the components that share a common database, see
# the topic "Common database specifications" in the information center
#

#
#################################################################################
#
# Database name
#
# Specify the name of the WebSphere Process Server database. This value
# will be substituted in the generated DDL/SQL produced during the
# configuration for components sharing the common database.
#
# This value must match the database name set when you created the
# database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is WPRCSDB.
#
```

```
dbName=WPRCSDB

################################################################################
#
# Database Storage Group
#
# Specify the Storage Group name for the WebSphere Process Server database.
# This value will be substituted in the generated DDL/SQL produced during the
# configuration for WPS.
#
# This value must match Storage Group name set When you created the database
# and storage group.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is WPRCSSTO.
#
dbStorageGroup=WPRCSSTO

################################################################################
# Specify the type of the WebSphere Process Server database.
# The default value is DB2UDBOS390_V8_1.
#
# Valid values are:
#
#       DB2UDBOS390_V8_1  for DB2 Version 8
#       DB2UDBOS390_V7_1  for DB2 Version 7
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbType=$DBPRODUCT

################################################################################
#
# Database Connection Location
#
# Specify the Database Connection Location WebSphere Process Server.
# Specifies the default database location for WBI Server. This value
# will be used in place of the database name on the Websphere data
# source definition.
#
# The default value is LOC1.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
#dbConnectionLocation=$DBLOCATION

################################################################################
#
# Database DB2 JCC Properties Directory
#
# Specify the directory that contains the DB2JccConfiguration.properties
# file.
#
# This file will be required by systems containing multiple DB2
# instances or using DB2 DataSharing. This value will vary depending
# on the version of DB2 being used and where DB2 is installed.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
# The default value is /db2810/jcc/properties.
#
#
dbJDBCProperties=$DBPROPERTIES

################################################################################
#
# Database Home Classes Directory
#
# Specify the Database Home Classes Directory. This value will vary
# depending on the version of DB2 being used and where DB2 is installed.
#
# The default value is /db2810/jcc/classes.
```

```
#
dbJDBCClasspath=$DBJDBCCLASSPATH

################################################################################
#
# Database User ID
#
# Specify the User ID for the WebSphere Process Server database.
# The default value is wsadmin.
#
dbUserId=$DBUSER

################################################################################
#
# Database Password
#
# Specify the Password for the WebSphere Process Server database.
# The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbPassword=$DBPASS

################################################################################
#
# Database Host Name
#
# This property is used for defining a type 4 DB2 data source and will
# only be used for annotating the WebSphere Data source.
#
# The configuration creates a type2 provider only.
#
# The default value is localhost.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbHostName=$DBHOSTNAME

################################################################################
#
# Database Port Number
#
# This property is used for defining a type 4 DB2 data source and will
# only be used for annotating the WebSphere data source. This value is
# not required.
#
# The configuration creates a type2 provider only.
#
# The default value is 446.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbServerPort=$DBSERVERPORT
```

Response file for managed node in a network deployment configuration - ManagedDB2.rsp

The following is a sample of the default response file used to create a managed node configuration for WebSphere Process Server for z/OS that uses a DB2 for z/OS database. WebSphere Process Server for z/OS configurations that use DB2 for z/OS have prerequisite tasks associated with configuring the database and the DB2 subsystem. For more information on how to prepare for a WebSphere Process Server for z/OS configuration that uses DB2 for z/OS, see Considerations for creating the database and Creating database and storage groups.

```
################################################################################
# Licensed Materials - Property of IBM
# 5655-N53, 5655-R15
# (C) Copyright IBM Corporation 2006, 2007. All Rights Reserved.
# US Government Users Restricted Rights - Use, duplication or disclosure
# restricted by GSA ADP Schedule Contract with IBM Corp.
################################################################################
```

```
################################################################################
#
# Sample Response file for creating a WebSphere Process Server for z/OS
# managed node configuration using DB2 for z/OS database.
#
################################################################################
################################################################################
# GLOBAL PROPERTIES
# Global properties prepended by the symbol $ may used as substitution for
# other values in this file. The global property value must be the complete
# value to be substituted.
#
# See examples of $JMSUSER used in this file.
#
# Note: Global Properties must physically appear in the response file before
#       they are referenced.
#
# For a description of how to use Global properties, see the section on
# "Reading the response file" in Working with response files topic in the
# information center.#

################################################################################
JMSUSER=ibmuser
JMSPASS=ibmuser
################################################################################
# The following Global Properties are used to set values for databases
# used by WebSphere Process Server for z/OS components.
#
# The default values assigned are used to set the corresponding value
# for all the component databases.
#
# If the same database product and version is used for all components,
# then most component values can be set by simply using the value of
# the Global Property.
#
# Note that some properties, such as dbName, must be unique for each
# component and therefore cannot be set with a Global Property.
#
# If database products or versions differ for components, then
# component-specific values must be set individually, and the Global
# Properties can not be used.
#
# See the individual components for property descriptions. Note that
# some components do not use all the database Global Properties.
#
# For information on databases used by WebSphere Process Server
# for z/OS components, see "Database specifications" in the information
# center.
# For information on databases used by WebSphere Process Server
# for z/OS components, see Database specifications in the information
# center.
#
# The default values of the Global Properties in this response file
# are for DB2 for z/OS version 8.
#
# For DB2 version 7, change the value of Global Property DBPRODUCT to
# DB2UDBOS390_V7_1 and set the other the other Global Properties as
# appropriate.
#
################################################################################
DBPRODUCT=DB2UDBOS390_V8_1
DBLOCATION=LOC1
DBPROPERTIES=/db2810/jcc/properties
DBJDBCCLASSPATH=/db2810/jcc/classes
DBUSER=wsadmin
DBPASS=wsadmin
DBHOSTNAME=localhost
DBSERVERPORT=446
################################################################################
#
# Common Properties
#
################################################################################
# The "augment" keyword is required in order for the configuration to
# invoke the profile augmentation actions.
#
# There is no value associated with this keyword. The "augment" keyword
# may be replaced with "unaugment" in order to undo the results from a
# previous augment action. However, "augment" must be specified for the
# configuration to complete successfully.
#
#  Note: The uninstall command also references the response file.
#  However, you do not need to change the value from "augment" to
#  "unaugment" when you run the zSMPInstall.sh to uninstall the
#  product. For more information on uninstalling the product, see
#  the topic "Run the install script to uninstall the product" in the
#  information center.

#
augment

################################################################################
#
# Profile name
#
# On z/OS, there is always one and only one profile and that profile is
# named "default" in each of the configurations.
#
# The profile referred to here is the default profile installed and
# and configured during the WebSphere Application Server for z/OS
# installation.
#
# As "default" is the standard name for the profile, accept the default
# value for this property.
#
profileName=default

################################################################################
#
# Profile path
#
# This value should reflect the default profile root directory for
# WebSphere Application Server for z/OS
#
# The server configuration for WebSphere Application Server for z/OS
# resides in a directory structure under a "profiles" directory. So, for
# a base stand-alone application server in the config_root, there will
# be a "profiles" directory, and in that directory there will be a
# "default" directory that contains all of the directories and files
# (such as commands, configuration files, and log files) that define
# the runtime environment for your server configuration.
#
# In most cases the WebSphere Application Server for z/OS installation
# uses the default value shown below.
#
# The /WebSphere/V6R0M0/AppServer portion of the profile path
# shown below should be the configuration HFS / app_server_root for your
# base application server.
#
profilePath=/WebSphere/V6R0M0/AppServer/profiles/default

################################################################################
#
# Template path
#
# This value determines which component augment actions are performed as
# part of the configuration process. Valid values are as follows:
#
# /WebSphere/V6R0M0/AppServer/profileTemplates/managed.wbicore
# /WebSphere/V6R0M0/AppServer/profileTemplates/managed.bfm
# /WebSphere/V6R0M0/AppServer/profileTemplates/managed.wbiserver
#
# /WebSphere/V6R0M0/DeploymentManager/profileTemplates/managed.*
#
# If one of the first three values is specified, only the actions within that
# directory are performed. The fourth value is the default (.../managed.*),
# which causes all of the actions in the first three template paths to be
# performed.
#
# Caution: Changing the default value causes an incomplete configuration of
# WebSphere Process Server for z/OS and should only be done if there is
# a valid reason to do so.
#
# The /WebSphere/V6R0M0/AppServer portion of the template path
# shown below should be the configuration HFS / app_server_root for your
# base appserver.
#
templatePath=/WebSphere/V6R0M0/AppServer/profileTemplates/managed.*
```

```
################################################################################
#
# Cell name
#
# This value determines the cell name within the WebSphere Application
# Server for which this Process Server configuration applies. The default value
# is v1cellx.

#
# The value assigned should be the LONGNAME of the cell associated with the
# unfederated empty managed node
#
cellName=v1cellx

################################################################################
#
# Node name
#
# This value determines the node name within the WebSphere Application
# Server for which this Process Server configuration applies.
#

#
# Node name under one cell has to be unique. The default value is v1dmnodea.
#
# The value assigned should be the LONGNAME of the node that you configured
# for your managed appserver.
#
nodeName=v1dmnodea

################################################################################
#
# Create new or use existing database
#
# For a configuration using DB2 for z/OS, the database will always be
# an existing database. The option to create a new database as part of the
# configuration process does not apply when the database is DB2 for z/OS.
#
# As a prerequisite you must create the databases and their associated
# storage groups.
#
# Because of this prerequisite, the value of this property must indicate
# that you are using an existing database.
#
# Accept the default value of false to indicate that you are not
# creating a new database.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# Accept the default value.
#
dbCreateNew=false

################################################################################
#
# Delay Configuration of the database table and datasource definitions
#
# If the database and storage groups were defined as described in "Creating
# the databases and storage groups", the configuration process can connect
# to the database and define the tables and tablespaces automatically.
#
# Specify "true" to prevent the configuration process from running
# the scripts that will create the data source and database table
# definitions automatically.
#
#  Note: Delaying automatic database configuration will require you to
#  run the scripts to create the data source and database table
#  definitions manually. You can do this after you have run the
#  configuration script successfully. For more information, see the
#  topic "Configuring the databases manually" in the information
#  center

#
# Specify "false" to allow the configuration process to run the scripts
# that will create the data source and database tables definitions
# automatically.
```

```
#
# Always consult your Database administrator with regard to updating DB2
# automatically.
#
# The default value is "true"
#
dbDelayConfig=true

################################################################################
#
# WBI Core Properties
#
################################################################################
#
# Configure SCA
#
################################################################################
#
# The following entries are used to create a Service Component Architecture
# sample configuration.
#
################################################################################
#
# Configure SCA security
#
# If you do not want to create a Service Component Architecture
# sample configuration, change the default value from true to false and
# leave the values for the properties specified below blank.
#
# Accepting the default will set up an authentication alias for your SIBUS
# if you turn on security at some point.
#
configureScaSecurity=true

################################################################################
#
# SCA security user id
#
# If you chose to create a Service Component Architecture configuration above
# then specify the userid for that configuration. Default value is ibmuser.
#
scaSecurityUserId=$JMSUSER

################################################################################
#
# SCA security password
#
# If you chose to create a Service Component Architecture configuration above
# then specify the password for SCA security user id for that configuration.
# Default value is ibmuser.
#
scaSecurityPassword=$JMSPASS

##############################################################################
#
# WBI Server Properties
#
########################################################################
#
# Application Scheduler configuration
#
# Application Scheduler is used to schedule migration application group
# events. If you choose to configure Application Scheduler, set the
# property to "true". Otherwise, specify false.
#
# You must specify the server (into which the Application Scheduler
# application is installed) in the appSchedulerServer property below.
#
# For information on the Application Scheduler, see the topic
# "Application Scheduler" in the information center.

#
configureAppScheduler=false################################################################################
#
# Configure the Common Database for the process server
# Various WebSphere Process Server components use a common database
# connection. The following properties define the values for the Common
# database.
#
```

```
# For information on the components that share a common database, see
# the topic "Common database specifications" in the information center
#

#
################################################################################
#
# Database name
#
# Specify the name of the WebSphere Process Server database. This value
# will be substituted in the generated DDL/SQL produced during the
# configuration for components sharing the common database.
#
# This value must match the database name set when you created the
# database and storage groups.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is WPRCSDB.
#
dbName=WPRCSDB

################################################################################
#
# Database Storage Group
#
# Specify the Storage Group name for the WebSphere Process Server database.
# This value will be substituted in the generated DDL/SQL produced during the
# configuration for WPS.
#
# This value must match Storage Group name set When you created the database
# and storage group.
#
# See the "Creating the databases and storage groups" topic in the
# information center for information on how to create the databases
# for the various components.

#
# The default value is WPRCSSTO.
#
dbStorageGroup=WPRCSSTO

################################################################################
# Specify the type of the WebSphere Process Server database.
# The default value is DB2UDBOS390_V8_1.
#
# Valid values are:
#
#       DB2UDBOS390_V8_1  for DB2 Version 8
#       DB2UDBOS390_V7_1  for DB2 Version 7
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbType=$DBPRODUCT

################################################################################
#
# Database Connection Location
#
# Specify the Database Connection Location WebSphere Process Server.
# Specifies the default database location for WBI Server. This value
# will be used in place of the database name on the Websphere data
# source definition.
#
# The default value is LOC1.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
#dbConnectionLocation=$DBLOCATION

################################################################################
#
# Database DB2 JCC Properties Directory
#
```

```
# Specify the directory that contains the DB2JccConfiguration.properties
# file.
#
# This file will be required by systems containing multiple DB2
# instances or using DB2 DataSharing. This value will vary depending
# on the version of DB2 being used and where DB2 is installed.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
# The default value is /db2810/jcc/properties.
#
#
dbJDBCProperties=$DBPROPERTIES

################################################################################
#
# Database Home Classes Directory
#
# Specify the Database Home Classes Directory. This value will vary
# depending on the version of DB2 being used and where DB2 is installed.
#
# The default value is /db2810/jcc/classes.
#
dbJDBCClasspath=$DBJDBCCLASSPATH

################################################################################
#
# Database User ID
#
# Specify the User ID for the WebSphere Process Server database.
# The default value is wsadmin.
#
dbUserId=$DBUSER

################################################################################
#
# Database Password
#
# Specify the Password for the WebSphere Process Server database.
# The default value is wsadmin.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbPassword=$DBPASS

################################################################################
#
# Database Host Name
#
# This property is used for defining a type 4 DB2 data source and will
# only be used for annotating the WebSphere Data source.
#
# The configuration creates a type2 provider only.
#
# The default value is localhost.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbHostName=$DBHOSTNAME

################################################################################
#
# Database Port Number
#
# This property is used for defining a type 4 DB2 data source and will
# only be used for annotating the WebSphere data source. This value is
# not required.
#
# The configuration creates a type2 provider only.
#
# The default value is 446.
#
# See the Global Properties section at the beginning of this response
# file for information on setting values for global properties.
#
dbServerPort=$DBSERVERPORT
```

Modifying a response file

WebSphere Process Server for z/OS is packaged with sample response files that include parameters used to create the WebSphere Process Server for z/OS runtime environment. Before running the configuration script you must modify the content of the response file in order to customize configuration to your environment.

Before you modify the sample response file, make sure of the following:

- Create the databases and storage groups for the WebSphere Process Server for z/OS components that utilize databases.

  See Creating the databases and storage groups information.
- Understand the various configuration types supported by WebSphere Process Server for z/OS.

  See Planning the configuration.
- Understand how the response file is organized and how to read and edit the response files.

  See Working with response files.
- This task assumes that you have access to the installed files and that you have knowledge of MVS system commands and tools for editing flat files.

You modify the property values in a default response file before running the WebSphere Process Server for z/OS configuration script.

1. Make a copy the desired response file from the HFS into which it was installed. Copy the file to a directory in which you have edit authority and save it with a new name such as **my_options_file.rsp**.
2. Edit the values in the response file to customize the profile property values for your WebSphere Process Server for z/OS configuration. The sample response file contains detailed comments that explain the properties. The commented response files are also documented in Sample response files.
3. Save the modified file to the directory in which it was copied. When you run the configuration script the command will include the path to the modified response file.

After modifying the response file you can run the product configuration script.

Creating a stand-alone configuration

Running **zWPSConfig.sh** or **zWESBConfig.sh** with the stand-alone response file creates a stand-alone configuration for WebSphere Process Server for z/OS. It does so by augmenting the WebSphere Application Server for z/OS default profile for a stand-alone application server with WebSphere Process Server for z/OS configuration data.

If you will be creating a WebSphere Process Server for z/OS stand-alone configuration that uses DB2 for z/OS, you should do the following:

- Decide a naming convention that will apply to all the databases, storage groups, tables, indexes and views that you will create in DB2.

  See Considerations for creating the database for information.
- Decide whether to execute the DB2 Data Definition Language (DDL) statements using z/OS tools like SPUFI or Unix tools like shell and ANT scripts.
- Decide on an approach to the security that will apply to the WebSphere Process Server for z/OS objects in DB2.

  Prepare the additional RACF groups and userids you will need to access DB2 and JMS queues.

See Considerations for creating the database for information.

- Verify that the DB2 Universal Driver has been configured for your DB2.

There are some useful hints and tips regarding database configuration in the technote titled, *WPS for z/OS Datasource & Database Configuration Hints & Tips*

Note: The *WPS for z/OS Datasource & Database Configuration Hints & Tips* technote applies to WebSphere Process Server for z/OS version 601, but some of the information regarding DB2 preparation may be useful.

The WebSphere Process Server for z/OS configuration process assumes a working knowledge of z/OS UNIX System Services. You may want to have access to the following documentation:

- z/OS V1R7.0 UNIX System Services User's Guide.
- z/OS V1R7.0 UNIX System Services Command Reference.

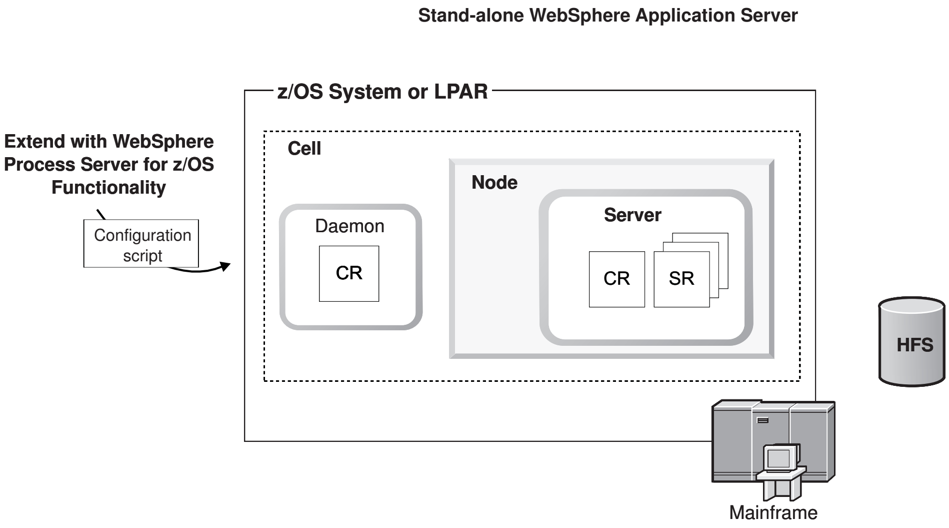
Before you run the configuration script make sure of the following:

1. Review "Planning to install and configure WebSphere Process Server for z/OS" on page 51 for an understanding of the task flow of the installation and configuration process.
2. That the systems programmer who is in charge of security and system access at your installation has set you up so that you can properly access the shell.

   This may involve making modifications to your RACF (security) profile and creating a home directory within the UNIX shell. The home directory is where you always start off when you begin a UNIX session. Within this directory, you keep any environmental variable files you may need to execute programs. These files contain information that's required in certain languages, like the location of the Java classes that are used in Java programs. You can also use the home directory as the root directory for keeping your work data. See Resource Access Control Facility Tools in the WebSphere Application Server for z/OS for information on tools and techniques to help you manage the security definitions used for WebSphere Application Server for z/OS.

   If you are creating a WebSphere Process Server for z/OS stand-alone configuration that uses DB2 for z/OS, see Considerations for creating the database for information regarding authorizations to create and access these databases and tables.
3. The WebSphere Application Server for z/OS product code has been loaded onto the system and you have configured a stand-alone application server.

   Because the WebSphere Process Server for z/OS configuration process augments the WebSphere Application Server for z/OS default profile, you need to have a configured stand-alone application server in place prior to augmenting its profile with WebSphere Process Server for z/OS configuration data:


Stand-alone WebSphere Application Server
z/OS System or LPAR
Extend with WebSphere Process Server for z/OS Functionality
Configuration script
Cell
Node
Daemon
CR
Server
CR
SR
HFS
Mainframe

Note: If you are creating a managed node configuration for WebSphere Process Server for z/OS, you must follow the steps for creating an empty cluster as described in the WebSphere Application Server for z/OS information center. As part of the post configuration process for creating a WebSphere Process Server for z/OS managed node configuration, you clone a stand-alone application server and you populate the ″serverless″ node that is created when you run the configuration script using the managed node response file.

4. The WebSphere Process Server for z/OS product code has been loaded onto the system.

   The system programmer has completed the SMP/E installation by successfully loading the contents of the WebSphere Process Server for z/OS tape onto the system.
5. Take a backup of the WebSphere Application Server configuration HFS.
6. The WebSphere Process Server for z/OS installation script has been run.

   Running the installation script creates the symbolic links from WebSphere Application Server configuration HFS to the WebSphere Process Server for z/OS SMP/E code and updates the administrative console with the WebSphere Process Server for z/OS definitions.

   Make sure when you run the installation script that the **-runtime** parameter value is the path to the configuration HFS of the WebSphere Application Server for z/OS stand-alone server. For example, `-runtime /WebSphere/V6R0/AppServer`.

   See “Running the installation script” on page 93 for more information.
7. If you are creating a WebSphere Process Server for z/OS stand-alone configuration that will use DB2, you need to Create the databases and storage groups required for your product configuration.

   See Creating the databases and storage groups for information.
8. Determine which response file that you want to use to create the WebSphere Process Server for z/OS stand-alone configuration.

   You can choose from the following two response files:
   - standAloneProfile.rsp

This sample response file contains parameters for creating and configuring WebSphere Process Server for z/OS stand-alone server that uses a Cloudscape database.

This sample standAloneProfile.rsp is located in <smpe_root>/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp .

- standAloneProfileDB2.rsp

  This sample response file contains parameters for creating and configuring WebSphere Process Server for z/OS stand-alone server that uses a DB2 for z/OS database.

  Note: Using DB2 for z/OS as your database for the WebSphere Process Server for z/OS stand-alone server will require some additional set up procedures. See“Considerations for creating the database” on page 133

See Choosing a response file for information.

9. Copy the stand-alone response file to your working directory.

   For example, to copy the stand-alone response file with the Cloudscape database parameters, perform the following:

   ```
   cp /usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp
   ```

   Where */usr/lpp/zWPS/V6R0/* represents the SMP_ROOT for WebSphere Process Server for z/OS.

10. Make sure that the response file has appropriate permissions.

    For stand-alone configuration using Cloudscape:

    ```
    chmod 755 standAloneProfile.rsp
    ```

    For stand-alone configuration using DB2 for z/OS:

    ```
    chmod 755 standAloneProfileDB2.rsp
    ```

11. Tailor the stand-alone response file to meet your environment specifications.

    See Working with response files for information on editing properties in the response file.

    For the standAloneProfileDB2.rsp, modify the following fields (always using the LONG names):

    ```
    JMSUSER=jmsuser-> The userids are used to create J2C authentication aliases for resources created.
    JMSPASS=passw0rd
    DBUSER=db2user
    DBPASS=passw0rd
    CONFIGSERVER=dmgr
    DBLOCATION=LOC1-> Needs to be DB2 LOCATION...not SSID
    profilePath=/WebSphere/V6R0/DeploymentManager/profiles/default
    templatePath=/WebSphere/V6R0/DeploymentManager/profileTemplates/dmgr.*
    cellName=cl1celld
    nodeName=cl1dmnod
    dbType=DB2UDBOS390_V8_1 dbJDBCClasspath=/usr/lpp/db2810/jcc/classes
    dbJDBCProperties=/etc/db2cfg
    ```

    Important: The properties listed above represent the properties you must customize for your stand-alone configuration. For all of the other properties in the sample response file, you can accept the default values (many of which use variable substitution). Depending on the database and storage groups created, you may also need to change the following:

    ```
    dbName=WPSDB
    dbStorageGroup=WPSDBSTO
    ```

    For information on creating the database and storage groups, see Creating the databases and storage groups.

    If you are using the standAloneProfileDB2.rsp response file, make sure that value for delaying database configuration is set to ″true″ **dbDelayConfig=true**.

Delaying database configuration means that the configuration script will not automatically run the DDL scripts to create the database resources. When **dbDelayConfig=true** the database definitions contained in the DDL are written to a directory and must be run manually after the configuration process. This value does not apply if you are using Cloudscape as your database. For Cloudscape the database definitions are run automatically. See "Configuring the database manually" on page 207 for information.

12. **Optional:** Create and customize a DB2JccConfiguration.properties file.

    The DB2JccConfiguration.properties files is used by the DB2 Universal JDBC driver when connecting to DB2 z/OS using a Type 2 datasource.

    To create and customize a DB2JccConfiguration.properties file, perform the following steps:

    a. Create a **DB2JccConfiguration.properties** file in /etc/db2cfg/ if your DB2 SSID does not equal DSN8.

    b. Make sure that you set the permissions on the directory so that the WebSphere Controller Region userid and the userid that will run your WebSphere Process Server for z/OS configuration jobs can read the properties file.

    c. The DB2JccConfiguration.properties file should have the following line in it and it should be readable to all:

       This properties file

       ```
       db2.jcc.ssid=DB15
       ```

       where **DB15** is your installation's SSID.

       For details of all the other possible properties you could code in the DB2JccConfiguration.properties files refer to the manual *DB2 Universal Database for z/OS V8.1 Application Programming Guide and Reference for Java*, SC18-741

Perform this task to create a WebSphere Process Server for z/OS stand-alone configuration.

1. Change directory to the application server bin directory.

   ```
   cd /WebSphere/V6R0/AppServer/bin/
   ```

2. If you have not already done so, switch to the administrator's user id. For example,

   ```
   su wsadmin
   ```

3. **Optional:** You may need to increase OMVS time limit to allow the product configuration script time to complete. To increase the OMVS time limit, type the following from the MVS console:

   ```
   SETOMVS MAXCPUTIME=86400
   ```

4. Access the OS/390 UNIX command shell. Enter the TSO command OMVS at the ISPF Command Shell or TSO OMVS from any other ISPF panel.

   Once you're in the UNIX shell, a command prompt (usually a dollar ($) or pound (#) sign) indicates that the system is ready to accept input.

5. From the command prompt, run the WebSphere Process Server for z/OS configuration command. Use the following syntax diagram as guide on how run the command:

   Syntax diagram for configuring WebSphere Process Server for z/OS

   ```
   ►►──zWPSConfig.sh──-augment──────────────────────────────────────────►
   ```

```
   (1)
►──────┬─-response─────────────────────────────────────────────┬──────────────►
       └─┬─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp────┘
         ├─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp─┤
         ├─/usr/lpp/zWPS/V6R0/zos.config/dmgrProfile.rsp──────────┤
         └─/usr/lpp/zWPS/V6R0/zos.config/managedProfile.rsp───────┘

►──-Z <Name of the value in the property file to override>.───────────────────►

                                  (2)
►──-trace──┬─trace arguments─┬──────────────────────────────────────────────►◄
           └─*=all=enabled───┘
```

Notes:

1 Type the absolute path of the response file that is associated with the product configuration you are creating. This example shows the default path to the response files packaged with the product. When you run the command, the **-response** parameter will represent the absolute path to the response file that you have modified with the values specific to your environment.

2 By including **-trace** the command writes to the appropriate trace file.

Syntax diagram for configuring WebSphere Process Server for z/OS as an ESB-only server

```
►►──zWESBConfig.sh──-augment──────────────────────────────────────────────────►

   (1)
►──────┬─-response─────────────────────────────────────────────┬──────────────►
       └─┬─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp────┘
         ├─/usr/lpp/zWPS/V6R0/zos.config/standAloneProfileDB2.rsp─┤
         ├─/usr/lpp/zWPS/V6R0/zos.config/dmgrProfile.rsp──────────┤
         └─/usr/lpp/zWPS/V6R0/zos.config/managedProfile.rsp───────┘

►──-Z <Name of the value in the property file to override>.───────────────────►

                                  (2)
►──-trace──┬─trace arguments─┬──────────────────────────────────────────────►◄
           └─*=all=enabled───┘
```

Notes:

1 Type the absolute path of the response file that is associated with the configuration onto which you are installing WebSphere Process Server for z/OS. This absolute pathname in this example assumes you are using the default response files. If you wanted to customize the response file, this pathname would need to represent the absolute path of the customized file.

2 By including **-trace** the command writes to the appropriate log file.

Note: The instructions described above indicate how to run the install script from a OS/390 UNIX command shell. Alternatively you can run the install script from a prompt using Telnet. To run the install script using

Telnet, type the following from the system prompt: **telnet TCPIP-Address port number** . Here is an example of a Telnet command: `telnet 99.9.99.999 2023`, where 99.9.99.999 is the TCP IP-Address and 2023 is the port number.

Review the messages from Standard Out. These are either displayed on the screen from which the **zWPSConfig.sh** or **zWESBConfig.sh** was run or in the file specified if the redirect **(″>″)** symbol was used on the command line.

There should be no error messages displayed.

An example of a successful execution of **zWPSConfig.sh** or **zWESBConfig.sh** with the -augment option is as follows:

```
parsing command arguments...
parsing arguments complete
setting up configuration...
runtimeRootDirName is: /WebSphere/V6R0M0/AppServer
WAS_HOME is: /WebSphere/V6R0M0/AppServer
WBI_HOME is: /WebSphere/V6R0M0/AppServer
set up configuration complete
augmenting profile(s)...
augmenting profile(s) complete
```

If you chose to delay the configuration of your databases, you will need to run the database definitions that were generated in the form of DDL scripts. These scripts reside in *configuration_root/app_server_root* /profiles/default/databases.

If you specified values for the Business Process Choreographer sample configuration in the response file, a sample configuration that includes the business process container, the human task container and the Business Process Choreographer Explorer are now part of your Process Server configuration. You can check to see if these components are configured by looking in the administrative console for enterprise applications with names that start with BPEContainer, BPCExplorer, and TaskContainer.

Note: The sample configuration is not suitable for a production system. Because you can only have one Business Process Choreographer configuration, you must remove the sample configuration, as described in as described in *Removing the Business Process Choreographer configuration* the Business Process Choreographer PDF, before you continue configuring Business Process Choreographer to use WebSphere MQ or a different database.

If you chose to delay the configuration of your databases, you will need to run the database definitions that were generated in the form of DDL scripts. These scripts reside in the WebSphere Application Server for z/OS configuration root *configuration_root/app_server_root*/profiles/default/dbscripts. For example, *WebSphere/V6R0/AppServer*//profiles/default/dbscripts.

For additional information, see Verifying the product configuration.

Create a Network Deployment configuration

Running **zWPSConfig.sh** or **zWESBConfig.sh** with the deployment manager response file creates a network deployment configuration for WebSphere Process Server for z/OS. It does so by running dozens of ant scripts to install applications, create resources, buses, etc, augmenting the WebSphere Application Server for z/OS default profile for a deployment manager cell with WebSphere Process

Server for z/OS configuration data. You can configure WebSphere Process Server for z/OS in a network deployment environment with managed server nodes across a sysplex.

Many DB2 databases and tables are needed for the WebSphere Process Server for z/OS configuration.

With regard to preparing DB2 for z/OS, you should perform the following:

- Decide a naming convention that will apply to all the databases, storage groups, tables, indexes and views that you will create in DB2.

  See Considerations for creating the database for information.

  A key point to be clear about is your naming convention and your choice of schema names and J2C authentication aliases.
- Decide whether to execute the DB2 Data Definition Language (DDL) statements using z/OS tools like SPUFI or Unix tools like shell and ANT scripts.
- Decide on an approach to the security that will apply to the WebSphere Process Server for z/OS objects in DB2.

  Prepare the additional RACF groups and userids you will need to access DB2 and JMS queues.

  See Considerations for creating the database for information.
- Verify that the DB2 Universal Driver has been configured for your DB2.

There are some useful hints and tips regarding database configuration in the technote titled, *WPS for z/OS Datasource & Database Configuration Hints & Tips*

Note: The *WPS for z/OS Datasource & Database Configuration Hints & Tips* technote applies to WebSphere Process Server for z/OS version 601, but some of the information regarding DB2 preparation may be useful.

The WebSphere Process Server for z/OS configuration process assumes a working knowledge of z/OS UNIX System Services. You may want to have access to the following documentation:

- z/OS V1R7.0 UNIX System Services User's Guide
- z/OS V1R7.0 UNIX System Services Command Reference

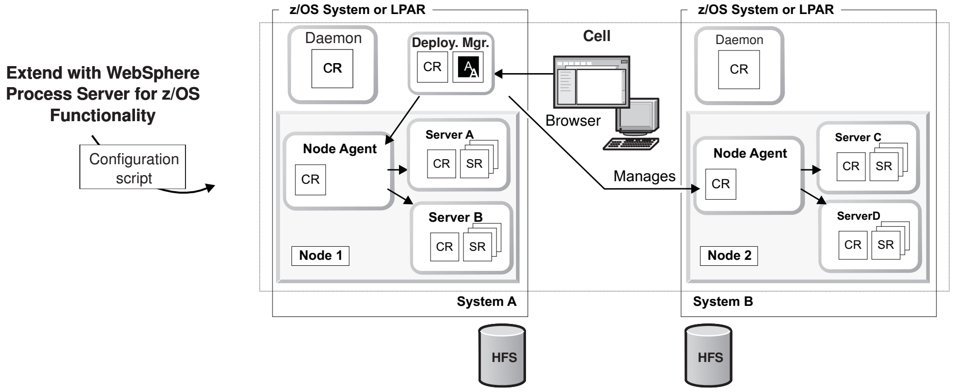
Before you create a deployment manager configuration, make sure of the following:

1. Review "Planning to install and configure WebSphere Process Server for z/OS" on page 51 for an understanding of the task flow of the installation and configuration process.
2. That the systems programmer who is in charge of security and system access at your installation has set you up so that you can properly access the shell.

   This involves making modifications to your RACF (security) profile and creating a home directory within the UNIX shell. The home directory is where you always start off when you begin a UNIX session. Within this directory, you keep any environmental variable files you may need to execute programs. These files contain information that's required in certain languages, like the location of the Java classes that are used in Java programs. You can also use the home directory as the main directory for keeping your work data. See Resource Access Control Facility Tools in the WebSphere Application Server for z/OS for information on tools and techniques to help you manage the security definitions used for WebSphere Application Server for z/OS.

3. The WebSphere Application Server for z/OS product code has been loaded onto the system and you have configured a network deployment cell and an empty managed node.

   Because the WebSphere Process Server for z/OS configuration process augments the WebSphere Application Server for z/OS default profile, you need to have a configured deployment manager cell in place prior to augmenting its profile with WebSphere Process Server for z/OS configuration data:

   Important: Make sure that when you create the empty managed node in WebSphere Application Server for z/OS, **that you do not run the job BBOWMNAN**. This job will federate your node into the specified Deployment Manager cell. You do not want to federate the node at this point, as you will federate the Deployment Manager cell as part of the WebSphere Process Server for z/OS configuration process.

4. The WebSphere Process Server for z/OS product code has been loaded onto the system.

   The systems programmer has completed the SMP/E installation by successfully loading the contents of the WebSphere Process Server for z/OS tape onto the system. See "Loading the product code from the installation media onto z/OS" on page 89 for more information.

5. The WebSphere Process Server for z/OS installation script has been run.

   Run the zSMPInstall.sh script against the Deployment Manager's node to prepare the cell for WebSphere Process Server for z/OS functionality.

   Running the installation script creates the symbolic links from the WebSphere Application Server for z/OS configuration root (sometimes referred to as WAS_HOME) to the WebSphere Process Server for z/OS SMP/E code and updates the administrative console with the WebSphere Process Server for z/OS definitions.

   Make sure when you run the installation script that the **-runtime** parameter value is the configuration root of WebSphere Application Server for z/OS deployment manager. For example, `-runtime /WebSphere/V6R0/DeploymentManager`.

   See "Running the installation script" on page 93 for more information.

6. Backup your WebSphere cell's configuration HFS dataset(s). Remember to backup both the Deployment Manager's configuration HFS and the empty node's configuration HFS.

7. Because the network deployment configuration supports DB2 for z/OS database only, make sure you complete the DB2 prerequisite actions before you run the WebSphere Process Server for z/OS configuration script.

   For more information, see DB2 for z/OS setup considerations in the section on planning to install and configure WebSphere Process Server for z/OS, as well as "Creating databases and storage groups" on page 135.

8. Copy the deployment manager response file to your working directory. For example,

   ```
   cp /usr/lpp/zWPS/V6R0/zos.config/DmgrDB2.rsp
   ```

9. Make sure that the response file has appropriate permissions:

   ```
   chmod 755 DmgrDB2.rsp
   ```

10. Tailor the deployment manager response file to meet your environment specifications.

    See Working with response files for information on editing properties in the response file.

    For the DmgrDB2.rsp, modify the following fields in the response file (always using the LONG names):

    ```
    JMSUSER=jmsuser---> The userids are used to create J2C authentication aliases for resources created.
    JMSPASS=passw0rd
    DBUSER=db2user
    DBPASS=passw0rd
    CONFIGSERVER=dmgr
    DBLOCATION=LOC1  ----> Needs to be DB2 LOCATION...not SSID
    profilePath=/WebSphere/V6R0/DeploymentManager/profiles/default
    templatePath=/WebSphere/V6R0/DeploymentManager/profileTemplates/dmgr.*
    cellName=cl1celld
    nodeName=cl1dmnod
    dbType=DB2UDBOS390_V8_1
    dbJDBCClasspath=/usr/lpp/db2810/jcc/classes
    dbJDBCProperties=/etc/db2cfg
    ```

    Important: The properties listed above represent the properties you must customize for your deployment manager configuration. For all of the other properties in the sample response file, you can accept the default values (many of which use variable substitution). Depending on the database and storage groups created, you may also need to change the following:

    ```
    dbName=WPSDB
    dbStorageGroup=WPSDBSTO
    ```

    For information on creating the database and storage groups, see Creating the databases and storage groups.

    Make sure that value for delaying database configuration is set to ″true″ **dbDelayConfig=true**. Delaying database configuration means that the configuration script will not automatically run the DDL scripts to create the database resources. When **dbDelayConfig=true** the database definitions contained in the DDL are written to a directory and must be run manually *after the configuration process*. See "Configuring the database manually" on page 207 for information.

11. **Optional:** Create and customize a DB2JccConfiguration.properties file.

    The DB2JccConfiguration.properties files is used by the DB2 Universal JDBC driver when connecting to DB2 z/OS using a Type 2 datasource.

    To create and customize a DB2JccConfiguration.properties file, perform the following steps:

    a. Create a **DB2JccConfiguration.properties** file in /etc/db2cfg/ if your DB2 SSID does not equal DSN8.

b. Make sure that you set the permissions on the directory so that the WebSphere Controller Region userid and the userid that will run your WebSphere Process Server for z/OS configuration jobs can read the properties file.

c. The DB2JccConfiguration.properties file should have the following line in it and it should be readable to all:

   This properties file

   ```
   db2.jcc.ssid=DB15
   ```

   where **DB15** is your installation's SSID.

   For details of all the other possible properties you could code in the DB2JccConfiguration.properties files refer to the manual *DB2 Universal Database for z/OS V8.1 Application Programming Guide and Reference for Java*, SC18-741

Perform this task to create a WebSphere Process Server for z/OS deployment manager configuration.

1. Stop the deployment manager server if need be. You can use the stopManager command to stop the deployment manager or you can use the administrative console to stop the deployment manager server.

   See Stopping an application server in the WebSphere Application Server for z/OS information center for detailed information on how to stop the server.

2. Prepare to run the configuration script. If you will run the configuration script from an OS/390 UNIX command shell, perform the following:

   a. Access the OS/390 UNIX command shell.

   b. Enter the TSO command OMVS at the ISPF Command Shell or TSO OMVS from any other ISPF panel.

   c. Once you're in the UNIX shell, a command prompt (usually a dollar ($) or pound (#) sign) indicates that the system is ready to accept input.

   If you will run the configuration script from a telnet prompt, perform the following:

   a. Return to the telnet session if need be.

      Type the following from the system prompt: **telnet TCPIP-Address port number**

      Here is an example of a Telnet command: `telnet 99.9.99.999 2023`, where 99.9.99.999 is the TCP IP-Address and 2023 is the port number.

   b. Switch from the administrators userid to your user ID.

3. Change directory to the deployment manager bin directory.

   ```
   cd /WebSphere/V6R0/DeploymentManager/bin/
   ```

4. If you have not already done so, switch to the administrator's user id. For example,

   ```
   su wsadmin
   ```

5. **Optional:** You may need to increase OMVS time limit to allow the product configuration script time to complete. To increase the OMVS time limit, type the following from the MVS console:

   ```
   SETOMVS MAXCPUTIME=86400
   ```

6. Run the product configuration script as demonstrated in the following syntax diagrams to configure WebSphere Process Server for z/OS in the Deployment Manager.

Syntax diagram for creating a network deployment configuration

```
                                  (1)
►►──zWPSConfig.sh──-augment──────┬-response─────────────────────────────┬──────►
                                 └──/usr/lpp/zWPS/V6R0/zos.config/DmgrDB2.rsp──┘

►─-Z <Name of the value in the property file to override>.─────────────────────►

►─-trace (By including -trace the command writes to the appropriate log file)────►◄
```

Notes:

1 Type the absolute path of the response file that is associated with the configuration onto which you are installing WebSphere Process Server for z/OS. This absolute pathname in this example assumes you are using the default response files. If you wanted to customize the response file, this pathname would need to represent the absolute path of the customized file.

Syntax diagram for creating a network deployment configuration as an ESB-only server

```
                                  (1)
►►──zWESBConfig.sh──-augment──────┬-response─────────────────────────────┬──────►
                                  └──/usr/lpp/zWPS/V6R0/zos.config/DmgrDB2.rsp──┘

►─-Z <Name of the value in the property file to override>.─────────────────────►

►─-trace (By including -trace the command writes to the appropriate log file)────►◄
```

Notes:

1 Type the absolute path of the response file that is associated with the configuration onto which you are installing WebSphere Process Server for z/OS. This absolute pathname in this example assumes you are using the default response files. If you wanted to customize the response file, this pathname would need to represent the absolute path of the customized file.

Attention: This job will run for nearly 75 minutes. While this job is running, you can begin the steps to configure the empty managed node.

7. Take another backup of your configuration HFS data sets.

Review the messages from Standard Out. These are either displayed on the screen from which the **zWPSConfig.sh** or **zWESBConfig.sh** was run or in the file specified if the redirect **(">")** symbol was used on the command line.

There should be no error messages displayed.

An example of a successful execution of **zWPSConfig.sh** or **zWESBConfig.sh** with the -augment option is as follows:

```
parsing command arguments...
parsing arguments complete
setting up configuration...
runtimeRootDirName is: /WebSphere/V6R0M0/DeploymentManager
WAS_HOME is: /WebSphere/V6R0M0/DeploymentManager
```

```
WBI_HOME is: /WebSphere/V6R0M0/DeploymentManager
set up configuration complete
augmenting profile(s)...
augmenting profile(s) complete
```

If you chose to delay the configuration of your databases, you will need to run the database definitions that were generated in the form of DDL scripts. These scripts reside in the WebSphere Application Server for z/OS configuration root *configuration_root/app_server_root*/profiles/default/dbscripts. For example, *WebSphere/V6R0/DeploymentManager*//profiles/default/dbscripts.

Creating an empty managed node configuration

Running **zWPSConfig.sh** or **zWESBConfig.sh** augments the empty managed node configuration with WebSphere Process Server for z/OS configuration data.

Many DB2 databases and tables are needed for the WebSphere Process Server for z/OS configuration. There are some useful hints and tips regarding database configuration in the technote titled, *WPS for z/OS Datasource & Database Configuration Hints & Tips*

Note: The *WPS for z/OS Datasource & Database Configuration Hints & Tips* technote applies to WebSphere Process Server for z/OS version 601, but some of the information regarding DB2 preparation may be useful.

The WebSphere Process Server for z/OS configuration process assumes a working knowledge of z/OS UNIX System Services. You may want to have access to the following documentation:

- z/OS V1R7.0 UNIX System Services User's Guide
- z/OS V1R7.0 UNIX System Services Command Reference

Before you create a managed node configuration, make sure of the following:

1. You have completed the steps in "Create a Network Deployment configuration" on page 183.
2. The WebSphere Process Server for z/OS installation script has been run.

   Running the installation script creates the symlinks from `$WAS_HOME` to the WebSphere Process Server for z/OS SMP/E code and updates the administrative console with the WebSphere Process Server for z/OS definitions.

   See "Running the installation script" on page 93 for more information.
3. Copy the deployment manager response file to your working directory. For example,

   ```
   cp /usr/lpp/zWPS/V6R0/zos.config/ManagedDB2.rsp
   ```
4. Make sure that the response file has appropriate permissions:

   ```
   chmod 755 ManagedDB2.rsp
   ```
5. Tailor the deployment manager response file to meet your environment specifications.

   See Working with response files for information on editing properties in the response file.

   For the ManagedDB2.rsp, modify the following fields in the response file (always using the LONG names):

   ```
   JMSUSER=jmsuser---> The userids are used to create J2C authentication aliases for resources created.
   JMSPASS=passw0rd
   DBUSER=db2user
   DBPASS=passw0rd
   CONFIGSERVER=dmgr
   DBLOCATION=LOC1  ----> Needs to be DB2 LOCATION...not SSID
   ```

```
profilePath=/WebSphere/V6R0/DeploymentManager/profiles/default
templatePath=/WebSphere/V6R0/DeploymentManager/profileTemplates/dmgr.*
cellName=cl1celld
nodeName=cl1dmnod
dbType=DB2UDBOS390_V8_1
dbJDBCClasspath=/usr/lpp/db2810/jcc/classes
dbJDBCProperties=/etc/db2cfg
```

Important: The properties listed above represent the properties you must customize for your deployment manager configuration. For all of the other properties in the sample response file, you can accept the default values (many of which use variable substitution). Depending on the database and storage groups created, you may also need to change the following:

```
dbName=WPSDB
dbStorageGroup=WPSDBSTO
```

For information on creating the database and storage groups, see Creating the databases and storage groups.

Make sure that value for delaying database configuration is set to ″true″ **dbDelayConfig=true**. Delaying database configuration means that the configuration script will not automatically run the DDL scripts to create the database resources. When **dbDelayConfig=true** the database definitions contained in the DDL are written to a directory and must be run manually *after the configuration process*. See "Configuring the database manually" on page 207 for information.

6. If you have not already done so, create a **DB2JccConfiguration.properties** file in /etc/db2cfg/ if your DB2 SSID does not equal DSN8.

   This properties file should have the following line in it and it should be read-able to all:

   ```
   db2.jcc.ssid=DB15
   ```

   where **DB15** is your installation's SSID.

Perform this task when your goal is to create an empty managed node configuration

1. Prepare to run the configuration script. If you will run the configuration script from an OS/390 UNIX command shell, perform the following:
   a. Access the OS/390 UNIX command shell.
   b. Enter the TSO command OMVS at the ISPF Command Shell or TSO OMVS from any other ISPF panel.
   c. Once you're in the UNIX shell, a command prompt (usually a dollar ($) or pound (#) sign) indicates that the system is ready to accept input.

   If you will run the configuration script from a telnet prompt, perform the following:
   a. Return to the telnet session if need be.

      Type the following from the system prompt: **telnet TCPIP-Address port number**

      Here is an example of a Telnet command: `telnet 99.9.99.999 2023`, where 99.9.99.999 is the TCP IP-Address and 2023 is the port number.
   b.
   c. Change directory to:
2. Switch userid to the administrator's (if not already running as wsadmin): `su wsadmin (pwd = passw0rd)`

3. **Optional:** You may need to increase OMVS time limit to allow the product configuration script time to complete. To increase the OMVS time limit, type the following from the MVS console:

   ```
   SETOMVS MAXCPUTIME=86400
   ```

4. Run the product configuration script as demonstrated in the following syntax diagrams:

   Syntax diagram for creating a managed node configuration

   ```
   ►►──zWPSConfig.sh──-augment──────────────────────────────────────────────────►

     (1)
   ►──────┬-response──────────────────────────────┬─────────────────────────────►
          └──/usr/lpp/zWPS/V6R0/zos.config/ManagedDB2.rsp──┘

   ►──Z <Name of the value in the property file to override>.───────────────────►

   ►──trace (By including -trace the command writes to the appropriate log file)───────►◄
   ```

   Notes:

   1 Type the absolute path of the response file that is associated with the configuration onto which you are installing WebSphere Process Server for z/OS. This absolute pathname in this example assumes you are using the default response files. If you wanted to customize the response file, this pathname would need to represent the absolute path of the customized file.

   Syntax diagram for creating a network deployment configuration as an ESB-only server

   ```
   ►►──zWESBConfig.sh──-augment─────────────────────────────────────────────────►

     (1)
   ►──────┬-response──────────────────────────────┬─────────────────────────────►
          └──/usr/lpp/zWPS/V6R0/zos.config/ManagedDB2.rsp──┘

   ►──Z <Name of the value in the property file to override>.───────────────────►

   ►──trace (By including -trace the command writes to the appropriate log file)───────►◄
   ```

   Notes:

   1 Type the absolute path of the response file that is associated with the configuration onto which you are installing WebSphere Process Server for z/OS. This absolute pathname in this example assumes you are using the default response files. If you wanted to customize the response file, this pathname would need to represent the absolute path of the customized file.

Review the messages from Standard Out. These are either displayed on the screen from which the **zWPSConfig.sh** or **zWESBConfig.sh** was run or in the file specified if the redirect **(″>″)** symbol was used on the command line.

There should be no error messages displayed.

An example of a successful execution of **zWPSConfig.sh** or **zWESBConfig.sh** with the -augment option is as follows:

```
parsing command arguments...
parsing arguments complete
setting up configuration...
runtimeRootDirName is: /WebSphere/V6R0M0/AppServer
WAS_HOME is: /WebSphere/V6R0M0/AppServer
WBI_HOME is: /WebSphere/V6R0M0/AppServer
set up configuration complete
augmenting profile(s)...
augmenting profile(s) complete
```

If you specified values for the Business Process Choreographer sample configuration in the response file, a sample configuration that includes the business process container, the human task container and the Business Process Choreographer Explorer are now part of your Process Server configuration. You can check to see if these components are configured by looking in the administrative console for enterprise applications with names that start with BPEContainer, BPCExplorer, and TaskContainer.

Note: The sample configuration is not suitable for a production system. Because you can only have one Business Process Choreographer configuration, you must remove the sample configuration, as described in as described in *Removing the Business Process Choreographer configuration* the Business Process Choreographer PDF, before you continue configuring Business Process Choreographer to use WebSphere MQ or a different database.

For additional information, see Verifying the product configuration.

Federating an empty managed node to a deployment manager

Run the BBOWMNAN job to federate a WebSphere Process Server for z/OS empty managed node into a deployment manager cell.

Do the following before using this procedure:

- Ensure you have installed WebSphere Process Server for z/OS, and that you have created a WebSphere Process Server for z/OS deployment manager and a custom node. This procedure assumes you did *not* federate the custom node during configuration process.
- Ensure that the deployment manager is running. If it is not, start it either from a telnet session or from an MVS console as follows:

  To start the deployment manager from a telnet session, perform the following:

  1. Make sure you are still the wsadmin userid: su wsadmin
  2. Issue the following command:cd /WebSphere/V6R0/DeploymentManager/bin ./startManager.sh

  To start the deployment manager from the MVS console:

  1. Issue the associated START command from the MVS console as follows:

     ```
     START <proc name>,JOBNAME=<DMserverShortName>,ENV=<cellShortName>.<DM nodeShortName>.<DM serverShortName>
     ```
- Ensure that the deployment manager is a WebSphere Process Server for z/OS profile (not a WebSphere Application Server for z/OS deployment manager profile).
- Ensure that the deployment manager uses the default SOAP JMX connector type and the connector is enabled.

Attention: Do *not* federate a custom node at this time if any one of the following is true:

- The deployment manager is not running or you are not sure if it is running.

  Note: If you federate an empty managed node when the deployment manager is not running or is not available for other reasons, federation will fail and the resulting augmentation to the profile will be unusable. You must then move this custom profile directory out of the profile repository (by default, the `profiles` directory in *install_root*, where *install_root* is the WebSphere Process Server installation directory) before creating another managed node profile with the same name.
- The deployment manager is a WebSphere Application Server for z/OS profile that has not yet been configured/augmented with WebSphere Process Server for z/OS configuration data.
- The SOAP connector is disabled.
- The deployment manager is re-configured to use the non-default remote method invocation (RMI) as the preferred Java Management Extensions (JMX) connector. (Select **System administration > Deployment manager > Administration services** in the administrative console of the deployment manager to verify the preferred connector type.)

Federate an empty managed node into a deployment manager cell by performing the following:

1. Run the BBOWMNAN job that was created when you built the empty-managed node. Verify that it completes with a RC=0.
2. Turn off tracing. From the MVS console, issue the following MVS command:
   ```
   f <DMControlRgn>,traceinit
   ```
3. The Node Agent should have started during federation but if you selected that it not be, start the Node Agent for the node now: cd /WebSphere/V6R0/ AppServer/bin./startNode.sh

The empty managed node is federated into the deployment manager.

After federating the empty managed node, go to the administrative console of the deployment manager to customize it.

Download and run the variables jacl script to adjust console settings

Run wpsFixvars.jacl to adjust variable content created by the configuration script.

Make sure you have run the configuration scripts for deployment manager and managed node successfully.

Before administering server configurations from the administrative console, download and run the script wpsFixvars.jacl

The script wpsFixvars.jacl customizes variable settings on the administrative console.

Note: You can also customize variable settings manually using the administrative console. For information on adjusting variable settings manually, see Adjusting variable settings using the administrative console.

1. Go to the site on which the sample script resides.

The support site containing the sample script to fix variable settings is at:Techdocs - The Technical Sales Library.

2. Upload it to your working directory in binary form.
3. Change to the deployment manager directory `cd /WebSphere/V6R0/DeploymentManager/bin`
4. Run the script as follows:

```
./wsadmin.sh -conntype none -f /u/wsuser/wpswork/wpsFixvars.jacl -wbiInstallRoot
 "/WebSphere/V6R0/DeploymentManager" -node "cl1dmnod"
-jcchome /usr/lpp/db2810/jcc -jccproperties /etc/db2cfg/ DB2JccConfiguration.properties
```

Continue configuration activities by federating the empty managed node.

Adjusting variable settings using the administrative console

Perform steps from the administrative console to adjust variable content created by the configuration script.

Make sure you have run the configuration scripts for deployment manager and managed node successfully.

Before administering server configurations from the administrative console, you need to adjust variable settings created by the configuration scripts. You can do this manually from the administrative console as described in this article, or you can run the script wpsFixvars.jacl to customizes variable settings, as describe in "Download and run the variables jacl script to adjust console settings" on page 193.

1. Add DB2UNIVERSAL_JDBC_DRIVER_PATH variable at the cell level Some JDBC resources will be created at the cell level. We therefore need to define a DB2UNIVERSAL_JDBC_DRIVER_PATH variable there.

   The DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH variable should already exist there.

   Go to `Environment` > `WebSphere Variables` and blank out all but the cell name.

   Click **Apply**
2. Select **New** and fill out the appropriate value for DB2UNIVERSAL_JDBC_DRIVER_PATH
3. Click **OK**

   Name: DB2UNIVERSAL_JDBC_DRIVER

   Value: /usr/lpp/db2810/jcc/classes
4. Set ws.ext.dirs fields for the Common Event Infrastructure on the Deployment Manager JVM.

   The ws.ext.dirs field needs to be set to ${CEI_HOME}/lib:${CEI_HOME}/client on Deployment Manager's JVM.

   Do this now before starting the Deployment Manager for the first time.

   Without this on the Deployment Manager, you will see the following message when trying to deploy the event application later:

   ```
   Exception message:
   com.ibm.websphere.management.exception.InvalidConfigDataTypeException:
   ADMG0007E: The configuration data type EventInfrastructureProvider is not valid.
   ```
5. Logon to the admin console and go to the deployment managers JVM Custom Properties

   `System Administration -> Deployment Manager -> Process Definition->Servant->Java Virtual Machine->Custom Properties.`
6. Select **New** and fill in the fields as shown:

Name: `ws.ext.dirs`

Value: `${CEI_HOME}/lib:${CEI_HOME}/client`

7. Click **OK**
8. Set db2.jcc.properties custom property on the Deployment Manager JVM

   From the same Custom Properties Dialog. Select **New** and fill in the fields as shown:

   Name: `db2.jcc.propertiesFile`

   `Value: /etc/db2cfg/DB2JccConfiguration.properties`
9. Click **OK**
10. Save the configuration, syncing with your node.

Continue configuration activities by federating the empty managed node.

Making Service Component Architecture services accessible across cells

One of the benefits of Service Component Architecture (SCA) is the ability for consumers to use services that already exist in other service modules. The service provider and the service consumer can reside on different cells. This distribution allows you to isolate and mange services better by placing the services amongst the cells.

For example, an application on server A may require a service installed in server B in a different cell. To use services this way, you must configure communications across the cells.

To support all consumers of a service, a service provider should provide both an asynchronous and a synchronous interface to the service on the cell.

For a service consumer, you can limit the type of configuration you perform to the call interaction style of the application. When the administrator does not know the interaction style used by the application, configure both styles.

Asynchronous view of cross-cell service usage

Figure 8 on page 196 illustrates an environment where there is a service on the provider cell that modules on consumer cell 1 and consumer cell 2 use asynchronously. In the asynchronous environment, both the providing and consuming cells must define the connections between the cells for the service to be available. If availability is a concern, it is best to use cluster connections so that the connections remain available if specific nodes fail. Note that the connections are between messaging engines on each cell or cluster.

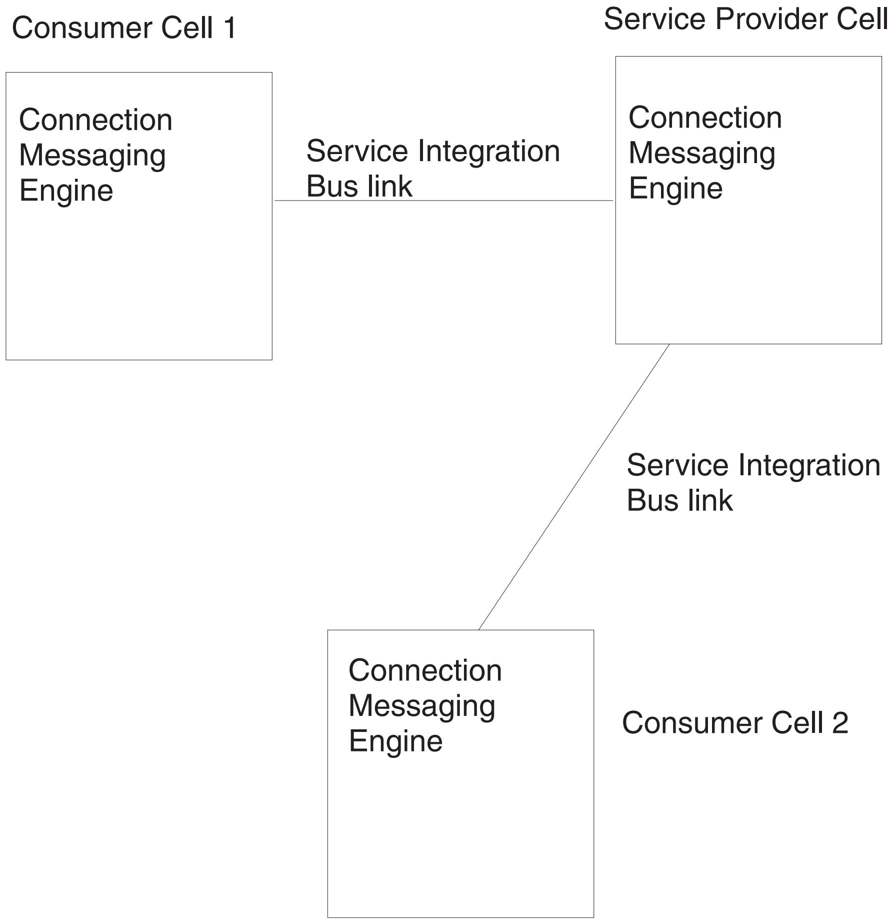

Consumer Cell 1
Connection Messaging Engine
Service Integration Bus link
Service Provider Cell
Connection Messaging Engine
Service Integration Bus link
Connection Messaging Engine
Consumer Cell 2

Figure 8. An asynchronous environment across cells

The connection between the service provider cell and the consumer cells is the service integration bus link and you configure the connection manually. To complete the configuration, the service provider cell must expose the IP address and port the consumer cells must use to communicate with the service they are using.

Synchronous view of cross-cell service usage

Figure 9 on page 197 illustrates an environment where there is a service on the provider cell that modules on consumer cell 1 and consumer cell 2 use synchronously. In the synchronous environment, the consuming cells must define the connections between the cells for the service to be available. If availability is a concern, it is best to use cluster connections so that the connections remain available if specific nodes fail.

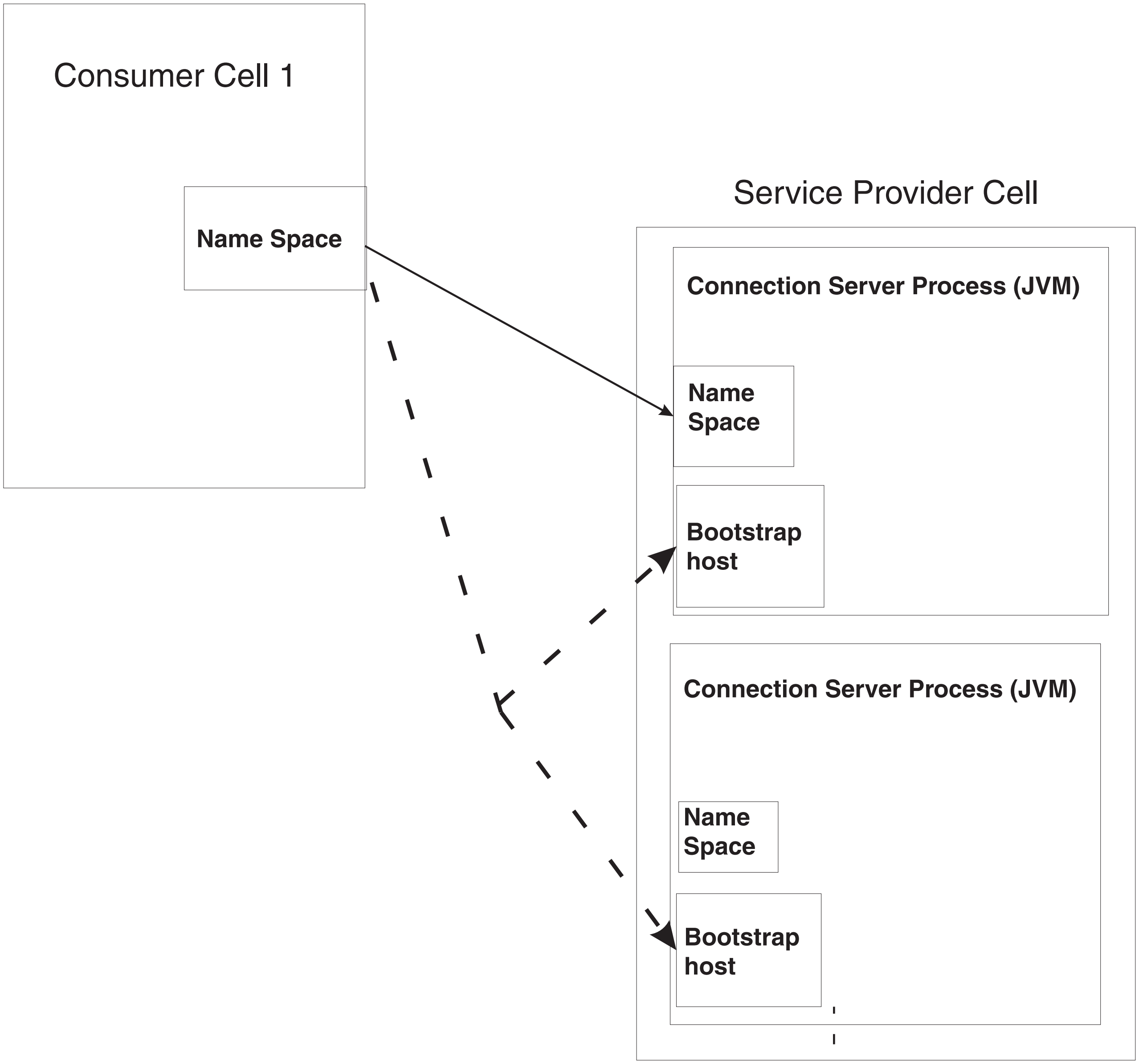


Figure 9. A synchronous environment across cells

You make a service available in the providing cell by exposing the Java Naming and Directory Interface (JNDI) name space that contains the reference to the service. In the consuming cell, making the service available is a matter of creating a name space binding to the exposed JNDI name space.

Providing synchronous access to Service Component Architecture services from outside the cell

For consuming cells to reference a service on a providing cell, the consuming cell administrator must establish a connection to the providing service. You establish and provide the information that the administrator of the consuming cells needs to configure the link in Java Naming and Directory Interface (JNDI).

This task assumes that:

- You have installed the Service Component Architecture (SCA) module containing the service on the providing cell.

- You are at the administrative console for the providing cell.

A SCA module, in a different cell, communicates synchronously with your SCA module by setting up a JNDI name in their cell to find your SCA module. When you perform this task you enable others to set up that JNDI name.

1. **Optional:** Identify the two or more servers that will be used by other cells as the bootstrap hosts for this cell.

   Note: Defining at least two bootstrap hosts keeps the service available even if one of the hosts fails for some reason.

   For example, the two servers in the service provider cell in Figure 10 will contain the bootstrap hosts. For this purpose, the IP addresses for the top server is 9.26.237.144, and the bottom server in the figure is 9.26.237.150.

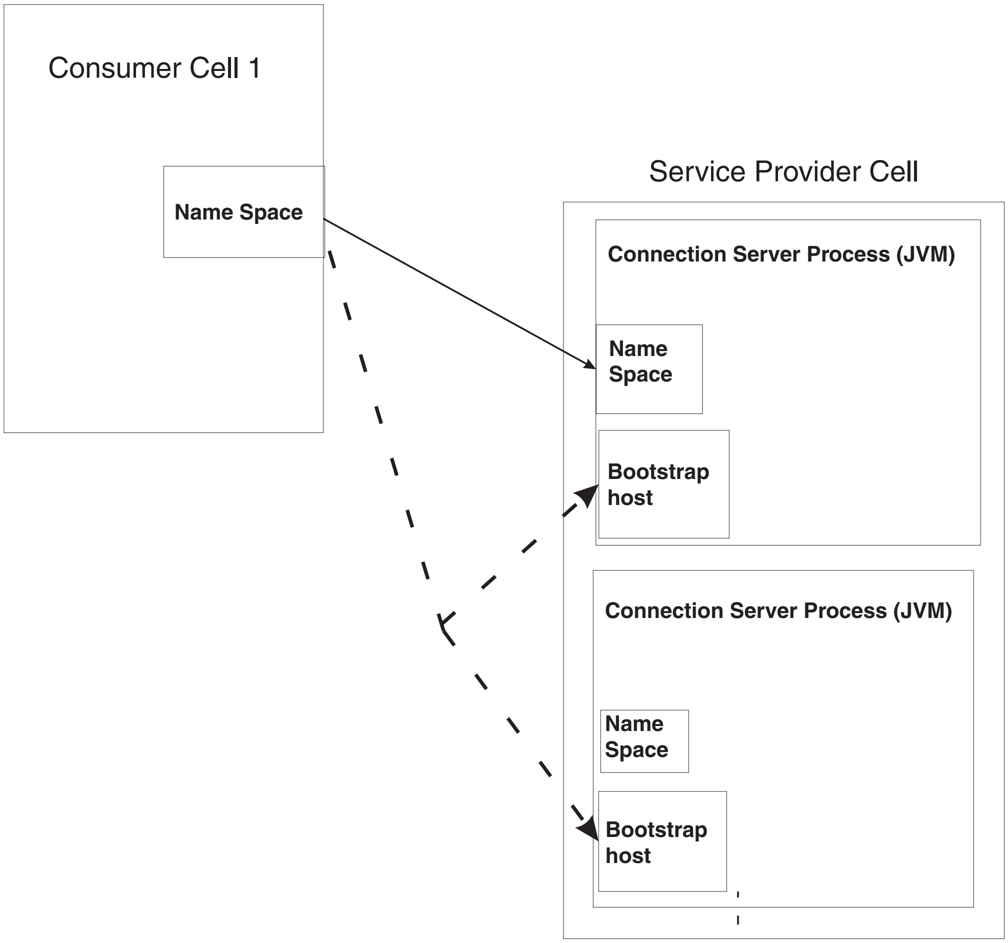


Figure 10. A synchronous environment across cells

   Important: This procedure only describes configuring a single bootstrap host. Configure at least one to the providing cell, if availability is not a requirement; two or more if availability is a requirement.

   Tips:

   - Using virtual hosts will improve the availability of the service when maintenance requires replacement or upgrading of the hardware.
   - You can use a single bootstrap host if you do not have a requirement for availability.

2. Locate the bootstrap port number for each of the servers you selected in step 1 on page 198.

   Display this information by clicking **Servers > Application servers >** ***server_name*** **> Communications > + Ports > Bootstrap_Address**.

   For example, the port for both servers in the figure is 2810.

3. Publish the IP addresses and port numbers to all cells who are consuming services that reside in your cell.

   In this example, a memo would be sent to consumers advertising that the bootstrap hosts established for this cell are:

   - 9.26.237.144:2810
   - 9.26.237.150:2810

   The format to publish would be

   ```
   corbaname:iiop:9.26.237.144:2810;9.26.237.150:2810
   ```

4. Provide the JNDI name of the enterprise Java bean (EJB) the consumers invoke.

Make sure to keep the published bootstrap host IP addresses and ports available regardless of any other changes to the physical makeup of the cell.

Calling Service Component Architecture services in another cell synchronously

When you invoke a Service Component Architecture (SCA) module in a different cell synchronously you must add a name space binding to bind to the remote SCA service.

The task assumes that:

- You have received bootstrap host information for the cell that is running the service your cell wants to invoke.
- You are at the administrative console for the consuming cell.
- You have already installed the SCA modules consuming the target service.

To communicate synchronously with a SCA service in another cell you need to configure your JNDI name space to find the service. You perform this task to configure your JNDI name space.

Note: For the purposes of this task, the consuming service module resides on cell A and the providing service module resides on cell B.

For the purposes of this task, Figure 9 on page 197 contains the IP information to use in the configuration.

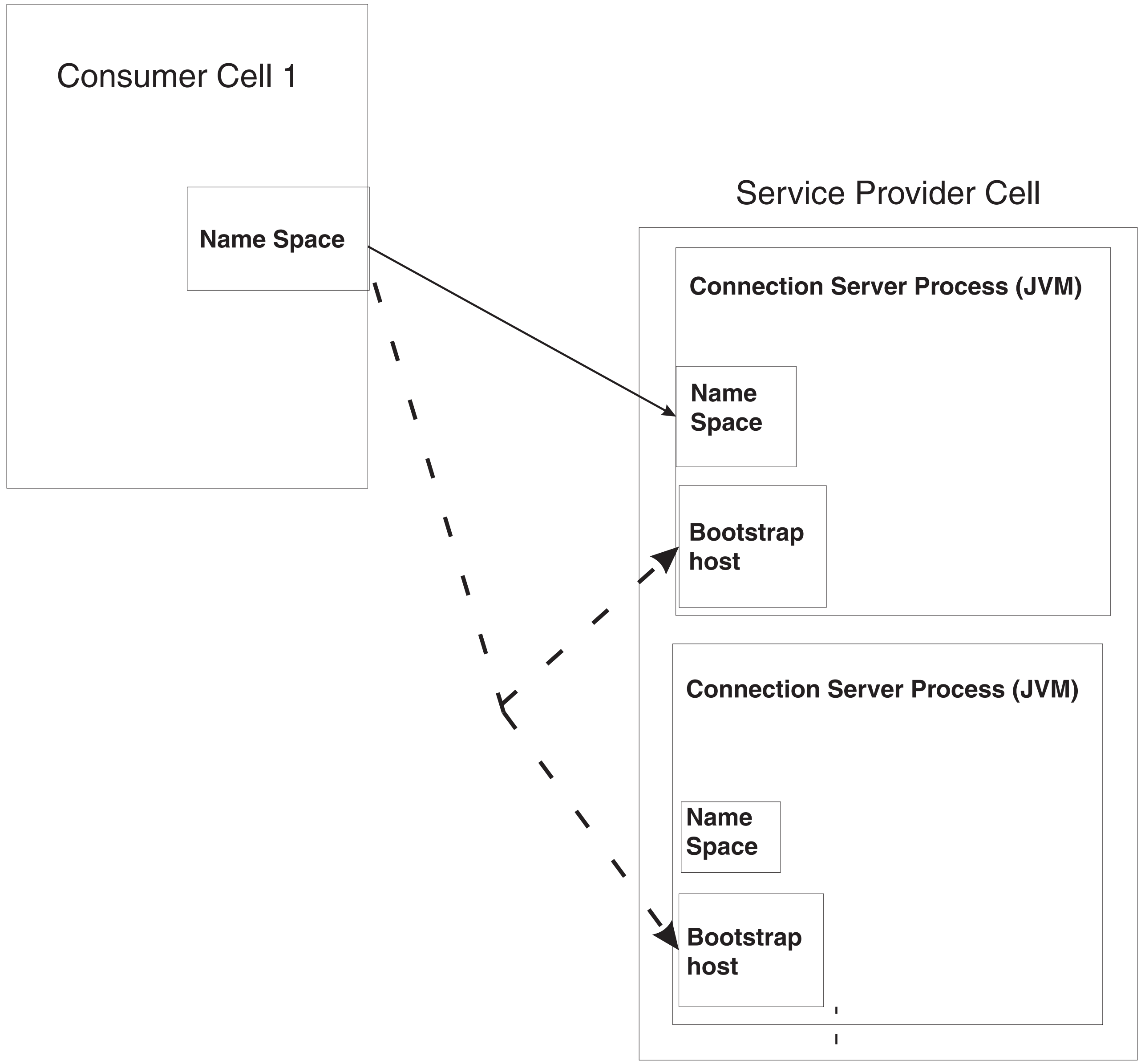


Figure 11. A synchronous environment across cells

1. Obtain the bootstrap host information from the administrator of the providing cell.

   Assuming that the bootstrap port is 2810 in Figure 9 on page 197, this information would look like corbaname:iiop:9.26.237.144:2810 (corbaname:iiop:9.26.237.150:2810).

2. Create a new namespace binding on the invoking cell (Cell A, in the example) pointing to the export in the providing cell.

   On the **Environment > Naming > Name Space Bindings** panel, select a scope of **Cell** and click **Apply**. With the changed scope, click **New** in the display to create the new binding.

   In the wizard, specify the following (the values are appropriate for the example configuration):

   a. Binding type is Indirect

   b. The basic properties are:

- **Binding identifier** is a unique string, for example: `sca_import_test_sca_cross_simple_custinfo_CustomerInfo`
- **Name in Name Space** is the JNDI name of the enterprise Java bean (EJB) of the service you are invoking in the providing cell, for example, `sca/SimpleBOCrsmB/export/test/sca/cros/simple/custinfo/CustomerInfo` . This names the export interface in the providing cell.
- **Provider URL** is the bootstrap host information of the providing cell you received in step 1 on page 200. For example `corbaname:iiop:9.26.237.144:2810` (`corbaname:iiop:9.26.237.150:2810`).
- **JNDI name** is the JNDI name of the enterprise Java bean (EJB) that represents the service you are invoking on the providing cell, for example, `sca/SimpleBOCrsmB/export/test/sca/cros/simple/custinfo/CustomerInfo`. This should be the same value as the **Name in Name Space**.

  Note: Deploying the module on the providing cell creates this name automatically. Contact the providing cell administrator to obtain this name.

When finished, click **Next** and verify the values on the **Summary** page. After verifying, click **Finish**.

The administrative console displays your new binding.

3. Save your changes by clicking **Save**.

Start the application. The SCA module on Cell A can now synchronously invoke the SCA module on Cell B.

Providing Service Component Architecture services asynchronously across cells

To enable communication between Service Component Architecture (SCA) modules in different cells you have to configure a communication link between the two cells. This topic describes the configuration you perform on the providing cell to enable the communication from modules that asynchronously invoke SCA services on other cells.

The task assumes that:

- You are using an administrative console on the providing cell.
- You have already installed the SCA modules involved, but you have not yet started the consuming modules.
- There is a different administrator for the cell on which the consuming module runs.

Before starting an SCA module that requires the services of an SCA module installed on another cell, you must configure both cells so they can communicate the requests. For SCA modules that use asynchronous invocations, the process involves foreign buses and Service Integration Bus (SIBus) links.

Note: For the purposes of this task, the consuming service module resides on cell A and the providing service module resides on cell B.

Figure 12 on page 202 contains the information to use in the configuration.

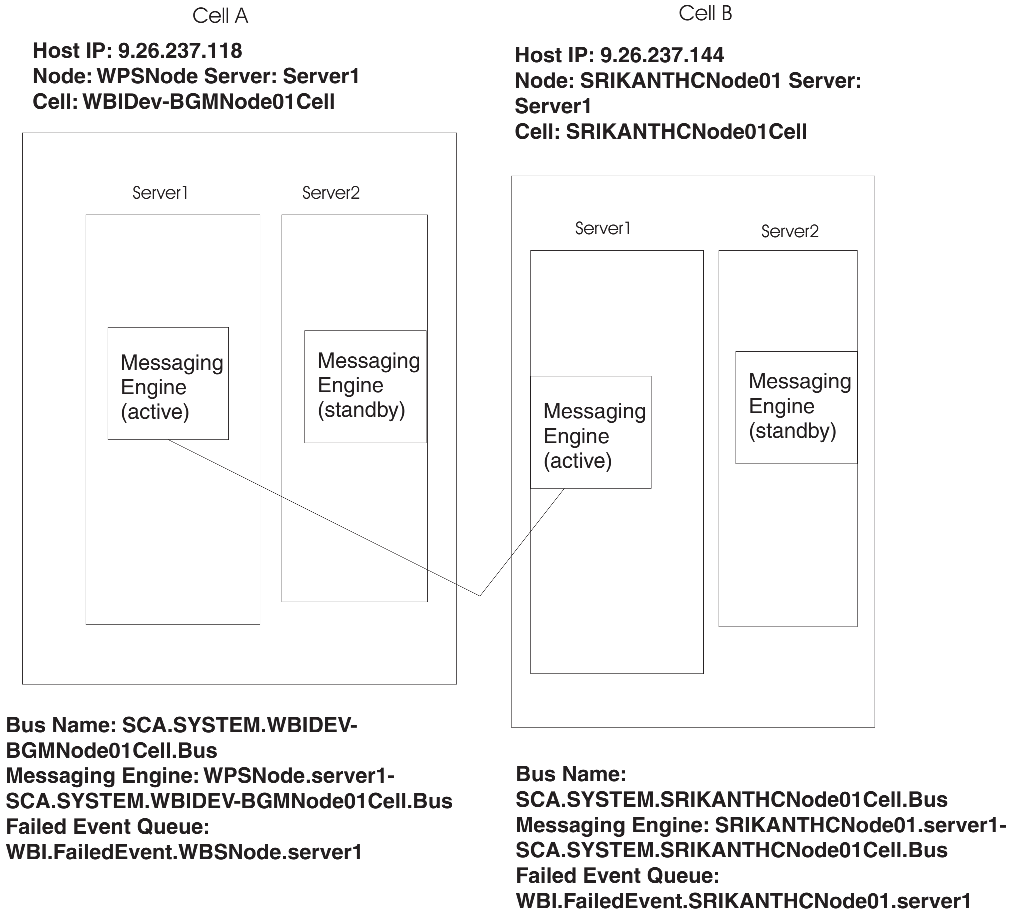

Cell A
Host IP: 9.26.237.118
Node: WPSNode Server: Server1
Cell: WBIDev-BGMNode01Cell
Server1
Server2
Messaging Engine (active)
Messaging Engine (standby)
Bus Name: SCA.SYSTEM.WBIDEV-BGMNode01Cell.Bus
Messaging Engine: WPSNode.server1-SCA.SYSTEM.WBIDEV-BGMNode01Cell.Bus
Failed Event Queue: WBI.FailedEvent.WBSNode.server1
Cell B
Host IP: 9.26.237.144
Node: SRIKANTHCNode01 Server: Server1
Cell: SRIKANTHCNode01Cell
Server1
Server2
Messaging Engine (active)
Messaging Engine (standby)
Bus Name: SCA.SYSTEM.SRIKANTHCNode01Cell.Bus
Messaging Engine: SRIKANTHCNode01.server1-SCA.SYSTEM.SRIKANTHCNode01Cell.Bus
Failed Event Queue: WBI.FailedEvent.SRIKANTHCNode01.server1

Figure 12. Invoking a SCA module in a different cell

1. Define the IP addresses that you will expose to other cells to use to connect to this cell. Choose one of these methods:
   - When a stand-alone profile hosts the service, use the IP address of the server hardware.
   - When you need isolation between the cells, use a virtual IP address.

     Tip: Using virtual IP addresses will improve the availability of the service when maintenance requires replacement or upgrading of the hardware.
   - When you require availability to the service, use multiple IP addresses, for example 9.26.237.144 and 9.26.427.123.

     Note: Defining at least two hosts keeps the service available even if one of the hosts fails for some reason.
2. Create a server or cluster member and include it as a member of the SCA system bus. The resulting messaging engine will be used as the connection to the consuming cells.
3. Give the information that identifies the providing cell to the administrator of the cell that runs the module consuming the service. This information includes:
   - Host IP address
   - Port number - find this by displaying the **BOOTSTRAP_ADDRESS** value at **Servers > Application servers >** *server_name* **> Communications > + Ports**

- Bus name - find this by clicking **Service integration > Buses** and locate the full name of the SCA.SYSTEM bus.
- Messaging engine find this by clicking on **Service integration > Buses >** ***SCA_SystemBusName*** **> Messaging engines** and locate the messaging engine in use by the service you are providing to the consuming cells.
- Failed Event Queue name - find this by displaying **Service integration > Buses >** ***SCA_SystemBusName*** **>** ***moduleDest*** and examining the **Exception destination** attribute. If this attribute has selected:
  - **Specify**, use the value in the text field
  - **System**, click on **Service integration > Buses >** ***SCA_SystemBusName*** **> Destinations** and use the value of the system exception destination.

Notes:

- *SCA_SystemBusName* has the format SCA.SYSTEM.*cellname*.Bus.
- *moduleDest* has the format of sca/*moduleName*

4. Get the information that identifies the consuming cell from the administrator of that cell. This information includes:
   - Host IP address
   - Port number
   - Bus name
   - Messaging engine
   - Failed Event Queue name
5. Create a foreign bus and set the routing definition type to `Direct, service integration bus link`. See Adding a foreign bus in the WebSphere Application Server for z/OS information center.
6. **Optional:** For each destination that requires a response to be sent to the invoking system, create a destination on the providing server and configure it to point back to the consuming SCA module in the other cell. This involves creating destinations, setting up forward routing paths, and setting exception destinations.

   Note: If the invoking system does not require a response, skip this step.

   a. Create destination.

      From the example on Cell B, based on the information from the consuming module in Cell A, you would create additional destinations on the bus in cell A:

      ```
      sca/SimpleBOCrsmA/import/test/sca/cros/simple/custinfo/CustomerInfo
      sca/SimpleBOCrsmA/component/test/sca/cros/simple/cust/Customer
      ```

   b. Set the forwarding paths to point to their counterparts on consuming cell.

      This would look like:

      ```
      SCA.SYSTEM.WBIDev-BGMNode01Cell.Bus:
      sca/SimpleBOCrsmA/import/test/sca/cros/simple/custinfo/CustomerInfo
      SCA.SYSTEM.WBIDev-BGMNode01Cell.Bus:
      sca/SimpleBOCrsmA/component/test/sca/cros/simple/cust/Customer
      ```

   c. Set the exception destination to the Failed Event queue for both of the destinations you created.

      From the example, the value would be:`WBI.FailedEventSRIKANTHCNode01.server1`.
7. On the messaging engine created in step 2 on page 202 set up an SIB link using the information from step 4.

   See Adding a service integration bus link in the WebSphere Application Server for z/OS information center for more information.

For example, on Cell B:

```
SIB Link: TestCrossCell
Remote ME: WPSNode.server1.SCA.SYSTEM.WBIDev-BGMNode01.Cell.Bus
Bootstrap: 9.26.237.118:7276:BootstrapBasicMessaging
```

Restriction: When providing a service that sends a response to the invoking system, there can be only one invoking system for each link.

Important: The port number in the bootstrap is the SIB endpoint address port. If you enabled security, you must use the secure SIB endpoint address port.

8. **Optional:** Add sender roles to the foreign buses, if you enabled security on the systems. Make sure to define the user each application uses on both systems from the operating system command prompt. The command to add the role is:

   ```
   wsadmin $AdminTask addUserToForeignBusRole -bus busName
     -foreignBus foreignBusName -role roleName -user userName
   ```

   Where:

   busName

   Is the name of the bus on the system you enter the command.

   foreignBusName

   Is the foreign bus to which you are adding the user.

   userName

   Is the userid to add to the foreign bus.
9. Verify the connection. Coordinate with the consuming administrator to recycle the servers involved with the connection by restarting the servers.

   You should see messages similar to:

   ```
   [8/24/05 11:00:09:741 PDT] 00000086 SibMessage    I   [SCA.SYSTEM.WBIDev-BGMNode01Cell.Bus:WPSNode.server1-SCA.SYSTEM.WBIDev-BGMNode01Cell.Bus]
   CWSIP0382I: messaging engine 2D7333574B0CD70B responded to subscription request, Publish Subscribe topology now consistent.
   ```
10. Repeat steps 4 on page 203 through 9 for each consuming cell.

Start the applications.

Using Service Component Architecture services asynchronously across cells

To enable communication between Service Component Architecture (SCA) modules in different cells, you have to configure a communication link between the two cells. This topic describes the configuration you perform on the consuming cell to enable the communication from modules that asynchronously invoke SCA services on other cells.

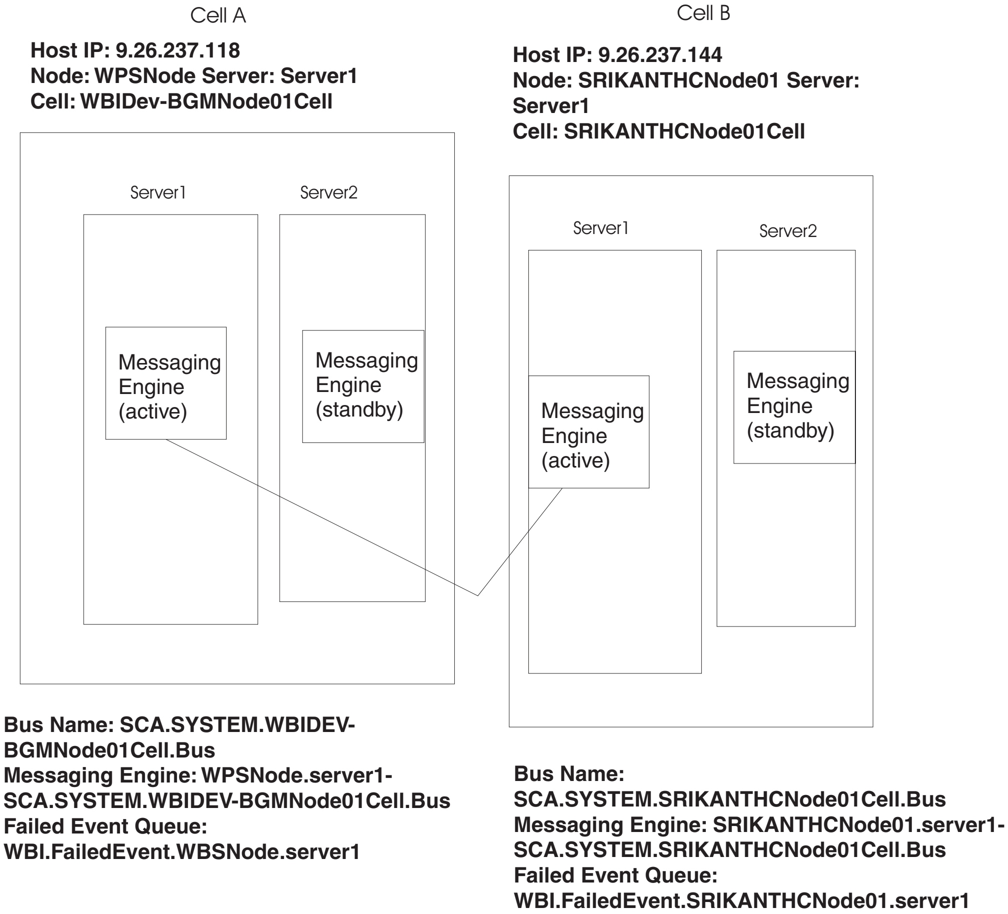
The task assumes that:

- You are using an administrative console on a consuming cell.
- You have already installed the SCA modules involved, but you have not yet started the consuming modules.
- There is a different administrator for the cell on which the providing module runs.

Before starting an SCA module that requires the services of an SCA module installed on another cell, you must configure both cells so they can communicate the requests. For SCA modules that use asynchronous invocations, the process involves foreign buses and Service Integration Bus (SIBus) links.

Note: For the purposes of this task, the consuming service module resides on cell A and the providing service module resides on cell B.

Figure 13 contains the information to use in the configuration.


Cell A
Cell B
Server1
Server2
Messaging Engine (active)
Messaging Engine (standby)
Server1
Server2
Messaging Engine (active)
Messaging Engine (standby)

Host IP: 9.26.237.118
Node: WPSNode Server: Server1
Cell: WBIDev-BGMNode01Cell

Host IP: 9.26.237.144
Node: SRIKANTHCNode01 Server: Server1
Cell: SRIKANTHCNode01Cell

Bus Name: SCA.SYSTEM.WBIDEV-BGMNode01Cell.Bus
Messaging Engine: WPSNode.server1-SCA.SYSTEM.WBIDEV-BGMNode01Cell.Bus
Failed Event Queue: WBI.FailedEvent.WBSNode.server1

Bus Name: SCA.SYSTEM.SRIKANTHCNode01Cell.Bus
Messaging Engine: SRIKANTHCNode01.server1-SCA.SYSTEM.SRIKANTHCNode01Cell.Bus
Failed Event Queue: WBI.FailedEvent.SRIKANTHCNode01.server1

Figure 13. Invoking a SCA module in a different cell

1. Create a server or cluster member and include it as a member of the SCA system bus. The resulting messaging engine will be used as the connection to the consuming cells.
2. Obtain the information that identifies the cell providing the service from the administrator of that cell. The information to provide includes:
   - Host IP address
   - Port number
   - Bus name
   - Messaging engine
   - Failed Event Queue name
3. Give the information about your cell to the administrator of the cell providing the service your module will invoke. The information to provide includes:
   - Host IP address
   - Port number - find this by displaying the **BOOTSTRAP_ADDRESS** value at **Servers > Application servers >** *server_name* **> Communications > + Ports**
   - Bus name - find this by clicking **Service integration > Buses** and locate the full name of the SCA.SYSTEM bus.

- Messaging engine find this by clicking on **Service integration > Buses >** ***SCA_SystemBusName*** **> Messaging engines** and locate the messaging engine in use by the service you are providing to the consuming cells.
- Failed Event Queue name - find this by displaying **Service integration > Buses >** ***SCA_SystemBusName*** **>** ***moduleDest*** and examining the **Exception destination** attribute. If this attribute has selected:
  - **Specify**, use the value in the text field
  - **System**, click on **Service integration > Buses >** ***SCA_SystemBusName*** **> Destinations** and use the value of the system exception destination.

4. Using the information from step 2 on page 205, create a foreign bus to represent the bus of this provider cell and set the routing definition type to `Direct, service integration bus link`. Repeat this step for each provider cell if you require multiple provider cells.

   SeeAdding a foreign bus in the WebSphere Application Server for z/OS information center for more information.

   From the example, the foreign bus on Cell A would be SCA.SYSTEM.SRIKANTHCNode01Cell.Bus. The foreign bus on Cell B would be SCA.SYSTEM.WBIDev-BGMNode01Cell.Bus.
5. On the messaging engine created in step 1 on page 205 set up a SIB link using the information from step 2 on page 205.

   See Adding a service integration bus link in the WebSphere Application Server for z/OS information center for more information.

   From the example, the SIB mediation link on Cell A would be:

   ```
   SIB Link: TestCrossCell
   Remote ME: SRIKANTHCNode01.server1-SCA.SYSTEM.SRIKANTHCNode01Cell.Bus
   Bootstrap: 9.26.237.144:7277:BootstrapBasicMessaging
   ```

   Attention: The port number in the bootstrap is the SIB endpoint address port. If you enabled security, you must use the secure SIB endpoint address port.
6. Display the destinations for each SCA module.
7. Modify the forwarding path of outgoing destinations of the consuming service module that must be wired to targets on the providing system.

   The destination to wire will have `importlink` in the destination name, for example on Cell A the destination would be sca/SimpleBoCrsmA/importlink/test/sca/cros/simple/custinfo/CustomerInfo. Modify the path by prefixing the foreign bus name to the destination name. From the example, the foreign bus name for the second cell is SCA.SYSTEM.SRIKANTHCNode01Cell.Bus. The result is

   ```
   SCA.SYSTEM.SRIKANTHCNode01Cell.Bus:sca/SimpleBoCrsmA/importlink/
   test/sca/cros/simple/custinfo/CustomerInfo
   ```
8. **Optional:** Add sender roles to the foreign buses, if you enabled security on the systems. Make sure to define the user each application uses on both systems from the operating system command prompt. The command to add the role is:

   ```
   wsadmin $AdminTask addUserToForeignBusRole -bus busName
     -foreignBus foreignBusName -role roleName -user userName
   ```

   Where:

   busName
   Is the name of the bus on the system you enter the command.

   foreignBusName
   Is the foreign bus to which you are adding the user.

userName
Is the userid to add to the foreign bus.

9. Verify the connection. Coordinate with the consuming administrator to recycle the servers involved with the connection by restarting the servers.

   You should see messages similar to:

```
[8/24/05 11:00:09:741 PDT] 00000086 SibMessage    I   [SCA.SYSTEM.WBIDev-BGMNode01Cell.Bus:WPSNode.server1-SCA.SYSTEM.WBIDev-BGMNode01Cell.Bus]
CWSIP0382I: messaging engine 2D7333574B0CD70B responded to subscription request, Publish Subscribe topology now consistent.
```

Start the applications.

Configuring the database manually

Certain property settings in the response file necessitate that you create database tables manually. If you set properties in the response file to delay the configuration of database tables, you need to run the generated SQL and DDL separately, after the product configuration.

Before you configure the databases manually, make sure of the following:

- You have created the databases and storage groups.
- You have run the installation script zSMPInstall.sh.
- You have run the configuration script and have chosen to delay the configuration of database table and data source definitions by accepting the default setting of **dbDelayConfig=true** in the response file.

  If the databases and storage groups have been created, the configuration script can connect to the database and create the database definitions automatically. However, if the database administrator at your site does not allow scripts to automatically update the database, you must accept the default value of **dbDelayConfig=true** .

  When you accept the default value of **dbDelayConfig=true** in the response file, the configuration script generates the SQL, *but does not execute the code to automatically create the database definitions*. Instead, the SQL is written out to a directory.

The database administrator, or a person with authority to create database definitions, performs the following steps to configure the database manually:

1. Access the generated SQL required to create the database table and data source definitions. The SQL to create the database table and data source definitions is written to the following directories:

   Note: The *variable portion* of the directory path represents the WebSphere Application Server for z/OS configuration HFS and app_server_root directory name combined, for example /WebSphere/V6R0M0/AppServer. This is sometimes referred to as WAS_HOME.

   Note: The path names below assume DB2 for z/OS version 8. If you are using DB2 for z/OS version 7, the portion of the directory path that contains the DB2 for z/OS product would vary accordingly. For example, instead of */DB2zOSV8*, the path name would include */DB2zOSV7*.

   - Business Process Choreographer SQL: *configuration_root/app_server_root*/profiles/default/dbscripts/ProcessChoreographer/DB2zOSV8/BPEDB/createDatabase.sql
   - Common Event Infrastructure SQL: *configuration_root/app_server_root*/profiles/default/dbscripts/CEI/DB2zOSV8/CEIDB/ceidef.sql

- Enterprise Service Bus SQL:*configuration_root/app_server_root*/profiles/default/dbscripts/EsbLoggerMediation/DB2zOSV8/ESBDB/Table_esb_DB2UDBOS390_V8_1.sql
- Common database SQL:*configuration_root/app_server_root*/profiles/default/dbscripts/CommonDB/DB2zOSV8/*dbname*

  Where dbname can be any one of the following values:
  - createTable_AppScheduler.sql
  - createTable_CommonDB.sql
  - createTable_customization.sql
  - createTable_lockmanager.sql
  - createTable_mediation.sql
  - createTable_Recovery.sql
  - createTable_Relationship.sql
  - insertTable_CommonDB.sql

2. Using the database tool of your choice, run the SQL to create the database tables.

   WebSphere Process Server for z/OS packages a utility named **DBUtility.sh**. This utility is located in the augmented WebSphere Application Server for z/OS configuration HFS, for example, /WebSphere/V6R0/AppServer/bin/DBUtility.sh.. If you choose to, you can use this utility run the SQL to create the database tables. Here is an example of the syntax used with the **DBUtility.sh** utility:

   ```
   /wasv6config/v2cell/dmgr/DeploymentManager/bin/DBUtility.sh createTable
   -DsqlScriptPath.default=/wasv6config/v2cell/dmgr/database/v2adropandcreate.sql
   -DdbType=DB2UDBOS390_V8_1 -DdbName=xxxxxV2
   -DprofilePath=/wasv6config/v2cell/dmgr/DeploymentManager/profiles/default
   -DdbJDBCProperties=/shared/db2810/jcc/properties
   -DdbConnectionLocation=WBDDB2
   -DdbJDBCClasspath=/shared/db2810/jcc/classes
   -DdbUserId=sysadm1
   -DdbPassword=sysadm1
   -DdbDelayConfig=false
   -DdbCreateNew=false
   ```

   Always consult with your database administrator on any utility that updates the database.

You have configured the database manually. You may also want to review the JDBC Provider and datasource definitions in the administrative console to further your understanding.

Configuring Business Process Choreographer

For information on how to configure Business Process Choreographer containers for business processes and human tasks, as well as how to configure Business Process Choreographer Explorer and Business Process Choreographer Observer, go to the WebSphere Process Server for z/OS, version 6.0.2, information center and review the topics under **Installing and configuring WebSphere Process Server > Configuring the software > Configuring Business Process Choreographer**. You can also find this information in the *Business Process Choreographer* PDF.

Verifying the WebSphere Process Server for z/OS installation and configuration as a stand-alone server

You can verify the WebSphere Process Server for z/OS configuration as a stand-alone server and ensure that your new runtime environment is prepared to host applications by performing a series of tasks.

Before verifying that WebSphere Process Server for z/OS has been installed and configured as a stand-alone server , you must:

- Complete the documented steps to install and configure as a stand-alone server WebSphere Process Server for z/OS.
- The database used is Cloudscape.

Access the following technical support document for instructions on how to verify the WebSphere Process Server for z/OS configuration:

- Performing Installation Verification for WPS on z/OS V6.0.1.

By following the steps in technical support document titled ″ Performing Installation Verification for WPS on z/OS V6.0.1″, you will have gained skills toward understanding the capabilities and the administration of WebSphere Process Server for z/OS. Having done so can help to prepare you for more advanced configurations which use DB2 for the runtime databases and which often involve a Network Deployment topology.

Applying product maintenance

Because WebSphere Process Server for z/OS is installed and configured into the WebSphere Application Server, the maintenance applied to WebSphere Process Server for z/OS is done so through the WebSphere Application Server product, using the WebSphere Application Server techniques for applying product maintenance.

Contact the IBM Software Support Center for information about preventive service planning (PSP) upgrades for the product. For more information about PSP upgrades for WebSphere Process Server for z/OS, see the *Program Directory for WebSphere Process Server for z/OS*. 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.

See *Applying product maintenance* in the WebSphere Application Server for z/OS information center for a description of how to apply product maintenance

You can maintain service to clients when upgrading the host cluster of WebSphere Application Server for z/OS.

Applying a service level or restoring to the previous accepted service level

Because WebSphere Process Server for z/OS is installed and configured into the WebSphere Application Server, the service level applied to WebSphere Process Server for z/OS is done so through the WebSphere Application Server product, using the WebSphere Application Server techniques for applying service level or restoring to the previous accepted service level

Service that is applied to the product data sets and product HFS occasionally requires corresponding changes to be made to the configuration HFS 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. See *Applying a service level or restoring to the previous accepted service level* in the WebSphere Application Server for z/OS information center for a description of how to apply service

Uninstalling

The Uninstalling section describes how to uninstall WebSphere Process Server for z/OS.

Overview

Like the install on z/OS, the uninstall is run from a command line.

Running the uninstall program may cause the administrative console to malfunction. Because of this, you should review the latest technote information on uninstalling WebSphere Process Server for z/OS before running the uninstall script.

You run the uninstall command using **zSMPInstall.sh** with the appropriate command arguments and parameters. Running the command **zSMPInstall.sh** with the `-uninstall` with appropriate command arguments and parameters restores the WebSphere environment back to the level it was at prior to installation.

Note: The Common Event Infrastructure and Business Process Choreographer components are uninstalled separately. For information on how to uninstall Common Event Infrastructure and Business Process Choreographer, consult the appropriate help topics.

The uninstall process results in the following actions:

- Disables WebSphere Process Server features by running Configuration Manager scripted actions.

  This will remove any administrative console plug-in extensions.
- Removes profile augmentation using WSPROFILE scripted actions.

  This will unaugment the WebSphere Application Server default profile.
- Deletes the post install file and removes code base permissions

Run the install script to uninstall WebSphere Process Server for z/OS

Run **zSMPInstall.sh** with the `-uninstall` keyword to remove the WebSphere Process Server for z/OS definitions from the installed configuration.

Running the uninstall program may cause the administrative console to malfunction. Because of this, you should review the latest technote information on uninstalling WebSphere Process Server for z/OS before running the uninstall script.

The WebSphere Process Server for z/OS install process assumes a working knowledge of z/OS UNIX System Services. You may want to have access to the following documentation:

- z/OS V1R7.0 UNIX System Services User's Guide
- z/OS V1R7.0 UNIX System Services Command Reference

Perform this task when your goal is to remove the WebSphere Process Server for z/OS definitions to restore your WebSphere environment to the level it was prior to installing and configuring WebSphere Process Server for z/OS.

1. Access the OS/390 UNIX command shell. Enter the TSO command OMVS at the ISPF Command Shell or TSO OMVS from any other ISPF panel.

   Once you're in the UNIX shell, a command prompt (usually a dollar ($) or pound (#) sign) indicates that the system is ready to accept input.
2. From the command prompt, run the uninstall command. Use the following syntax diagram as guide on how run the command:

Uninstall command syntax diagram

```
►►──zSMPInstall.sh──-smproot /usr/lpp/zWPS/V6R0──────────────────────────────────────────────────►

                                                                                              (1)
►──-runtime /WebSphere/V6R0M0/AppServer──-uninstall──-response──/usr/lpp/zWPS/V6R0/zos.config/standAloneProfile.rsp──────────────►◄
```

Notes:

1 Type in one of the absolute path names of the response file according to your configuration. The path file names above assume the uninstaller is using the default response files. If the response file was customized, the path name would need to represent the absolute path of the customized file.

Note: The instructions described above indicate how to run the install script from a OS/390 UNIX command shell. Alternatively you can run the install script from a prompt using Telnet. To run the install script using Telnet, type the following from the system prompt: **telnet TCPIP-Address port number**. Here is an example of a Telnet command: `telnet 99.9.99.999 2023`, where 99.9.99.999 is the TCP IP-Address and 2023 is the port number.

The product should be uninstalled for the appropriate configuration depending on the response file properties used.

The following will be accomplished with the -uninstall option.

- Disable product features by running Configuration Manager Scripted actions that remove the associated Administrative console plug-in extensions
- Un-augments the default profile.

Check the associated log file and trace file if the uninstall was not successful.

- **Standard out messages**

  Standard output messages display directly on the screen from which you ran the installer script to uninstall the product. You can choose to redirect these messages to a file by using redirect symbol and a file name at the end of the command line. For example, adding the syntax `>run.log` to the end of the install command redirect the standard output messages to the file named **run.log** in the present working directory. The standard out messages display as follows:

  ```
  parsing command arguments...
  parsing arguments complete
  setting up configuration...
  ```

```
runtimeRootDirName is: /WebSphere/V6R0M0/AppServer
WAS_HOME is: /WebSphere/V6R0M0/AppServer
WBI_HOME is: /WebSphere/V6R0M0/AppServer
running Configuration Manager for uninstall...
Configuration Manager for uninstall complete
unaugmenting profile(s)...
unaugmenting profile(s) complete
```

- **Log file**

  Log messages are written to the **zSMPInstall.log** file in the run-time directory. Standard location for this file is /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.log.
- **Trace file**

  Review the zSMPInstall.trace (ASCII) file in the run-time directory. Standard location for this file is /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.trace..

  There should be no error messages (i.e. messages with an ″E″ suffix) in the trace file.

Perform other troubleshooting tasks as follows:

- **Review the actions of the Update Configuration Manager task.**

  These actions are recorded by writing to a log file (ASCII). The log file name is `cmtInstall.log`.

  Standard location for this file is in directory /WebSphere/V6R0M0/AppServer/logs/wbi.

  Search this Configuration Manager log for `>SEVERE<` or `>WARNING<` level messages to determine overall error in processing.

  Each Ant script run from the install directory writes to it’s own log (ASCII).

  Default name for the install directory that contains the ant scripts is:/WebSphere/V6R0M0/AppServer/properties/version/install.wbi/6.0.0.0/config/full/uninstall.

  The resulting ant logs are written to the product log directory. The default name for this directory is /WebSphere/V6R0M0/AppServer/logs/wbi. Ant logs include the following (review these logs to determine errors in processing) :

  - 90SDeleteFirstStepsFilesWBI.ant.log
  - 90SRemoveJavaOptions.ant.log
  - 90SUninstallCEI.ant.log
  - 98SUndeployBPCAdminConsolePlugins.ant.log
  - 98SUndeployServerAdminConsolePlugins.ant.log
  - 99SUndeployCoreAdminConsolePlugins.ant.log

  Each of these logs should contain a Build Successful message.
- **Review the contents of the Unaugment Log**

  The unaugment profile task records it’s actions by writing to a log file (ASCII). The log file name has the name **wasprofile_unaugment_default.log.** Standard location for this file is in directory /WebSphere/V6R0M0/AppServer/logs/wasprofile. Search this WAS Profile Augment log for `>SEVERE<` or `>WARNING<` level messages to determine overall error in processing.

  There should be no SEVERE messages.

Uninstalling Business Process Choreographer

For information on how to remove the Business Process Choreographer component from a WebSphere Process Server installation, go to the WebSphere Process Server for z/OS, version 6.0.2, information center and review the topics under **Installing and configuring WebSphere Process Server > Uninstalling the software > Removing the Business Process Choreographer configuration**. You can also find this information in the *Business Process Choreographer PDF*.

Removing the Common Event Infrastructure configuration

If you need to remove the configuration for the Common Event Infrastructure, in preparation for uninstalling WebSphere Process Server, you must first remove the deployed enterprise applications and the database configuration.

The Common Event Infrastructure is installed in the base installation of WebSphere Process Server, but will not be active unless it is already configured. This topic tells you only how to remove a previously configured instance of the Common Event Infrastructure. To remove the configuration for the Common Event Infrastructure, follow these steps:

Removing the Common Event Infrastructure application

If you need to remove the event server enterprise application and resources from WebSphere Process Server, you can use the event-application.jacl script.

If you prefer, you can remove the event server enterprise applications manually using the administrative console rather than using the `event-application.jacl` script. If use the administrative console, you must also manually remove the Common Event Infrastructure resources. For more information about these resources, see *Default configuration* in the section on Configuring the Common Event Infrastructure.

To remove the event server enterprise application, use the wsadmin tool to run the `event-application.jacl` script.

To run the script on a Windows system, go to the *`profile_root`*`\event\application` directory and run the following command (all on one line):

```
wsadmin -f event-application.jacl -profile event-profile.jacl -action uninstall
-node node_name -server server_name
```

To run the script on a Linux or UNIX system, go to the *`profile_root`*`/event/application` directory and run the following command (all on one line):

```
wsadmin.sh -f event-application.jacl -profile event-profile.jacl
  -wsadmin_classpath install_root/event/lib/cei_installer.jar -action
uninstall -node node_name -server server_name
[-appname app_name] [-trace]
```

The `event-application.jacl` script uses these parameters:

node_name
: The WebSphere Process Server node from which you want to remove the event server enterprise application.

server_name
: The WebSphere Process Server from which you want to remove the event server enterprise application. This parameter is optional. If you do not specify a server, the enterprise application is removed from all servers in the node.

app_name
: The name of the deployed event server enterprise application you want to

remove. This parameter is optional. If you do not specify an application name, all registered Common Event Infrastructure enterprise applications are removed.

The optional **-trace** parameter causes additional debugging information to display on the standard output.

Removing the event messaging enterprise application

Before uninstalling the Common Event Infrastructure, you must remove the event messaging enterprise application.

To remove the event messaging enterprise application, use the wsadmin tool to run the `event-message.jacl` script.

The parameters of the `event-message.jacl` script are as follows:

node_name
: The WebSphere Process Server node from which you want to remove the event messaging enterprise application.

  To discover the node name, run the *profile_root*/bin/setupCmdLine and then the echo $WAS_NODE command.

server_name
: The WebSphere Process Server from which you want to remove the event messaging enterprise application. This parameter is optional. If you do not specify a server, the application is removed from all servers in the specified node.

app_name
: The name of the deployed messaging enterprise application you want to remove. This parameter is required; if you installed the Common Event Infrastructure with the default values, then the application name is `EventServerMdb`.

The optional **-trace** parameter causes additional debugging information to display on the standard output.

Removing the event database

If you need to remove the event database, you can use the provided scripts. You must remove the database before you uninstall the Common Event Infrastructure.

When the database is configured, the configuration script also creates scripts for removing the database and the Java database connectivity (JDBC) provider. The scripts for removing the event database are placed in database-specific subdirectories of the *profile_root*/event/dbscripts directory. The scripts for removing the JDBC provider are placed in database-specific subdirectories of the *profile_root*/event/dsscripts directory.

Note: The event database can be shared among multiple event servers using the same JDBC provider configuration. Therefore, remove the JDBC provider configuration only if you have uninstalled the associated event database.

To remove the event database and JDBC provider, run the appropriate scripts from the following table.

Type	Operating system	Database script	JDBC configuration script
Cloudscape	Windows	`rm_event_cloudscape.bat`	`rm_cloudscape_jdbc_provider.bat`

Type	Operating system	Database script	JDBC configuration script
Cloudscape	Linux/UNIX	`rm_event_cloudscape.sh`	`rm_cloudscape_jdbc_provider.sh`
DB2	Windows	`rm_event_db2.bat`	`rm_db2_jdbc_provider.bat`
DB2	Linux/UNIX	`rm_event_db2.sh`	`rm_db2_jdbc_provider.sh`
Oracle	Windows	`rm_event_oracle.bat`	`rm_oracle_jdbc_provider.bat`
Oracle	Linux/UNIX	`rm_event_oracle.sh`	`rm_oracle_jdbc_provider.sh`
Cloudscape	z/OS (Windows script)	`rm_event_cloudscape.bat`	`rm_cloudscape_jdbc_provider.bat`
Cloudscape	z/OS (Linux/UNIX script)	`rm_event_cloudscape.sh`	`rm_cloudscape_jdbc_provider.sh`
DB2	z/OS (Windows script)	`rm_event_db2zos.bat`	`rm_db2zos_jdbc_provider.bat`
DB2	z/OS (Linux/UNIX script)	`rm_event_db2zos.sh`	`rm_db2zos_jdbc_provider.sh`

You can remove the event database or JDBC provider at any time by running the appropriate script. To remove the JDBC provider, use the appropriate script and specify the scope in which you want to remove the JDBC provider:

```
rm_db_jdbc_provider scope [server_name]
```

The generated scripts use these parameters:

scope The scope in which you want to remove the JDBC provider. The valid values are `cell`, `node`, and `server`.

server_name
The name of the WebSphere Process Server from which you want to remove the JDBC provider, if **scope** is `server`. (If **scope** is `cell` or `node`, this parameter is ignored.)

Troubleshooting the installation and configuration

If the product installation and configuration are not successful, use this information to help you assess and correct the problems.

This topic assumes that you have attempted to install and or configure but have not been successful.

You should be aware that the installer program records multiple indicators of success in the following ways:

- Standard output messages

  Standard output messages display directly on the screen from which you run the installation script (zSMPInstall.sh), or configuration script(s) zWPSConfig.sh / zWESBConfig.sh. You can choose to *redirect* these messages to a file by using redirect symbol and a file name at the end of the command line. For example, adding the syntax `>run.log` to the end of the installation command will redirect the standard output messages to the file named **run.log** in the present working directory.
- Log file messages

  Log messages for installation are written to the **zSMPInstall.log** file in the runtime directory. Standard location for this file is /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.log.

Log messages for configuration are written to the **zWPSConfig.log** or the **zWESBConfig.log** file in the runtime directory. Standard locations for these files are /WebSphere/V6R0M0/AppServer/logs/wbi/zWPSConfig.log and /WebSphere/V6R0M0/AppServer/logs/wbi/zWESBConfig.log respectively.

- Trace file messages

  The installation trace messages are written to the **zSMPInstall.trace** file in the runtime directory. Standard location for this file is /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.trace.

  Trace messages for configuration are written to the **zWPSConfig.trace** or the **zWESBConfig.trace** file in the runtime directory. Standard locations for these files are /WebSphere/V6R0M0/AppServer/logs/wbi/zWPSConfig.trace and /WebSphere/V6R0M0/AppServer/logs/wbi/zWESBConfig.trace respectively.

Make sure that you have installed and configured WebSphere Application Server for z/OS successfully. Refer to the installation troubleshooting information in the WebSphere Application Server for z/OS information center if you are having trouble installing and configuring WebSphere Application Server for z/OS.

For current information available from IBM Support on known problems and their resolution, see the WebSphere Process Server Support page.

1. Review the messages from Standard Out. There should be no error messages displayed. The standard output messages display on either the screen from which you ran the installation command or in a file that you specified by using the redirect (">") symbol on the command line.

   The following is an example of a successful execution of the installation script with the **-install** option:

   ```
   parsing command arguments...
   parsing arguments complete
   setting up configuration...
   runtimeRootDirName is: /WebSphere/V6R0M0/AppServer
   WAS_HOME is: /WebSphere/V6R0M0/AppServer
   WBI_HOME is: /WebSphere/V6R0M0/AppServer
   set up configuration complete
   creating the symbolic links...
   invokeSymLink
   creation of symbolic links complete
   doing post install file updates...
   post install updates complete
   running Configuration Manager update...
   Configuration Manager update complete
   augmenting profile(s)...
   augmenting profile(s) complete
   ```

   Error messages indicate an unsuccessful installation. Some errors as displayed in Standard Out will be self explanatory and can be easily corrected. If the installation completed to the point where the log and trace files were created, continue with the following steps.

2. Review the zSMPInstall.log (ASCII) file in the runtime directory. For a stand-alone configuration, the standard location for this file is /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.log. For a deployment manager configuration, the standard location for this file is /WebSphere/V6R0M0/DeploymentManager/logs/wbi/zSMPInstall.log

   If there are error messages, try to determine which of the following tasks were in progress when the error occurred.

   - create symbolic links
   - create post installation file
   - update codebase permissions

- update Configuration Manager

Knowing the task that was in progress at the time of an error will help you assess the information in the trace file.

3. Review the zSMPInstall.trace or the zWPSConfig.trace / zWESBConfig.trace (ASCII) file in the runtime directory. For a stand-alone configuration, the standard location for these files are /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.trace or /WebSphere/V6R0M0/AppServer/logs/wbi/zWPSConfig.trace For a deployment manager configuration, the standard location of this file is /WebSphere/V6R0M0/DeploymentManger/logs/wbi/zSMPInstall.trace or /WebSphere/V6R0M0/DeploymentManger/logs/wbi/zWPSConfig.trace

   On a successful installation, only informational messages (messages with a suffix of **I**, for example CWPIZ0044**I**) should be listed in the trace file.

   If warning messages (messages with a suffix of **W**) or error messages (messages with a suffix of **E**) are listed in the trace, further review is required.

   If the warning or error occurred during the create symbolic links, create post installation file, or update codebase permissions tasks, the trace message should contain information that will help you diagnose and correct the problem.

   If the warning or error occurred in the update Configuration Manager task, proceed to the next step.

   If the warning or error occurred in the augment profile(s) task, proceed to step 5.

4. Review the actions of the Update Configuration Manager task. These actions are recorded by writing to a log file (ASCII). The log file name is `cmtInstall.log`.

   Standard location for this file is in directory /WebSphere/V6R0M0/AppServer/logs/wbi.

   Search the Configuration Manager log for `>SEVERE<` or `>WARNING<` level messages to determine overall error in processing.

   Each Ant script run from the installation directory writes to its own log (ASCII).

   To determine what Ant file was running at the time of the error, you can look for 'Buildfile' previous.

   The default name for the installation directory that contains the ant scripts is: /WebSphere/V6R0M0/AppServer/properties/version/install.wbi/6.0.0.0/config/full/install.

   The resulting ant logs are written to the product log directory. The default name for this directory is /WebSphere/V6R0M0/AppServer/logs/wbi. Ant logs include the following (review these logs to determine errors in processing) :

   ```
   100SUgradeCoreAdminConsolePlugins.ant.log
   101SUgradeServerAdminConsolePlugins.ant.log
   101SWbiWebuiUpgrade.ant.log
   102SUpgradeIscdeploy.ant.log
   90SConfigNoProfileFirstStepsESB.ant.log
   90SConfigNoProfileFirstStepsWBI.ant.log
   90SConfigureWSProfileForWBI.ant.log
   90SInstallCEI.ant.log
   90SUpdateJavaOptions.ant.log
   91SConfigNoProfileFirstStepsCharset.ant.log
   98SDeployBPCAdminConsolePlugins.ant.log
   98SDeployServerAdminConsolePlugins.ant.log
   99SDeployCoreAdminConsolePlugins.ant.log
   99WbiProfileUpgrade.log
   ```

If there were no problems a **BUILD SUCCESSFUL** message displays at the end of the file.

5. Review the actions of the WebSphere Application Server profile augment task. The augment profile(s) task records its actions by writing to a log file (ASCII). The log file name has the name **wasprofile_augment_default.log**. Standard location for this file is in the directory /WebSphere/V6R0M0/AppServer/logs/wasprofile.

   Search the Profile Augmentation log for `>SEVERE<` or `>WARNING<` level messages to determine overall error in processing.

   Individual Ant action logs are located in /WebSphere/V6R0M0/AppServer/profiles/default/logs.

   To determine what Ant file was running at the time of the error, you can look for 'Buildfile' previous.

After troubleshooting the problems that caused the installation errors, and after you have run the installation script successfully, you should perform the following steps:

1. Start the WebSphere Application Server.
2. Launch the Administrative Console and verify that the product components have been installed.

   For example, by installing WebSphere Process Server successfully, you should see evidence of Process Choreographer under the Enterprise Applications with names that start with `BPEContainer`, `BPCExplorer`, and `TaskContainer`.

Message reference for WebSphere Process Server for z/OS installation and configuration

The message reference for WebSphere Process Server for z/OS lists the message codes that may display while running the install script or when running the configuration script.

About the installation error messages

Use the data in the Explanation and User response fields to troubleshoot the WebSphere Process Server for z/OS message codes.

The message code displays as CWPIZyyyyz, where:

- CWPIZ = The WebSphere Process Server for z/OS message prefix
- yyyy = The numeric identifier assigned to the number
- z = Descriptor (E, I or W) for the type of message, where:
  - E = Error message
  - I = Informational message
  - W = Warning message

The WebSphere Process Server for z/OS installation error messages are documented in the information center under **Reference > Messages >CWPIZ**.

The WebSphere Process Server for z/OS installation error messages are written to the zSMPInstall.log file in the run-time directory. The standard default location for the log file is /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.log.

The WebSphere Process Server for z/OS configuration error messages are written to the zWPSConfig.log file and the zWESBConfig.log file in the run-time directory.

The standard default location for these log files are /WebSphere/V6R0M0/AppServer/logs/wbi/zWESBConfig.log and /WebSphere/V6R0M0/AppServer/logs/wbi/zWPSConfig.log respectively.

Log files

Various log files are created during the product installation and configuration process.

Purpose

Consult the applicable logs if problems occur during the product installation and configuration process.

Standard out messages redirected to log file

Standard out messages report high-level actions such as the starting and completing of the action that verifies the command line arguments.

By default, these messages display directly on the screen from which you run the product installation script. However, you can *redirect* these messages to a file by using the redirect symbol and a file name at the end of the command line. For example, specifying `>run.log` at the end of the installation command redirects the standard out messages to a file named `run.log` in the present working directory.

Standard out messages also report severe errors that occur prior to the Log and Trace File being opened. For instance, the following message block displays if a required keyword (-runtime) was not included in the installation command.

```
parsing command arguments...
CWPIZ0101E -runtime keyword and value not specified on command line.
com.ibm.ws390.installer.InstallFailureException: -runtime keyword and value not specified
CWPIZ0017E  install task failed.
```

Log file

These messages include the messages written to Standard Out, but provide additional information and settings that were used by the installer program.

For instance, the following log portion shows the response properties and their values being used. It also shows the source and target directories being used during the creation of the symbolic links.

```
response property: profilePath=/WebSphere/V6R0M0/AppServer/profiles/default
response property: nodeName=SY1
response property: scaSecurityPassword=ibmuser
response property: dbType=Cloudscape
response property: ceiSampleJmsUser=ibmuser
response property: scaSecurityUserId=ibmuser
response property: configureScaSecurity=true
response property: mqUser=ibmuser
response property: serverName=server1
response property: adminBFMGroups=ibmuser
response property: profileName=default
response property: dbCreateNew=true
response property: ceiSampleJmsPwd=ibmuser
response property: cellName=SY1
response property: dbLocation=/WebSphere/V6R0M0/AppServer/cloudscape/databases/WBIDB
response property: mqPwd=ibmuser
response property: was.install.root=/WebSphere/V6R0M0/AppServer
response property: augment=
response property: ceiDbProduct=CLOUDSCAPE_V51_1
response property: wbi.install.root=/WebSphere/V6R0M0/AppServer
response property: ceiSampleServerName=server1
response property: templatePath=/WebSphere/V6R0M0/AppServer/profileTemplates/default.*
response property: dbName=WBIDB
```

```
set up configuration complete
creating the symbolic links...
Source=/usr/lpp/zWPS/V6R0M0

Target=/WebSphere/V6R0M0/AppServer
creation of symbolic links complete
doing post install file updates...
post install updates complete
running Configuration Manager update...
Configuration Manager update complete
```

Trace file

These messages are written to the **zSMPInstall.trace** file in the run-time directory.

The example below shows some preliminary informational messages and then a **CWPIZ0322E** error indicating that the required profileName property was not found in the response file that the user specified on the installation script command line (nor was provided as a -Z override).

The subsequent **CWPIZ0017E** error message is a general message indicating the final outcome of the zSMPInstall.sh run.

```
[8/16/05 17:00:45:380 EDT] 0000000a ManagerAdmin  I   BBOO0222I:
TRAS0017I: The startup trace state is *=info.

[8/16/05 17:00:48:230 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0044I Begin install task.

[8/16/05 17:00:48:273 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0117I WPS installer log data will be written to
/WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.log.

[8/16/05 17:00:48:282 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0024I WPS installer trace data will be written to
/WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.trace.

[8/16/05 17:00:48:292 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0014I Trace specification is "*=all=disabled".

[8/16/05 17:00:48:298 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0045I WPS SMP/E root directory is /zrockuser/wbi/Install.

[8/16/05 17:00:48:302 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0052I WAS SMP/E root directory is /web/usr/lpp/zWebSphere/V6R0.

[8/16/05 17:00:48:307 EDT] 0000000a WPSInstaller  I   BBOO0222I:
CWPIZ0046I Destination application server root directory is
/WebSphere/V6R0M0/AppServer.

[8/16/05 17:00:48:314 EDT] 0000000a WPSInstaller  E   BBOO0220E:
CWPIZ0322E profileName property not specified in Response File.

[8/16/05 17:00:48:318 EDT] 0000000a WPSInstaller  E   BBOO0220E:
CWPIZ0017E install task failed.

A trace file from a zSMPInstall.sh executed with the trace specification argument
set to "*=all=enabled" provides additional debugging information. It may contain
information that is meaningful only to a developer. The following is a partial
trace using "*=all=enabled":

************ Start Display Current Environment ************
Host Operating System is z/OS, version 01.04.00
Java version = J2RE 1.4.2 IBM z/OS Persistent Reusable VM build cm142-20050623
(JIT enabled: jitc), Java
Compiler = jitc, Java VM name = Classic VM
was.install.root = /WebSphere/V6R0M0/AppServer
user.install.root = /WebSphere/V6R0M0/AppServer/profiles/default
Java Home = /web/usr/lpp/zWebSphere/V6R0/java/J1.4
ws.ext.dirs = /WebSphere/V6R0M0/AppServer/java/lib:/WebSphere/V6R0M0/AppServer/java/lib/
ext:/WebSphere/V6R0M0/AppServer/classes:/WebSphere/V6R0M0/AppServer/lib:/WebSphere/V6R0M0/AppServer/
installedChannels:/WebSphere/V6R0M0/AppServer/lib/ext:/WebSphere/V6R0M0/AppServer/deploytool/itp
/plugins/com.ibm.etools.ejbdeploy/runtime:/WebSphere/V6R0M0/AppServer/MQSeries/pubsubroot/lib
Classpath = /zrockuser/bbzconfig.jar:/WebSphere/V6R0M0/AppServer/lib/admin.jar:/WebSphere/V6R0M0
/AppServer/lib/ant.jar:/WebSphere/V6R0M0/AppServer/lib/bootstrapws390.jar:/WebSphere/V6R0M0
/AppServer/lib/bootstrap.jar:/WebSphere/V6R0M0/AppServer/lib/configmanager.jar:/WebSphere
/V6R0M0/AppServer/lib/emf.jar:/WebSphere/V6R0M0/AppServer/lib/ras.jar:/WebSphere/V6R0M0
/AppServer/lib/runtimefw.jar:/WebSphere/V6R0M0/AppServer/lib/utils.jar:/WebSphere/V6R0M0
/AppServer/lib/wasjmx.jar:/WebSphere/V6R0M0/AppServer/lib/wasproduct.jar:/WebSphere/V6R0M0
/AppServer/lib/wccm_base.jar:/WebSphere/V6R0M0/AppServer/lib/wjmxapp.jar:/WebSphere/V6R0M0
/AppServer/lib/wsanttasks.jar:/WebSphere/V6R0M0/AppServer/lib/wsexception.jar:/WebSphere
/V6R0M0/AppServer/lib/wsprofile.jar:/WebSphere/V6R0M0/AppServer/profiles/default/properties:
```

```
/WebSphere/V6R0M0/AppServer/properties:/WebSphere/V6R0M0/AppServer/lib/bootstrap.jar:/WebSphere
/V6R0M0/AppServer/lib/j2ee.jar:/WebSphere/V6R0M0/AppServer/lib/lmproxy.jar:/WebSphere/V6R0M0
/AppServer/lib/urlprotocols.jar:/WebSphere/V6R0M0/AppServer/lib/bootstrapws390.jar
Java Library path = /web/usr/lpp/zWebSphere/V6R0/java/J1.4/bin/classic/libjvm.so:/web/usr
/lpp/zWebSphere/V6R0/java/J1.4/bin/classic:/web/usr/lpp/zWebSphere/V6R0/java/J1.4/bin/:
/WebSphere/V6R0M0/AppServer/lib:/WebSphere/V6R0M0/AppServer/lib:/WebSphere/V6R0M0/AppServer
/MQSeries/pubsubroot/lib:/mqm/java/bin:/mqm/java/lib:/db2810/lib:/db2beta/db2710/lib:
/web/usr/lpp/WebSphere/lib:/lib:/usr/lib:/java/J1.3/bin:/java/J1.4/bin:/java/J5.0/bin:
/staf/lib:/WebSphere/V6R0M0/AppServer/lib:/usr/lib
Current trace specification = *=all
************* End Display Current Environment *************
[10/3/05 16:35:05:709 EDT] 0000000a ManagerAdmin  I   BBOO0222I: TRAS0017I:
The startup trace state is *=all.
[10/3/05 16:35:08:638 EDT] 0000000a WPSInstaller  >  setup Entry
                                    /web/usr/wbi/zWebSphere/V6R0
                                    APPSERVER
                                    zSMPInstall.sh
                                    -smproot
                                    /web/usr/wbi/zWPS/V6R0
                                    -runtime
                                    /WebSphere/V6R0M0/AppServer
                                    -response
                                    /web/usr/wbi/zWPS/V6R0/zos.config/standAloneProfile.rsp
                                    -prereqonly
                                    -trace
                                    *=all=enabled
[10/3/05 16:35:08:640 EDT] 0000000a WPSInstaller  3   logFileDeleted
                                    true
[10/3/05 16:35:08:660 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0044I:
Begin install task.
[10/3/05 16:35:08:702 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0117I:
WPS installer log data will be written to /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.log.
[10/3/05 16:35:08:712 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0024I:
WPS installer trace data will be written to /WebSphere/V6R0M0/AppServer/logs/wbi/zSMPInstall.trace.
[10/3/05 16:35:08:722 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0014I:
Trace specification is "*=all=enabled".
[10/3/05 16:35:08:726 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0052I:
WAS SMP/E root directory is /web/usr/lpp/zWebSphere/V6R0.
[10/3/05 16:35:08:730 EDT] 0000000a WPSInstaller  >  checkPathName Entry
                                    /web/usr/wbi/zWPS/V6R0
[10/3/05 16:35:08:731 EDT] 0000000a WPSInstaller  <  checkPathName Exit
[10/3/05 16:35:08:732 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0045I:
WPS SMP/E root directory is /web/usr/wbi/zWPS/V6R0.
[10/3/05 16:35:08:736 EDT] 0000000a Symlink       >  isSymlink Entry
                                    /web/usr/wbi/zWPS/V6R0
[10/3/05 16:35:08:737 EDT] 0000000a Symlink       3   absolute path
                                    /web/usr/wbi/zWPS/V6R0
[10/3/05 16:35:08:737 EDT] 0000000a Symlink       3   canonical path
                                    /web/usr/wbi/zWPS/V6R0
[10/3/05 16:35:08:738 EDT] 0000000a Symlink       <  isSymlink Exit
                                    false
[10/3/05 16:35:08:738 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0046I:
Destination application server root directory is /WebSphere/V6R0M0/AppServer.
[10/3/05 16:35:08:744 EDT] 0000000a WPSInstaller  I   BBOO0222I: CWPIZ0247I:
Response file is /web/usr/wbi/zWPS/V6R0/zos.config/sample.rsp.
[10/3/05 16:35:08:764 EDT] 0000000a WPSInstaller  3   response property
                                    profilePath=/WebSphere/V6R0M0/AppServer/profiles/default
[10/3/05 16:35:08:765 EDT] 0000000a WPSInstaller  3   response property
                                    nodeName=SY1
```

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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
577 Airport Blvd., Suite 800
Burlingame, CA 94010
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States or other countries, or both: IBM, IBM (logo), AIX, CICS, Cloudscape, DB2, DB2 Connect, DB2 Universal Database, developerWorks, Domino, IMS, Informix, iSeries, Lotus, MQSeries, MVS, OS/390, Passport Advantage, pSeries, Rational, Redbooks, Tivoli, WebSphere, z/OS, zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

This product includes software developed by the Eclipse Project (http://www.eclipse.org/).

IBM WebSphere Process Server for z/OS version 6.0.2

Printed in USA