



IBM Software Group

WebSphere® Process Server for z/OS® V6.0.1 WebSphere Enterprise Service Bus for z/OS V6.0.1

Installation and Configuration Overview



@business on demand.

© 2006 IBM Corporation
Updated July 25, 2006

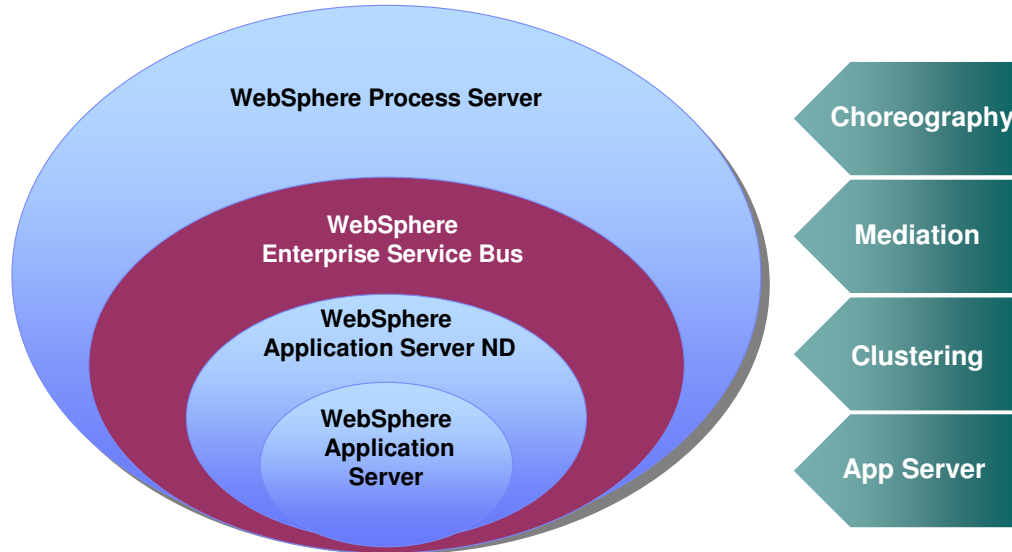
This presentation will cover installation of WebSphere Process Server for z/OS V6.0.1 and WebSphere Enterprise Service Bus for z/OS V6.0.1.

Goals

- Describe prerequisites and product packaging
- Describe WebSphere Process Server for z/OS and WebSphere Enterprise Service Bus V6.0.1 configuration process

The goal is to look at the pre-requisites and packaging of the products and then look at the installation of the products at a high level.

WebSphere Application Server, ESB, and Process Server



This slide is meant to show where the WebSphere Process Server and the WebSphere Enterprise Service Bus fit in the product stack. Notice that the WebSphere Enterprise Service Bus is built on top of the WebSphere Application Server which provides the Java™ 2 Enterprise Edition framework for the stack products. The WebSphere Enterprise Service Bus adds Mediation capabilities to the picture, allowing mediation of message flows between service requestors and providers. WebSphere Process Server is built on top of that, which means that the WebSphere Process Server includes the Mediation functions found in the WebSphere Enterprise Service Bus. The WebSphere Process Server also adds Choreography capabilities for business process management applications.

Section

Prerequisites and product packaging

This next section will look briefly at the prerequisites for the products and how the products are packaged.

Prerequisites (derived from base WebSphere)

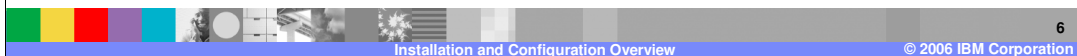
- Hardware requirements
 - ▶ Any hardware that supports z/OS 1.4 or higher or z/OSe 1.4 or higher
- Software requirements
 - ▶ z/OS 1.4 or higher or z/OSe 1.4 or higher with the following features installed, enabled and configured
 - z/OS Communications Server (TCP/IP) or equivalent
 - z/OS UNIX® System Services and the hierarchical file system (HFS)
 - SecureWay Security Server (RACF) or equivalent security management product
 - System logger
 - System SSL security required when using SSL
 - Workload management in goal mode
 - Resource recovery services



The requirements for WebSphere Process Server for z/OS V6.01 and WebSphere Enterprise Service Bus for z/OS V6.0.1 come from the Base WebSphere that the Process Server or the Enterprise Service Bus will be installed on top of. Notice that z/OS 1.4 is the minimum z/OS level but if you plan to exploit any zAAPs in your configuration, you will need a z990 GA3 or z890 and will need to run z/OS 1.6. The features listed are needed for the base application server configuration.

Implementation Options (Optional)

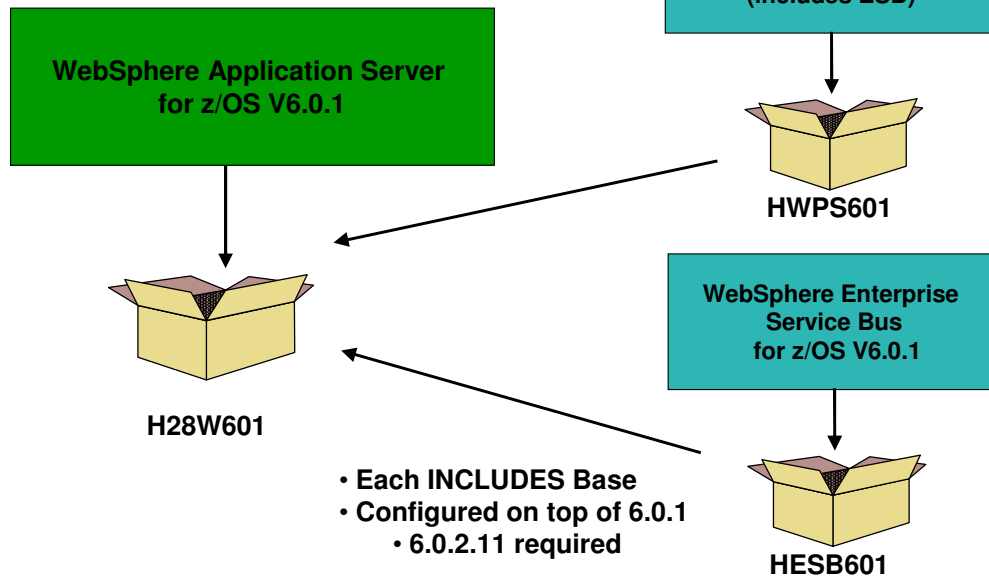
- IBM HTTP Server for z/OS or equivalent
- DB2® Universal Database Server for z/OS, Version 7 or higher
 - ▶ Need stored procedure builder enabled (DSNTPSMP) if using Relationships
- CICS® TS 2.2.0
- Information Management System IMS V8.1 or later; IMS Connect V2.2
- WebSphere MQSeries® for z/OS V5.3.1 and higher
- LDAP
- Tivoli® Access Manager (TAM) V5.1
- WebSphere Integration Developer, V6.0.1.1



Many of the implementation options shown here, such as CICS and IMS, are determined by the applications that are run on the server. In the case of DB2, however, if you plan to implement a Network Deployment solution with WebSphere Process Server or WebSphere Enterprise Service Bus, DB2 becomes a requirement. WebSphere Process Server and WebSphere Enterprise Service Bus require many databases which will be shown on a later slide. DB2 V7 is sufficient for the Process Choreographer function in WebSphere Process Server but may pose limitations for Common Event Infrastructure, Business Rules and Selector functions. DB2 v8 is recommended. The Stored Procedure Builder (DSNTPSMP) in DB2 also needs to be enabled if using Relationships.

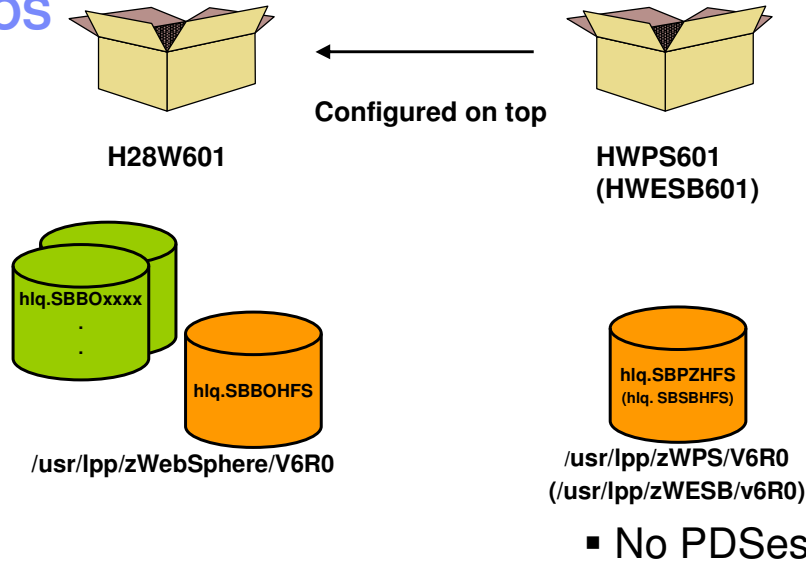
The tools to build the applications to be run in a WebSphere Process Server environment are provided by the WebSphere Integration Developer.

Packaging



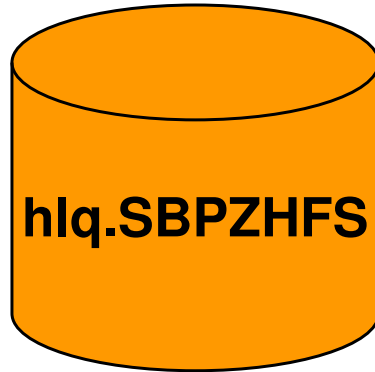
This slide shows the packaging of the products. WebSphere Process Server for z/OS V6.0.1 and WebSphere Enterprise Service Bus for z/OS V6.0.1 are built on top of WebSphere Application Server for z/OS V6.0.1, as shown on a previous slide. When you order either product, you will receive a copy of the base WebSphere Application Server for z/OS at the 6.0.2.9 PTF level. You should then order PTFs to bring your configuration to the 6.0.2.11 level, which is the required level. There are two new FMIDs for the products, HWPS601 and HESB601 as shown on the slide.

Structure for WebSphere Process Server for z/OS and WebSphere Enterprise Service Bus for z/OS



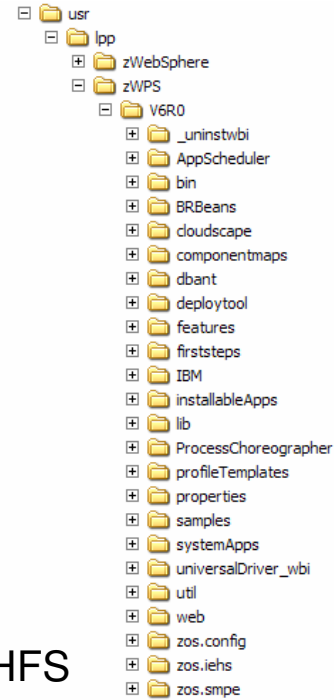
WebSphere Process Server for z/OS and WebSphere Enterprise Service Bus for z/OS are pure Java implementations so there is no need for PDSes in the products. The code is fully contained in the HFses shown. Unlike the WebSphere Business Integration Server Foundation product, there is no need for the high-level qualifier (HLQ) for the stack products to be the same as the high-level qualifier used for the WebSphere base product.

SMP/E directory configuration

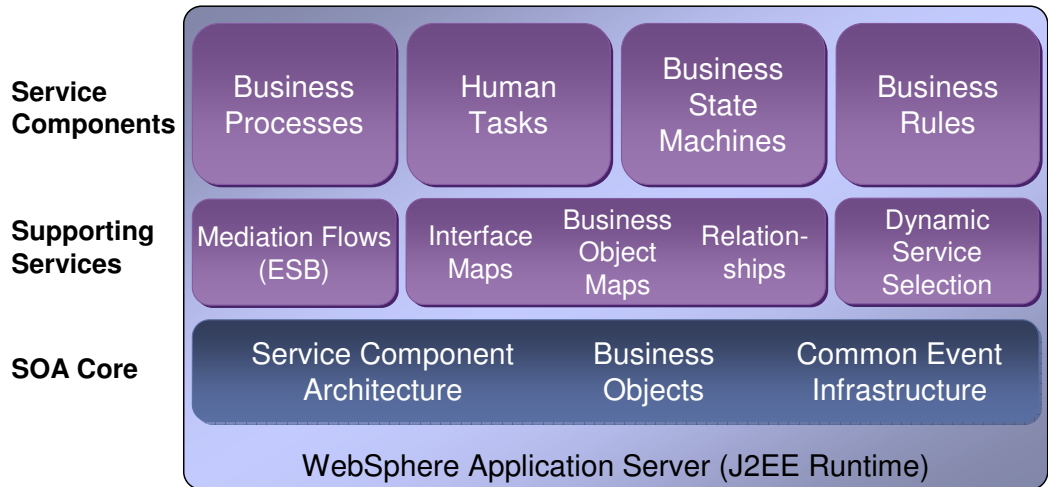


/usr/lpp/zWPS/V6R0

Or ESB-only configuration:
/usr/lpp/zWESB/V6R0 → hlq.SBSBHFS



WebSphere Process Server for z/OS V6.01 Components



This slide provides an overview of the components that make up the WebSphere Process Server and are enabled by the WebSphere Integration Developer.

The WebSphere Process Server is built on WebSphere Application Server Version 6, providing a robust J2EE application server runtime with capabilities that the process server implementation can exploit such as JMS messaging and enterprise beans. It can also make use of the application server qualities of service such as transactions, security and clustering. Overall, this provides a well proven runtime environment for WebSphere Process Server.

The Service Oriented Architecture (SOA) Core is the foundation in WebSphere Process Server. The main components of the SOA Core are the Service Component Architecture (SCA), Business Objects (BOs) and the Common Event Infrastructure (CEI).

SCA is the uniform programming and invocation model for business services that publish or operate on business data.

Business Objects (BOs) represent the data that is passed within that framework. Business objects are extensions to Service Data Objects (SDOs), which carry additional information needed for some integration scenarios.

The Common Event Infrastructure (CEI) provides the foundation architecture for the management and handling of events produced by business processes. This is essential for enabling the monitoring of business processes with products such as the WebSphere Business Monitor.

On top of the SOA Core are a set of Supporting Services, which provide the transformation primitives required by integration scenarios built using Service Component Architecture and Business Objects. WebSphere Enterprise Service Bus is found in this layer and provides Mediation capabilities for the WebSphere Process Server. As noted, WebSphere Enterprise Service Bus is also a separate product.

Interface maps are used to enable components making use of a particular interface to make calls to a component that provides a semantically similar but syntactically different interface.

Business object maps enable the transformation of business data between fields of Business Objects representing the same business entity but of differing types.

Relationships enable the correlation and synchronization of data representing the same business entity stored in multiple back end systems.

Selectors provide for a dynamic invocation of a target component based on a date and time criteria.

Business Processes are a fundamental part of the programming model, providing the composition aspect of the programming model. In WebSphere Process Server, the business processes are defined using BPEL. Business processes provide an implementation of a process model that describes the logical order in which the different activities of the process are being run, making calls out to the individual SCA services that implement the specific activities. As a result, a business process is the set of business-related activities, rules and conditions that are invoked in a defined sequence to achieve a business goal.

Human Tasks are enabled by the Human Task Manager and provide the human task capabilities for WebSphere Process Server.

Human Tasks allow people to participate in a business process in a machine-to-human scenario, a human-to-machine scenario and in a human-to-human scenario.

In the machine-to-human scenario an automated process creates tasks for people who participate in the business process, whereas the human-to-machine scenario allows a person to create a task that is run by an automated service. The human-to-human scenario allows a task to be created by a person for another person.

Human tasks can be integrated directly into the BPEL for a business process or can be packaged as an SCA component for use by any client that can invoke an SCA component.

Business State Machines are another way of modeling a business process. There are some processes that are highly event driven and are well suited to being thought of in terms of a state transition diagram. For example, a business process for order processing has to deal with the fact that an order can be canceled at any time during the order process up until the order is actually shipped. It can be difficult to model these kinds of processes in a business process. The Business State Machine component allows you to model the business process using similar constructs as UML 2.0 state machine support, and then generates BPEL for the implementation.

Business rules are a means of implementing and enforcing business policy through externalization of business function. Externalization enables the business rules to be managed independently from other aspects of an application. This independence allows for dynamic updating capabilities of the business rules to provide for a more agile business. There are two styles of business rules, if then rulesets and decision tables.

A Web client is provided where the parameters of business rules can be changed by a business user using a natural language specification of these rules rather than requiring an application developer or integration developer to change the application.

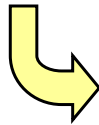
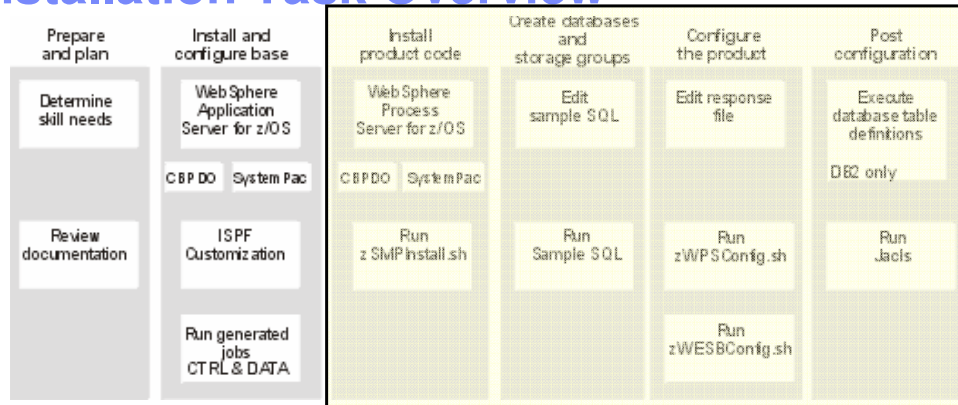
Section

Installation/Configuration Overview



The next section will look at an overview of the installation and configuration of the WebSphere Process Server for z/OS V6.0.1 and the WebSphere Enterprise Service Bus for z/OS V6.0.1.

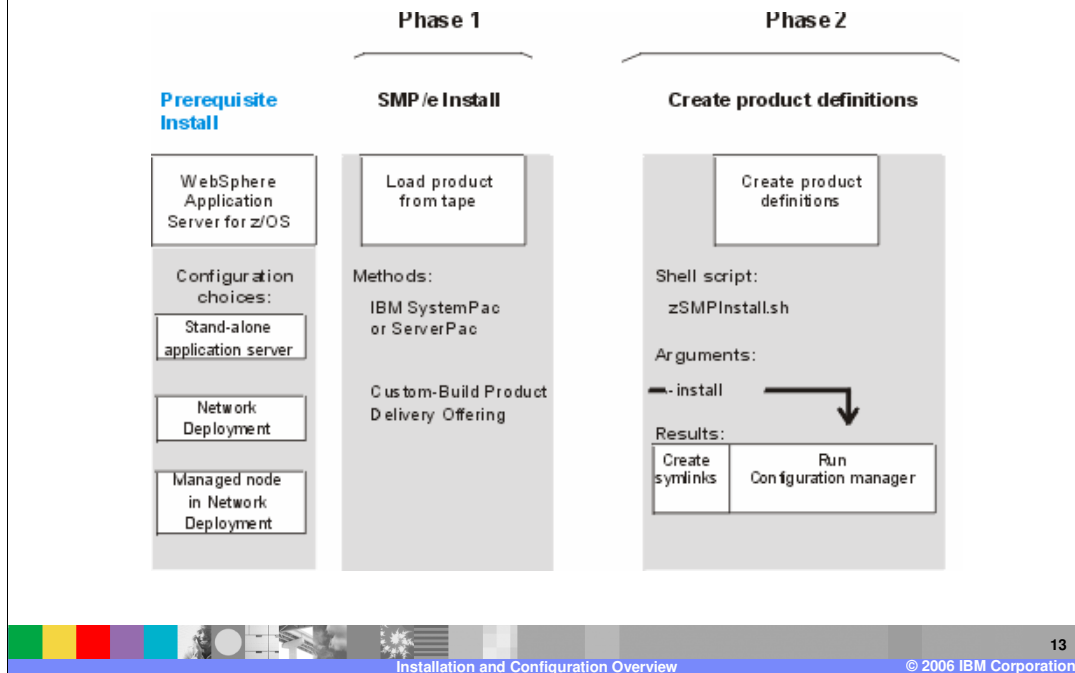
Installation Task Overview



- Stand-Alone Application Server
- OR-
- Deployment Manager/Empty Managed Node

Shown here is the Installation Task Overview. The WebSphere Process Server and WebSphere Enterprise Service Bus configuration is highlighted. Note that you are installing on top of a base WebSphere Application Server Configuration. This has to be either a Stand-Alone Application Server OR a Deployment Manager/Empty Managed Node combination. You CANNOT federate a Stand-Alone Application Server into a Deployment Manager Cell at this point... Highlighted functions will be the topic of this education. This education assumes you have installed and configured a base application server already.

Install product code



As with the Base Application Server installation, the WebSphere Process Server and WebSphere Enterprise Service Bus is really a two-phase operation. The first thing that needs to be done is the SMP/E installation of the product code. Once you have a base WebSphere Application Server defined, you are then ready to create product definitions for WebSphere Process Server or WebSphere Enterprise Service Bus. To do this, you will run the zSMPInstall.sh script which will create symlinks back to the WebSphere Process Server or WebSphere Enterprise Service Bus product code in your base WebSphere Application Server configuration. The zSMPInstall.sh script will also run the Configuration Manager which runs many Ant scripts, resulting in the administrative console being updated with WebSphere Process Server or WebSphere Enterprise Service Bus definitions.

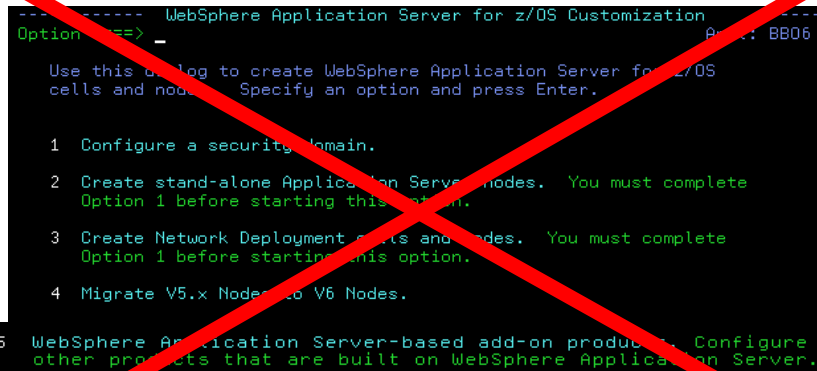
Configuration - Two step process

1. 'Install' WebSphere Process Server or WebSphere Enterprise Service Bus into your profile ('Phase 2' on previous slide)
 - ▶ Links <WAS_HOME> to the WebSphere Process Server SMP/E code
 - ▶ Updates the administration console
 - ▶ Runs applyPTF as well
2. 'Augment' the profile to use the new function
 - ▶ Create resource definitions
 - ▶ Create resources (databases, buses, queues)
 - ▶ Deploy needed applications

Configuring a server to include WebSphere Process Server for z/OS or WebSphere Enterprise Service Bus for z/OS is a two-step process. Keep in mind that you are updating an existing WebSphere Application Server Profile to have WebSphere Process Server or WebSphere Enterprise Service Bus function. In the first step (which is really 'Phase 2' on the previous slide), you will establish links to the WebSphere Process Server or WebSphere Enterprise Service Bus product code and update the administrative console to include new functions available with the products. During this phase, applyPTF will be run as well. This first step is really geared towards the systems programmer. There are two pieces of information that are needed to complete this step: the <WAS_HOME> directory whose profile you are updating and the directory where the WebSphere Process Server or WebSphere Enterprise Service Bus products are SMP/E installed.

The second step is geared towards the WebSphere Administrator and is called 'augmentation'. In this step, you will create the resource definitions and create the resources needed to use the WebSphere Process Server or WebSphere Enterprise Service Bus. This includes such things as databases, service integration buses and message queues. Applications are also installed into the server during this phase.

Configuration (cont.)



```
----- WebSphere Application Server for z/OS Customization -----
Option ==> _          A : BB06

Use this dialog to create WebSphere Application Server for z/OS
cells and nodes. Specify an option and press Enter.

1  Configure a security domain.
2  Create stand-alone Application Server nodes. You must complete
   Option 1 before starting this option.
3  Create Network Deployment cells and nodes. You must complete
   Option 1 before starting this option.
4  Migrate V5.x Nodes to V6 Nodes.

5  WebSphere Application Server-based add-on products. Configure
   other products that are built on WebSphere Application Server.
```

No ISPF Dialog Configuration!!

As you learned on a previous slide, the WebSphere Process Server for z/OS and WebSphere Enterprise Service Bus for z/OS products are fully contained in an HFS. You will notice that ISPF Dialogs are NOT used for configuration of the products which would require PDSes. By using only shell scripts for configuration, the installation among the various platforms is more consistent and takes advantage of common code.

Configuration - zSMPInstall.sh

Command line structure

Installing WebSphere Process Server for z/OS Definitions

Script name	Install HFS for WebSphere Process Server and Base	Procedure	Trace
{	{	{	{
zSMPInstall.sh -smproot <install HFS> -runtime <base install HFS> -install -trace			
-prereqonly			
-uninstall			

Step 1

The first step in the configuration process uses the zSMPInstall.sh script. This slide shows the syntax for the command. You need only supply values for the smproot parameter to tell the script where the WebSphere Process Server or WebSphere Enterprise Service Bus product code was SMP/E installed and the runtime parameter to tell the script what profile you plan to add WebSphere Process Server or WebSphere Enterprise Service Bus functionality to. The 'procedure' parameter will normally be install. The prereqonly parameter verifies arguments and the environment. When specifying install, the code will automatically validate the arguments and environment, or in other words, perform the function of the prereqonly parameter anyway. The install parameter will create *symbolic links* from the WebSphere Process Server or WebSphere Enterprise Service Bus read-only HFS directories to the WebSphere Application Server for z/OS /lib and /bin directories, installing the WebSphere Process Server or WebSphere Enterprise Service bus definitions. It will also enable WebSphere Process Server or WebSphere Enterprise Service bus features by running Configuration Manager scripted actions. This will create any new administrative console plug-in extensions needed.

Configuration procedure options

-prereqonly

Verify command arguments and environment.

-install

Validates prerequisites (see -prereqonly)

Creates *symlinks* from the read-only HFS directories to the **/lib** and **/bin** directories, installing the definitions.

Enables features by running Configuration Manager scripted actions, thus creating new administrative console plug-in extensions.

Creates post install file

Updates code base permissions

-uninstall

Disables features by running Configuration Manager scripted actions and removes administrative console plug-in extensions...

This slide spells out the various procedure options and talks in more detail about what each of them does when specified. You should note that the **uninstall** option will NOT completely remove the WebSphere Process Server or WebSphere Enterprise Service Bus function. Save or backup the WebSphere Application Server configuration root prior to installing WebSphere Process Server or WebSphere Enterprise Service Bus.

Configuration process - zWPSConfig.sh or zWESBConfig.sh

Command line structure

Installing WebSphere Process Server for z/OS Definitions

Script name	Procedure	Configuration response file	Property Override	Trace
zWPSConfig.sh	-augment	-response <response file path>	-Z <property name>	-trace
zWESBConfig.sh	-prereqonly			

- -Z parameter can override any values in the response file (useful for user ID/password)
- -trace '*=all=enabled'

Step 2

In the second step of the configuration process, you will use the zWPSConfig.sh or the zWESBConfig.sh script to augment the server with WebSphere Process Server or WebSphere Enterprise Service Bus function. This step requires that you provide a response file to provide information to the configuration process. Sample response files are found in the product HFS; those will be on a later slide. The **Z**-parameter is used to override any of the properties specified in the response file. An example usage of this parameter is to specify user IDs and passwords on the command line so that they are not specified in clear text in the response file. You can also specify a trace string with the **trace** parameter if needed. You would put the trace string in single quotes as shown on the slide.

Both scripts are available with the WebSphere Process Server for z/OS product. When dealing with the WebSphere Process Server for z/OS product, it is possible to configure an Application Server with the WebSphere Process Server function OR the WebSphere Enterprise Service Bus function only. Remember that WebSphere Process Server INCLUDES the WebSphere Enterprise Service Bus function. If you have the WebSphere Enterprise Service Bus for z/OS product, only the zWESBConfig.sh script is available.

Configuration procedure options

-prereqonly

Validates that zSMPInstall.sh has successfully created the product definitions for WebSphere Process Server.

-augment

Validates prerequisites
Verifies arguments
Enable profile augmentation using scripted actions.

- Mutually exclusive actions

This slide again shows the various procedure options and talks in more detail about what each of them does when specified. They are mutually exclusive and you can only specify one.

Database Creation

```
<install_root>/zWPS/V6R0/zos.config/wps_DB_StorGrp.sql
```

- Must create the databases and storage groups utilized by WebSphere Process Server/WebSphere Enterprise Service Bus for z/OS prior to running the configuration script in 'Step 2' if using DB2 AND IF dbDefineSQL=true specified in response file
- If dbDefineSQL=false specified, must just know the names that DB administrator plans to use
- Configuration scripts will create DDL and SQL for the database tables
- Sample .sql provided in above file to create databases and storage groups



As mentioned earlier, WebSphere Process Server and WebSphere Enterprise Service Bus require some databases. Those are covered on the next slide. This slide highlights the .sql file that is provided to create the DB2 databases and storage groups that you will use for the WebSphere Process Server or WebSphere Enterprise Service Bus if using DB2. Before running the zWPSConfig.sh or zWESBConfig.sh script, you will need to create the databases and storage groups if you are using DB2 AND if the dbDefineSQL parameter is set to true in the response file. The dbDefineSQL parameter determines whether the database definitions that are created during augmentation are run during the augmentation to configure the databases. If the dbDefineSQL parameter is set to 'false', you will need to run the definitions created during the augmentation manually after augmentation. Note that since it is an .sql file, it is an ASCII file.

Databases

- WPSDB – common database used by:
 - ▶ Relationship services
 - ▶ Mediation
 - ▶ Recovery
 - ▶ Business rules
 - ▶ Selector
 - ▶ Scheduler
- EVENT and EVENTCAT – used by Common Event Infrastructure
- ESBDB -- Used for logging ESB mediation events
- BPEDB – Used by Business Process Container and Human Task Manager

The next few slides describe the databases needed by the WebSphere Process Server and WebSphere Enterprise Service Bus. This shows the main databases needed with the default names used. It is important to note that you really only need to configure the databases for the functions you plan to use. The WebSphere Process Server database, or WPSDB, is the common database that is the only 'required' database. It is used by many of the functions introduced with the WebSphere Process Server and the WebSphere Enterprise Service Bus products such as Mediation, Relationship services and Business Rules. The Common Event Infrastructure, or CEI, requires two databases, EVENT and EVENTCAT, but if you do not plan on using CEI, it is not necessary to configure them. The ESBDB is needed only if you plan to log ESB mediation events. The BPEDB is one used by the Business Process Container and the Human Task Manager, so again, it is only needed if you plan to configure those components and use them in the server. Note that WebSphere Enterprise Service Bus would only use the WPSDB, EVENT and ESBDB databases.

Databases – System integration bus

- When configuring WebSphere Process Server or WebSphere Enterprise Service Bus, a default bus environment is created for you.
- Default bus environment utilizes Cloudscape as the data store for the messaging engines.
- sibDDLGenerator.sh script generates DDL so may be redefined in DB2.
 - ▶ Need to edit the DDL to either change the DBNAMEs or the STORAGE Groups. LOBS cannot be shared so they have to be unique. This means each System Integration bus must have a different DBNAME or STORAGE Group.



During the augmentation phase of the WebSphere Process Server or WebSphere Enterprise Service Bus configuration, a default service integration bus environment is created for you. This default bus environment utilizes Cloudscape as the data store for the messaging engines. You can use the sibDDLGenerator.sh script to generate .ddl so that the databases can be redefined in DB2. Note that you will need to edit the .ddl created to either change the DBNAME or the STORAGE Group names. Large Objects, or LOBS, cannot be shared between System Integration Buses so each System Integration Bus must have a different DBNAME or STORAGE Group. One option is to change the DBNAME to be equal to the SCHEMA Qualifier and then have all the databases use the same Storage group.

Databases – System Integration Bus (cont.)

- SIBAPP -- used by the SCA.APPLICATION.<cell_name>.Bus messaging engine which is used to define JMS queue destinations and other JMS resources for modules deployed with JMS bindings
- SIBSCA -- used by SCA.SYSTEM.<cell_name>.Bus which is used by the SCA runtime to support asynchronous interactions between components and modules.
- SIBCEI -- used by CommonEventInfrastructure_Bus messaging engine which is used by the CEI for asynchronous event transport
- SIBBPC – used by BPC.<cell_name>.Bus messaging engine

There are actually four service integration bus databases that might be used by the WebSphere Process Server or the WebSphere Enterprise Service Bus. The first two, SIBAPP and SIBSCA are used by the Service Component architecture so they are required. SIBCEI and SIBBPC are needed only if you plan to configure CEI and Business Process Choreographer functions.

Response files

- Needed by the zWPSConfig.sh and zWESBConfig.sh scripts

- Samples found in:

```
<install_root>/zWPS/V6R0/usr/lpp/zWPS/V6R0/zos.config
```

```
<install_root>/zWESB/V6R0/usr/lpp/zWPS/V6R0/zos.config
```

- Samples provided:

- ▶ Stand-aloneProfile.rsp – Cloudscape only
- ▶ Stand-aloneProfileDB2.rsp
- ▶ DmgrDB2.rsp
- ▶ ManagedDB2.rsp

Here are the response files that are needed by the configuration scripts. Again, the response files supply values needed for the configuration. Sample response files are found in the zos.config directory in the SMP/E install root. Four sample response files are provided. “stand-aloneProfile.rsp” is the response file used for the simplest configuration and uses Cloudscape databases. It allows for an automated configuration of a stand-alone application server. To configure a stand-alone application server with WebSphere Process Server or WebSphere Enterprise Service Bus function using DB2, you would start with the stand-aloneProfileDB2.rsp file. That configuration can also be highly automated if you are able to configure the databases during augmentation. Keep in mind that you cannot currently federate a stand-alone application server that has been augmented with WebSphere Process Server or WebSphere Enterprise Service Bus function into a network deployment cell though. When you move to the network deployment configuration, DB2 is the only option for the databases and there are two response files for that configuration: DmgrDB2.rsp and ManagedDB2.rsp. You will first use the DmgrDB2.rsp file to augment the Deployment Manager with WebSphere Process Server or WebSphere Enterprise Service Bus function. You would then augment an empty-managed node using the ManagedDB2.rsp file before federating it into the Network Deployment Cell.

Two other presentations are available that talk about the ‘simple configuration’ and the ‘network deployment configuration’ in more detail.

Response files (cont.)

- Common sections:
 - ▶ Global Properties -- used in variable substitution with a '\$' (For example, x=\$DBHOME...NOT x=/\$DBHOME/jcc/)
 - ▶ Common Properties – 'augment' keyword, cell name, server name, and so on
 - ▶ WebSphere Business Integration Core Properties – SCA Configuration, CEI (stand-alone)
 - ▶ Business Process Choreographer Properties (stand-alone)
 - ▶ WebSphere Business Integration Server Properties – AppScheduler configuration
 - ▶ Database Configuration

Each of the sample response files have many of the same common sections shown on this slide. Each starts with 'Global Properties' definitions which are used in variable substitutions throughout the rest of the response file. Using the dollar sign with these global properties, you can specify your value once and use variable substitution for the parameters in the following sections. Common properties include parameters that provide information such as cell name, server name and profile name. WebSphere Business Integration Core Properties and Business Process Choreographer properties sections are found only in the stand-alone response files since these are manual post-configuration steps in the Network Deployment Configuration. WebSphere Business Integration Server Properties provides parameters for the AppScheduler configuration if needed and finally, some database parameters are needed such as the dbDefineSQL parameter to determine if the databases will be configured during augmentation. There is documentation in each one of the response files to describe the parameters being asked for.

Post configuration

- When augmenting stand-alone server with Cloudscape or DB2, everything can be done automatically with zWPSConfig.sh and zWESBConfig.sh scripts
 - ▶ Specify dbDefineSQL=true to have .ddl scripts run during configuration
- All other configurations will need post configuration steps:
 - ▶ Run .sql/.ddl scripts to configure databases
 - ▶ Create DB2 data sources for Service Integration Bus resources
 - ▶ Configure Business Process and BPEL/HTM containers

Once you have run the zWPSConfig or the zWESBConfig shell scripts, you may still need to do some manual configuration. As mentioned earlier, the stand-alone configurations are highly automated. In the DB2 case, however, you will most likely need to have the databases configured after augmentation since that will normally be handled by a DB2 administrator. In the case of the Network Deployment configuration, you will need to actually create a WebSphere Process Server or WebSphere Enterprise Service Bus-capable server once the managed node has been federated. Depending on the functions you plan to exploit, you may also need to configure CEI, the Business Process Container and Human Task Manager after augmentation, and to configure the DB2 databases. The Network Deployment Configuration presentation goes into more detail on these manual post-configuration steps.

Summary

- WebSphere Process Server and WebSphere Enterprise Service Bus installs on top of WebSphere V6.0.2
- Uses response files and .jacl scripts to configure
 - ▶ Quickstart configuration available with CloudScape
 - ▶ More advanced configurations include DB2

In summary, this presentation has given an overview of the steps needed in order to configure an application server with WebSphere Process Server or WebSphere Enterprise Service Bus function. For more detailed information, see the Stand-alone and Network Deployment configuration presentations.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM (logo)	Cloudscape	Informix	OS/390	WebSphere
e(logo)/business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel 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 registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. 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 in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2006. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.