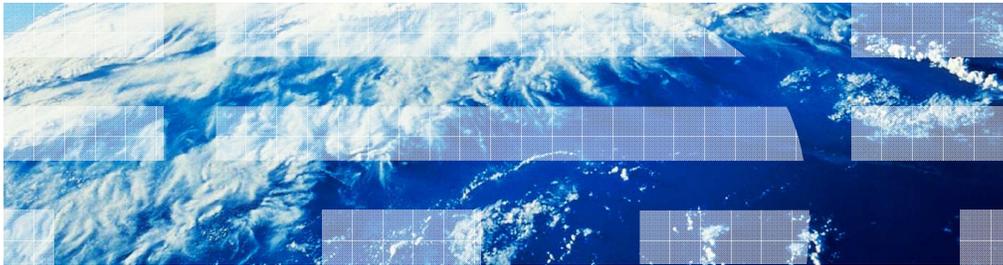


WebSphere Process Server for z/OS V7.0 WebSphere Enterprise Service Bus for z/OS V7.0

z/OS network deployment configuration



This presentation will look at the configuration of a network deployment cell to enable WebSphere® Process Server for z/OS V7 or WebSphere Enterprise Service Bus for z/OS V7 function in its servers or clusters. You should look at the **z/OS installation and configuration overview** and the **z/OS DB2 configuration** presentations as prerequisites to this one.

Goals

- Describe WebSphere Process Server for z/OS V7 and WebSphere Enterprise Service Bus for z/OS V7 configuration process using a network deployment configuration scenario

The goal of this presentation is to take you through the necessary steps to complete the configuration of WebSphere Process Server for z/OS V7 and WebSphere Enterprise Service Bus for z/OS V7 in a network deployment environment.

Configure network deployment environment

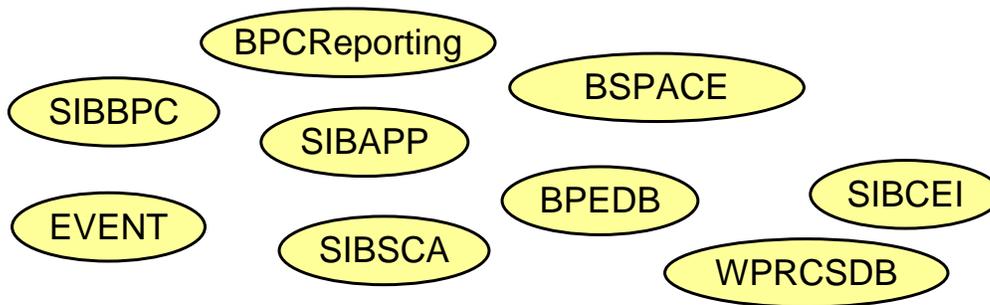
- The network deployment configuration supports a DB2® for z/OS database only
- Need to 'configure' the deployment manager node and an empty node or stand-alone node before federation
 - Do not run BBOWMNAN
 - Cloning is available once server is configured as needed

To configure WebSphere Process Server for z/OS V7 or WebSphere Enterprise Service Bus for z/OS V7 in a network deployment environment, DB2 for z/OS is a requirement. Derby is not supported in this environment. In order to configure the products in this environment, you will see that you will first 'configure' the deployment manager node and then 'configure' an empty node before federating it. This means you will create an empty node but not run the BBOWMNAN job until you have run the WebSphere Process Server or WebSphere Enterprise Service Bus configuration scripts against the empty node. You will create a server in this node as a manual process.

You are also able to configure a stand-alone profile with WebSphere Process Server or WebSphere Enterprise Service Bus and then federate that into the network deployment cell. Again, the deployment manager node needs to be configured for either WebSphere Process Server or WebSphere Enterprise Service Bus before the federation of the stand-alone profile. As you will see later, this approach has some limitations and drawbacks when it comes to resource naming.

Planning for DB2

- Review the [z/OS DB2 configuration presentation](#)
- Determine naming conventions
- Talk to your DB2 administrator
 - Run Db2DesignGenerator script to design database configuration and create SQL



4

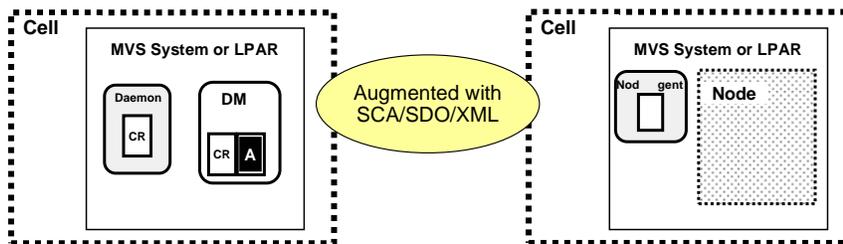
z/OS network deployment configuration

© 2010 IBM Corporation

As noted on the previous slide, the network deployment environment requires DB2. Before going any further in the configuration of WebSphere Process Server or WebSphere Enterprise Service Bus, stop to do some planning. You will soon need to know the database and schema names that will be used. Review the [z/OS DB2 configuration presentation](#) and talk to your DB2 administrator about the DB2 artifacts that are needed. You can create your database design early with your DB2 administrator's input using the Db2DesignGenerator script. The script allows you to generate the required SQL and is also used as input to the configuration process.

Configure network deployment environment

1. Configure deployment manager node for WebSphere Process Server or WebSphere Enterprise Service Bus
2. Configure empty node or stand-alone node for WebSphere Process Server or WebSphere Enterprise Service Bus
3. Federate empty node or stand-alone node
4. Perform post-configuration tasks



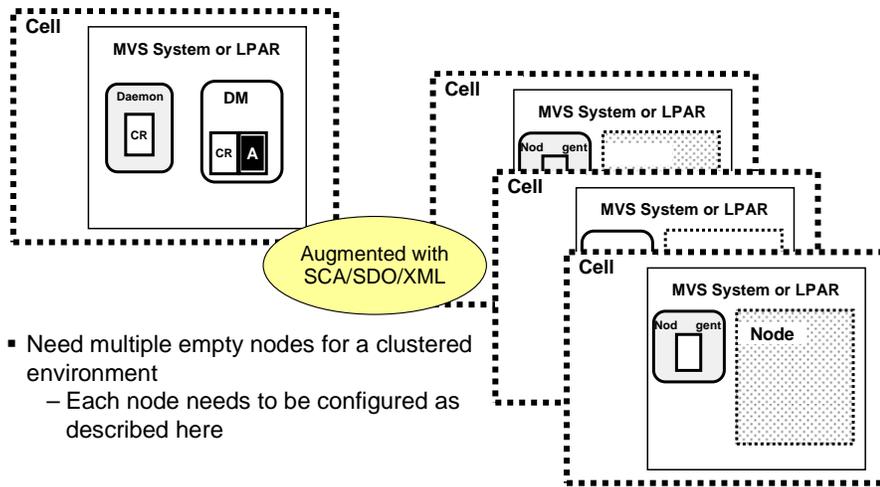
5

z/OS network deployment configuration

© 2010 IBM Corporation

The configuration of WebSphere Process Server for z/OS or WebSphere Enterprise Service Bus for z/OS in a network deployment environment can be thought of as a four step process. You will run the configuration shell scripts against the deployment manager node first. This sets the deployment manager up to be able to manage a WebSphere Process Server or Enterprise Service Bus environment. You will then configure an empty node or stand-alone node to be able to host servers that have the WebSphere Process Server or Enterprise Service Bus function. Next you will federate the empty node or stand-alone node into the network deployment cell. Finally you will perform any necessary post-configuration tasks such as configuring DB2 databases and configuring servers or clusters. Note that some of the post-configuration tasks can take place in parallel with the configuration process. This is true for the DB2 database configuration. The starting point for the empty node is shown in the graphic. You have a deployment manager cell and an empty node already configured for WebSphere and augmented with the required feature packs but you have not yet federated the empty node into the deployment manager cell.

Configure network deployment environment: clustered environment



- Need multiple empty nodes for a clustered environment
 - Each node needs to be configured as described here

In order to create a high availability clustered topology, you need to start with as many empty nodes as you have planned for your cluster. The slide shows a deployment manager that will manage a cluster of three nodes that have WebSphere Process Server or WebSphere Enterprise Service Bus function. Each of those nodes need to be configured as this presentation will describe and federated into the deployment manager cell. The steps shown in this presentation have to be repeated for each empty node.

Configure deployment manager node

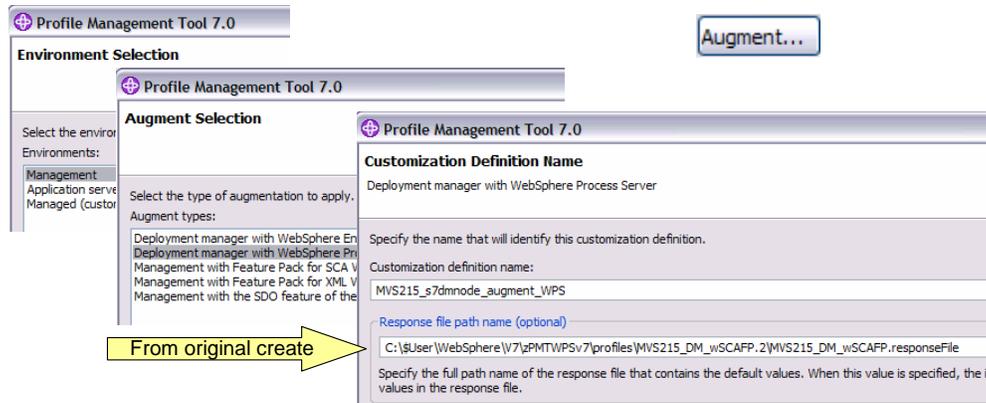
The first step you need to complete is configuring the deployment manager node with WebSphere Process Server or WebSphere Enterprise Service Bus. You will see that on the next slide.

Steps for deployment manager node configuration

- 'Augment' deployment node using the WebSphere Customization Tools
- Run zWPSInstall.sh or zWESBInstall.sh to 'install' code
- Run DbDesignGenerator tool to create a dbDesign document
- Run zWPSConfig.sh or zWESBConfig.sh to 'augment' code
- Run createDB shell script

There are a few steps involved in configuring the deployment manager node for use with WebSphere Process Server or WebSphere Enterprise Service Bus. You see these outlined here. The first thing you need to do is 'augment' your deployment manager node using the WebSphere Customization Tools. This creates a job that allows you to create a symbolic link to the product code and a response file for augmentation later. Once that is done, you need to 'install' the WebSphere Process Server or WebSphere Enterprise Service Bus product code into your deployment manager node configuration. The next step is running the DBDesignGenerator tool to create the dbDesign document that is needed in the 'augment' step. The 'augment' step is accomplished by running the zWPSConfig or zWESBConfig script. Finally, you will run the createDB shell script to finish up SQL generation. You will look at each of these steps on the next slides.

WebSphere customization tools: Augment deployment manager node



- Augment deployment manager node with WebSphere Process Server or WebSphere Enterprise Service Bus in the WCT
 - Specify the response file for the deployment manager node you created

As mentioned earlier, this presentation assumes that you have already augmented your nodes with the required feature packs (SCA, XML and SDO). The first step then in WebSphere Process Server for z/OS or WebSphere Enterprise Service Bus for z/OS configuration is to then augment the deployment manager node with WebSphere Process Server or WebSphere Enterprise Service Bus. To do this, start with the WebSphere customization tool (WCT) and select to 'Augment' a 'Management' environment. You'll be given the option to augment with either WebSphere Process Server or WebSphere Enterprise Service Bus as seen in the middle box on the slide. To save you some typing and the possibility for typos, input the response file from your original deployment manager creation.

WCT screens

Profile Management Tool 7.0
Deployment manager with WebSphere Process Server

Base File Systems
Deployment manager with WebSphere Process Server

Configuration file system
Mount point:
/etc/wasv7config/s7cell/s7dmnode
Directory path name relative to mount point:
DeploymentManager

WebSphere Application Server product file system
Product file system directory (or path name of intermediate symbolic link):
/etc/wasv7config/s7cell/s7dmnode_wassmpel

Profile Management Tool 7.0
Deployment manager with WebSphere Process Server

Database Design
Deployment manager with WebSphere Process Server

You can use a design file that is generated from the database design tool (DDT) to configure the database.

Use a database design file for database configuration
Name (fully qualified) of the database design file:
/u/wuser/wpswork/wps.nd.topology.dbDesign

Delay execution of database scripts (must select if using a remote database).

- Update to specify intermediate symbolic link, is using
- Specify a file name for database design

As you go through the augment in the WCT, most information is filled out from your response file if you specified one. To see all the screens, you can look at the **z/OS simple configuration** presentation. Here you see two that need information from you. In the first one, you will need to change the base file system path name to its intermediate symbolic link if you are using one. It will default to the absolute path from the response file. The second screen shows the 'Database Design' input. Since DB2 is a requirement in this environment, you need to specify a database design. You have not actually created the design yet, so take note of the name you fill in here. You will need this later when you run the DbDesignGenerator tool. You should also check the box to delay execution of the database scripts.

Process deployment manager configuration

Process...

- Upload generated jobs and data
 - hlq.CNTL(BPZDOLNK)
 - hlq.DATA(BPZRSPM)

```

mvs215 - [32 x 80]
File Edit View Communication Actions Window Help
Menu Utilities Compilers Help
BROWSE      WSUSER.WASV70.S7CELL.WPSV7.CNTL(BPZDOLNK)  Line 00000000 Col 001 080
Command ==> sub_                                     Scroll ==> CSR_
***** Top of Data *****
//BPZDOLNK JOB (ACCTNO,ROOM), 'USERID',CLASS=A,REGION=0M,      00000000
//              MSGCLASS=A,NOTIFY=&SYSUID                    00000000
//*                                                         00000000
  
```

- Run BPZDOLNK job, is using symbolic links

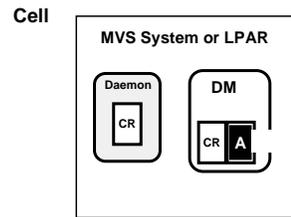
Once you have filled out the screens in the WCT, specify 'Process' in the WCT to upload the data to your z/OS system. There are two members you are interested in, BPZDOLNK and BPZRSPM. You will use BPZDOLNK right away. BPZRSPM is a response file and is used later in the process. The first thing you want to do is run the BPZDOLNK job. This job will create an intermediate symbolic link within your existing deployment manager node profile to the WebSphere Process Server for z/OS or WebSphere Enterprise Service Bus for z/OS product code.

Configure deployment manager node – 'install'

- Run zWPSInstall.sh or zWESBInstall.sh
 - 'runtime' should point to deployment manager configuration:
 - /WebSphere/V7R0/DeploymentManager
 - Must already be augmented with SCA/XML/SDO feature packs
 - Run using WebSphere Application Server administrator user id

```
<wps_smpe_root>/zos.config/bin/zSMPInstall.sh
-smproot <wps_smpe_root>
-runtime <deployment_manager_root>
-install
```

Back up file system first!!!!



The next step in the process is running the zSMPInstall shell script, pointing the runtime to the deployment manager configuration HFS as shown on the slide. Note that the deployment manager node must already be augmented with the SCA, XML and SDO feature packs before running the installation job. This is a task for the system administrator, since it is somewhat of an extension of the SMP/E installation. You should use a WebSphere Administrator user ID to run the script.

zWPSInstall example

- zWPSInstall.sh example JCL
 - Will set up symbolic links to the product code
 - Update the administrative console

```
//INSTV70 JOB (ACCTNO,ROOM), 'HONKEN',CLASS=A,REGION=0M,
// NOTIFY=HONKEN,TIME=NOLIMIT
// *
// *
// *****
// * WebSphere Process Server -installonly *
// *****
//INSTO EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
export WPS_DEBUG=true; +
/etc/WAS60C/usr/lpp/zWPS/V7R0/zos.config+
/bin/zWPSInstall.sh +
  -smproot /etc/wasv7config/s7cell/s7dmnode_wpssmpe +
  -runtime /etc/wasv7config/s7cell/s7dmnode/DeploymentManager +
  -install +
1> /tmp/installonly_84821.out +
2> /tmp/installonly_84821.err
/*
```

...or run from
USS environment

This slide shows an example of running the job from JCL. It can also be run directly from the USS environment. This job will create symlinks in your WebSphere Application Server deployment manager configuration to the WebSphere Process Server or WebSphere Enterprise Service Bus product code. It will also add plug-ins to the administrative console for new functions needed for the WebSphere Process Server or WebSphere Enterprise Service Bus.

DbDesignGenerator

- Run DbDesignGenerator tool to create a dbDesign document and generate SQL
 - Found in: <WAS_HOME>/util/dbUtils

```
(10)[save and exit]
```

```
Please enter the number for the database component :10
```

```
[status] wps.nd.topology is complete with 0 remaining items:
```

```
Please enter the output directory
```

```
[default=/etc/wasv7config/s7cell/s7dmnode/DeploymentManager/util/dbUtils]
```

```
:/u/wsuser/wpswork/
```

```
Please enter the output file name [default=wps.nd.topology.dbDesign] :
```

```
[info] The database design has been generated in  
/u/wsuser/wpswork/wps.nd.topology.dbDesign
```

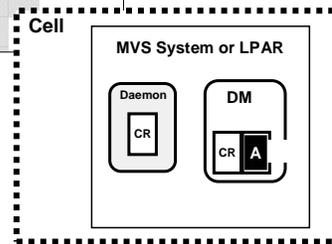
```
generate database scripts? (y/n) [default=y] :y
```

The next step in the process involves running the DbDesignGenerator tool to create the dbDesign document for augmentation and to generate the required SQL. For more details on the tool, see the **z/OS DB2 configuration** presentation. This slide shows the generation of the dbDesign document and the prompt for generation of the SQL. Remember that the database design location and name must match what you specified in the WCT earlier. If they do not, you will need to regenerate your profile definition in the WCT and upload the information to your z/OS environment again.

Configure deployment manager node – ‘augment’

- Run zWPSConfig.sh or zWESBConfig.sh
 - ‘response’ should point to the response file created by WCT
 - Run using WebSphere Application Server administrator user ID

```
cp "'HLQ.DATA(BPZRSPM)'" /u/wsuser/DmgrDB2.rsp
<configuration_root>/DeploymentManager/bin/zWPSConfig.sh
  -response /u/wsuser/DmgrDB2.rsp
  -augment
```



The next step in the process is augmentation. You will accomplish this by running the zWPSConfig or zWESBConfig shell scripts. The only necessary parameter, other than ‘augment’, is the file name of the response file. The response file was created for you when running the augment in the WebSphere Customization Tools. You uploaded it to the target HLQ.DATA dataset as BPZRSPM. In order to specify it here on the zWPSConfig command, you need to first copy it over to the HFS. One way to do that is shown in the gray box. Another copy alternative is to use OPUT. You’ll see that in the JCL on the next slide. After the augment script has been run, various resources are configured and needed applications installed. It must be run from the WebSphere Application Server administrator user ID.

zWPSConfig example

- zWPSConfig.sh example JCL
 - Creates resources and install applications needed to run the WebSphere Process Server or WebSphere Enterprise Service Bus Installs application

```

//S7DMAUG JOB (ACCTNO,ROOM) , 'HONKEN' ,CLASS=A,REGION=0M,
// NOTIFY=&SYSUID,TIME=NOLIMIT
//*****
//* Run zWPSConfig.sh *
//*****
//COPY EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
oput 'WSUSER.WASV70.S7CELL.WPSV7.DATA(BPZRSPM)' +
    '/u/wsuser/DmgrDB2.rsp'
/
//AUGMT EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
BPXBATCH SH +
cd /etc/wasv7config/s7cell/s7dmnode/DeploymentManager/bin; +
./zWPSConfig.sh +
-response /u/wsuser/DmgrDB2.rsp +
-augment +
1> /tmp/zWPSConfigDM_40135.out +
2> /tmp/zWPSConfigDM_40135.err

```

...or run from
USS environment

This slide shows an example of running the job from JCL. Again it can also be run directly from the USS environment. This also shows an example of copying the response file from the DATA dataset using the oput command. The augment job, or zWPSConfig, will create needed resources in the WebSphere Application Server environment and install needed applications.

createDB

- Run createDB shell script to generate SQL for CEI and concatenate SQL generated by DbDesignGenerator
 - Found in: `-prefix-/usr/lpp/zWPS/V7R0/zos.config/samples`
 - Copy to a work directory and update
 - `./createDB.sh -RunSQL`
 - Results found in `cdbtmp/output.out` and `cdbtmp/error.out` if SQL run
- Concatenated SQL scripts found here:
 - `<work directory>/cdbtmp`
 - `Bspace.sql`
 - `bpc.sql`
 - `bpcr.sql`
 - `ceidb.sql`
 - `ceidbx.sql`
 - `common.sql`
 - `crdb.sql`
 - `sibAPP.sql`
 - `sibBPC.sql`
 - `sibCEI.sql`
 - `sibSCA.sql`



You will also need to run the createDB shell script. This script will do a couple of things for you. First of all, it will create tailored CEI SQL. It will also concatenate the SQL created by DbDesignGenerator into component-related scripts. createDB can also run the SQL for you if you have the required authority. In order to run the createDB shell script, you need to copy it from the `zos.config/samples` directory and update it for your installation. You can also update the 'defaults' that are chosen. For instance, the script is setup to run the SQL, by default, with the `DBRunSQL=true` parameter. If you do not have the authority to run the SQL, you can change that default in the script or you can specify minus `RunSQL` as shown on the slide. If you do run the SQL using createDB, the results are found in the `cdbtmp` directory as shown. The concatenated SQL scripts are also found in the `cdbtmp` directory. These files can be given to your DB2 administrator or run through the `Ddl2Pds` script for transfer to the z/OS environment. Again, more details on using the script are found in the [z/OS DB2 configuration](#) presentation.

Configure empty node or stand-alone node

After having completed the deployment manager node configuration, move your attention to the empty node, or nodes, that you have configured. You need to do the same processing to the empty or stand-alone nodes.

Steps for empty node or stand-alone node configuration

- 'Augment' empty node or stand-alone node using the WebSphere Customization Tools
- Run zWPSInstall.sh or zWESBInstall.sh to 'install' code
- Run zWPSConfig.sh or zWESBConfig.sh to 'augment' code

While the empty or stand-alone node configuration is less complicated, there are still a few steps involved in the configuration. You see these outlined here. Like the deployment manager node, the first thing you need to do is 'augment' your empty node or stand-alone node using the WebSphere Customization Tools. Again, this creates a job that allows you to create a symbolic link to the product code and a response file for augmentation later. Once that is done, you need to 'install' the WebSphere Process Server or WebSphere Enterprise Service Bus product code into your empty node or stand-alone node configuration. Finally, you need to 'augment' the node which is accomplished by running zWPSConfig or zWESBConfig. You will look at each of these steps on the next slides.

WebSphere Customization Tools: Augment empty node'

Profile Management Tool 7.0

Environment Selection

Profile Management Tool 7.0

Augment Selection

Select the environment

Environments:

Management
Application server
Managed (custom) node

Select the type of augment

Augment types:

Managed (custom) node
Managed (custom) node
Managed (custom) node
Managed (custom) node
Managed (custom) node

Augment...

Profile Management Tool 7.0

Customization Definition Name

Managed (custom) node with WebSphere Process Server

Specify the name that will identify this customization definition.

Customization definition name:

MVS215_ManagedNode_Augmen_tWPS

Response file path name (optional)

C:\\$User\WebSphere\7\zPMTWPSv7\profiles\MVS215_ManagedNode_wSCAFP.6\MVS215_ManagedNode_wSCAFP.responseFile

Specify the full path name of the response file that contains the default values. When this value is specified, the input fields in the tool values in the response file.

From original create

- Augment empty managed node (or stand-alone node) with WebSphere Process Server or WebSphere Enterprise Service Bus in the WCT
 - Specify the response file for the empty node you created

You will again start with augment in the WCT. These slides show processing for an empty node but the same basic processing is true for a stand-alone as well. You'll notice that you should select 'Managed (custom) node' in the selections for environment and augment here. Again, point it to the response file from the original empty node you created to prevent typos in the fields that can be primed from your base profile.

Process empty node configuration

Process...

- Upload generated jobs and data
 - hlq.CNTL(BPZDOLNK)
 - hlq.DATA(BPZRSPN)

```

mvs215 - [32 x 80]
File Edit View Communication Actions Window Help
Menu Utilities Compilers Help
BROWSE      WSUSER.WASV70.S7CELL.S7NODEA.WPSV7.CNTL(BP Line 00000000 Col 001 080
Command ==> sub_                               Scroll ==> CSR
***** Top of Data *****
//BPZDOLNK JOB (ACCTNO,ROOM),'USERID',CLASS=A,REGION=0M, 00000000
//          MSGCLASS=A,NOTIFY=&SYSUID 00000000
  
```

- Run BPZDOLNK job, is using symbolic links

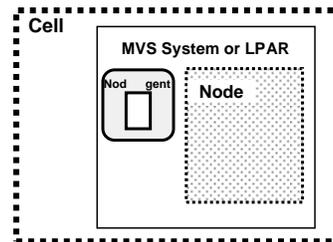
As you did with the deployment manager node, you need to 'Process' the definition in the WCT to upload the data to your z/OS system. The same files of interest are uploaded except the response file is now called BPZRSPN. Run the BPZDOLNK job now to create the intermediate symbolic link to the WebSphere Process Server for z/OS or WebSphere Enterprise Service Bus for z/OS product code.

Configure empty or stand-alone node – 'install'

- Run zWPSInstall.sh or zWESBInstall.sh
 - 'runtime' should point to empty or stand-alone node configuration:
/WebSphere/V7R0/AppServer
 - Must already be augmented with SCA/XML/SDO feature packs
 - Run using WebSphere Application Server administrator user id

```
<wps_smpe_root>/zos.config/bin/zSMPInstall.sh
-smproot <wps_smpe_root>
-runtime <app_server_root>
-install
```

Back up file
system first!!!!

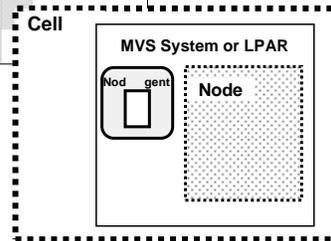


Now you are ready to run the installation for the empty node. Remember, you should not have run the BBOWMNAN job yet to federate it. You will run the zWPSInstall or zWESBInstall shell script again, this time specifying the configuration HFS for the empty node or stand-alone node that was created. This will again set up the symlinks to the product code from the configuration HFS.

Configure empty node – ‘augment’

- Run zWPSConfig.sh or zWESBConfig.sh
 - ‘response’ should point to the response file created by WCT
 - Run using WebSphere Application Server administrator user id

```
cp "//HLQ.DATA(BPZRSPN)" /u/wsuser/ManagedDB2.rsp
<configuration_root>/AppServer/bin/zWPSConfig.sh
  -response /u/wsuser/ManagedDB2.rsp
  -augment
```



Finally, you will run the augment for the empty node. This is accomplished by running the zWPSConfig or zWESBConfig shell script. The response parameter should again point to the response file created for you by the WCT and uploaded to the BPZRSPN member of the DATA dataset. You again need to copy the response file to the HFS first.

Federate empty node or stand-alone node

Once the nodes are configured, you are ready to federate the nodes into the deployment manager cell.

Federate augmented node or nodes

- Run SQL to create database objects, if not done already
- Start deployment manager
- Run the BBOWMNAN job to federate the empty node (or stand-alone node)

```
addNode.sh <DM_HOST> <DM_PORT> -includeapps -includebuses  
-nodegroupname DefaultNodeGroup -username xxADMIN  
-password xxx -nodeagentshortname xxAGNT1 -replacelog
```

 - includebuses parameter required for stand-alone node
 - Configuration is now ready for WebSphere Process Server or WebSphere Enterprise Service Bus function
 - Note: stand-alone node is already configured with WebSphere Process Server/WebSphere Enterprise Service Bus function

Once both the deployment manager node and the empty or stand-alone node are configured, you are ready to federate the node or nodes into the network deployment cell. Since you will be starting the deployment manager in order to federate the nodes, you should ensure that the database objects have been created for WebSphere Process Server or WebSphere Enterprise Service Bus. Without having run the SQL to create the database objects, you will receive errors on starting the deployment manager. In order to federate the nodes, you can run the BBOWMNAN job that was created during node creation. Also shown on the slide is the addNode shell script invocation that will do the same thing. The includebuses parameter is required on the addNode command if federating a stand-alone node to include the systems integration buses that were created for you.

If you federate an empty node, there are still no servers defined where you can run a workload that uses the new WebSphere Process Server or WebSphere Enterprise Service Bus functions. In the case of the stand-alone node, there is a server defined and some of the additional function that needs a server might already be configured. For instance, it is possible that the process choreography and human task function may already exist and the common event infrastructure environment also may exist.

Federation of augmented stand-alone node (1 of 3)

- Start with an augmented empty deployment manager node
 - Only initial federation supported
 - After the first federated node with applications and buses, only empty nodes can be federated
 - Can then promote WebSphere Process Server/WebSphere Enterprise Service Bus Server to a cluster

As mentioned earlier, there are limitations when it comes to federating a stand-alone node that has been augmented with WebSphere Process Server or WebSphere Enterprise Service Bus. First of all, only initial federation is supported. This means that it is only supported if there are currently no other nodes in the network deployment cell. Then after the federation of the first stand-alone node, all other WebSphere Process Server or WebSphere Enterprise Service Bus-capable nodes that are federated must be empty nodes. Once federated though, you are able to promote the configured server to a cluster.

Federation of augmented stand-alone node (2 of 3)

- Naming of resources based on stand-alone node cellname
 - Bus names
 - BPC.xibase2.Bus
 - CEI_xibase2.BUS
 - SCA.APPLICATION.xibase2.Bus
 - SCA.SYSTEM.xibase2.Bus
 - Messaging engines
 - xxnode2.xxsr012-BPC.xibase2.Bus
 - xxnode2.xxsr012-CEI_BUS
 - xxnode2.xxsr012-SCA.APPLICATION.xibase2.Bus
 - xxnode2.xxsr012-SCA.SYSTEM.xibase2.Bus



Drawbacks were also mentioned when it comes to resource naming. After federation into the network deployment cell, you might notice that the names of the resources are still based on the original stand-alone node's cell name. This can be confusing in your configuration. On the slide here, the original cell name was xibase2. If resources were originally created in the deployment manager node instead, they would have the correct xxcell name.

Federation of augmented stand-alone node (3 of 3)

- Eventually going to cluster
 - Applications:
 - BPEContainer_xxnode2_xxsr012
 - BPCEXplorer_xxnode2_xxsr012
 - TaskContainer_xxnode2_xxsr012
 - BSpaceEAR_xxnode2_xxsr012
 - BusinessRules_xxnode2_xxsr012

Choose not to configure WebSphere Process Server components

Configure a sample Business Process Choreographer

Configure Business Space

Configure the Business Rules Manager

- Empty node better choice

The application names are also based on the stand-alone configuration. If you will eventually be configuring a cluster to run WebSphere Process Server or WebSphere Enterprise Service Bus applications, you can choose not to configure the various WebSphere Process Server components during augmentation of the stand-alone node. If you wait until you have created the cluster, the names will better reflect your configuration. The same goes for the SCA configuration. You probably do not want to inherit the stand-alone node's SCA configuration, but in that case, you have no choice but to configure it in the stand-alone node first. The empty node gives you better control over your configuration and is the recommended alternative.

Perform post-configuration tasks

Now that you have a deployment manager cell and an empty node federated, you are ready to configure the various components that are part of WebSphere Process Server or WebSphere Enterprise Service Bus. You have a couple of options to do this.

Post-configuration, option one (1 of 2)

- 'Manual' configuration
 - Create cluster/server manually using ProcessServer template

Create a new cluster

Create a new cluster

→ Step 1: Enter basic cluster information

Step 2: Create first cluster member

Step 3: Create additional cluster members

Step 4: Summary

Enter basic cluster information

* Cluster name
s7sr01

Short name
S7SR01

Prefer local. Specifies whether enterprise bean requests will be routed to the node on which the client resides when possible.

Configure HTTP session memory-to-memory replication

Next Cancel

Select basis for first cluster member:

Create the member using an application server template.

defaultProcessServerZOS

The first option is more manual than the second. It involves creating a cluster or server manually using the ProcessServer template as shown on the slide.

Post-configuration, option one (2 of 2)

– Configure components individually

Best option
pre-V7

[WebSphere application server clusters](#) > **s7sr01**

Use this page to change the configuration settings for a cluster. A server cluster consists of a group of application servers. If one of the application servers that is a member of the cluster fails, requests are routed to other members of the cluster.

Configuration **Local Topology**

| General Properties | Cluster messaging |
|--|---|
| * Cluster name s7sr01 | ■ Messaging engines |
| * Short name S7SR01 | Business Integration |
| Unique ID CS3C616F6885623000001A00000000100000000 | ■ Business Integration Configuration |
| Bounding node group name DefaultNodeGroup | ■ Business Space Configuration |
| <input checked="" type="checkbox"/> Prefer local | ■ REST services |
| <input type="checkbox"/> Enable failover of transaction log recovery | ■ Service Component Architecture |
| | <input type="checkbox"/> Common Event Infrastructure |
| | <input type="checkbox"/> Business Process Choreographer |
| | <input type="checkbox"/> Business Rules |
| | <input type="checkbox"/> Service Monitor |

Once you have a server or cluster created, you can then configure each of the needed components such as SCA, CEI and business process choreographer. As you click each of the highlighted links, you are prompted for information needed to configure that component on the server or cluster. This was the best option before version 7.

Post-configuration, option 2

- Deployment environment configuration
 - Configure 'Deployment Environment'



Now in version 7, you have a more integrated option available to you. In version 7, the pattern based deployment environment wizard is a second option. You'll see that option shown here. You'll find the 'Deployment Environments' option under 'Servers' as seen on the slide. You want to create a new one.

Configure deployment environment

Create a deployment environment that is based on a pattern or a custom deployment environment, or load an external deployment environment definition.

Enter the deployment environment name and select the path to take through the wizard (Fast path or Detailed). If you choose "Fast path: Show only needed steps", the wizard displays only those pages that do not have assigned default values.

At the end of the wizard, you can generate the deployment environment by clicking "Finish and Generate Environment", or you can click "Finish" to save the deployment environment definition. To generate the environment, you must properly configure all of the required parameters.

Create Deployment Environment

- Create a deployment environment based on a pattern
- Create a custom deployment environment
- Create a deployment environment based on an imported design

File path: Browse

* Deployment environment name: s7sr01

Fast path: Show only needed

Detailed: Show all steps

Next Cancel

Select the feature for the deployment environment.

| Select | Features | Description |
|----------------------------------|----------|----------------------------------|
| <input checked="" type="radio"/> | WPS | WebSphere Process Server |
| <input type="radio"/> | WESB | WebSphere Enterprise Service Bus |

Previous Next Cancel

<http://www.redbooks.ibm.com/redpieces/abstracts/SG247831.html>

33 z/OS network deployment configuration © 2010 IBM Corporation

Here you are given a few options. This presentation will focus on creating a deployment environment based on a pattern. The URL shown on the slide points to an IBM Redbooks® publication that provides a spreadsheet to use with the imported design option. You will find directions on how to use that option in there. Once you have selected that you want to create a deployment environment based on a pattern, you are given the option of configuring WebSphere Process Server or WebSphere Enterprise Service Bus in the environment.

Configure deployment environment: Patterns

Create new deployment environment

Deployment environment patterns are rules-based configurations of the most commonly used business integration topologies. A pattern provides a template for an environment configuration. Deployment environment patterns represent well-known, tested topologies with component configurations that work together; using patterns ensures reliable deployment environment functionality. You can create custom deployment environments if you require a configuration other than those supplied by the patterns.

Select a pattern that provides the topological characteristics of the deployment environment:

| Select | Deployment Environment Patterns | Description |
|----------------------------------|-------------------------------------|---|
| <input checked="" type="radio"/> | Single Cluster | The single cluster pattern is the simplest pattern. It defines one application deployment target cluster, which includes the messaging infrastructure, the Common Event Infrastructure (CEI), and supporting applications. |
| <input type="radio"/> | Remote Messaging | The remote messaging pattern defines one cluster for application deployment and one remote cluster for the messaging infrastructure. The Common Event Infrastructure (CEI) and other supporting applications are configured on the application deployment target cluster. |
| <input type="radio"/> | Remote Messaging and Remote Support | The remote messaging and remote support pattern defines one cluster for application deployment, one remote cluster for the messaging infrastructure, and one remote cluster for the Common Event Infrastructure (CEI) and other supporting applications. |

Previous Next Cancel

Suggested pattern for z/OS

Three different patterns are available. These are the most common patterns used by customers. The single cluster topology is the recommended pattern for z/OS. The two remote messaging patterns serve to split out the messaging function and possibly CEI into their own Java™ Virtual Machine (JVM) . Because z/OS's unique architecture already provides multiple JVMs within a servant and a separate one for messaging in the form of the adjunct, there often isn't the need for a further split. By providing an adjunct address space with a separate JVM for the message engines, the architecture of a single-cluster in WebSphere Application Server for z/OS is analogous to the remote messaging topology seen on the slide. For more information on the various topologies, you can reference the SG247831 Redbooks publication titled **WebSphere Business Process Management V7 Production Topologies for System z®**.

Configure deployment environment: Nodes/clusters

The screenshot shows two overlapping windows from the 'Create new deployment environment' wizard.

Top Window: Step 1: Select Nodes

Select Nodes

Select the nodes to use for the development environment. The *Single Cluster* deployment environment *s7sr01* requires at least **1 node**. For high-availability and failover environments, select two nodes. For scalability, select more than two nodes.

| Select | Node | Version | Host |
|-------------------------------------|---------|-------------|----------------------|
| <input checked="" type="checkbox"/> | s7nodea | WPS 7.0.0.0 | myr215.devel.ibm.com |

Bottom Window: Step 2: Clusters

Map the clusters to the listed nodes by indicating the number of cluster members to configure.

| Node | Version | Application Deployment Target |
|---------|-------------|-------------------------------|
| s7nodea | WPS 7.0.0.1 | 1 |

A yellow callout bubble points to the '1' in the 'Application Deployment Target' column of the 'Clusters' table, containing the text: "Create one cluster member in node s7nodea".

35

z/OS network deployment configuration

© 2010 IBM Corporation

Continuing on the 'Single Cluster' path, you are prompted for the nodes that you want to include in your WebSphere Process Server or WebSphere Enterprise Service Bus environment. This shows only one node but you can have several. On the next screen, you need to specify how many cluster members, or servers, you want created in each node. Again, since z/OS has an architecture that automatically allows more than one server per servant, one should suffice.

Configure deployment environment: REST/database configuration

The screenshot shows two overlapping windows from the 'Create new deployment environment' wizard. The top window is titled 'System REST Service Endpoints' and contains a sidebar with steps 1 through 9. Step 3, 'System REST Service Endpoints', is selected. The main content area explains that this page is for configuring REST service endpoints and includes fields for Protocol (http://), Host name or virtual host in a load-balanced environment (mvs215.rtp.raleigh.ibm.com), and Port (19547). The bottom window is titled 'Import database configuration' and also has a sidebar with steps 1 through 5. Step 4, 'Import database configuration', is selected. The main content area explains that this page is for importing a database design document. A field labeled 'Import database configuration file for features:' contains the file path 'C:\temp\wps.nd.topology.dbDesign', which is highlighted with a red box. A yellow callout bubble points to this field with the text: 'Need to bring the dbDesign file to the PC – use same one specified during augmentation'. The bottom window also has 'Browse...' and 'Clear' buttons.

The next screen will ask for some information to configure the REST Services application which is used by the Business Space component. The second screen shown asks for you database configuration. You need to supply the dbDesign file that you created with DbDesignGenerator here. Note that it needs to be downloaded to your PC.

Configure deployment environment: Database

Database

Edit the database parameters for the data sources that are needed by this deployment.

| Select | Component | Database Name | Schema | Create Tables | User Name | Password | |
|--------------------------|---|---------------|--------|--------------------------|-----------|----------|----------------------------|
| <input type="checkbox"/> | Business Process Choreographer | MVS215D1 | S7CELL | <input type="checkbox"/> | DB2D | ***** | mvs215.rtg DB2 for z/OS v8 |
| <input type="checkbox"/> | Business Process Choreographer | MVS215D1 | S7S1B | <input type="checkbox"/> | DB2D | ***** | mvs215.rtg DB2 for z/OS v9 |
| <input type="checkbox"/> | Business Process Choreographer reporting function | MVS215D1 | S7CELL | <input type="checkbox"/> | DB2D | ***** | mvs215.rtg DB2 for z/OS v8 |
| <input type="checkbox"/> | Business Space | MVS215D1 | S7CELL | <input type="checkbox"/> | DB2D | ***** | mvs215.rtg DB2 for z/OS v8 |
| <input type="checkbox"/> | Common Event Infrastructure | MVS215D1 | S7CELL | <input type="checkbox"/> | DB2D | ***** | mvs215.rtg DB2 for z/OS v8 |
| <input type="checkbox"/> | Common Event Infrastructure | MVS215D1 | S7S1C | <input type="checkbox"/> | DB2D | ***** | mvs215.rtg DB2 for z/OS v9 |
| <input type="checkbox"/> | Service Component Architecture | MVS215D1 | S7S1S | <input type="checkbox"/> | DB2D | ***** | mvs215.rtg DB2 for z/OS v9 |
| <input type="checkbox"/> | Service Component Architecture | MVS215D1 | S7S1A | <input type="checkbox"/> | DB2D | ***** | mvs215.rtg DB2 for z/OS v9 |

Values should be taken from dbDesign file (some need updates)

The next screen summarizes the database configuration that the wizard will create. These fields are taken from your dbDesign document and while most of them are correct, there are a few that might need to be updated. Check them before continuing and update as needed!

Configure deployment environment: Security

The screenshot shows the 'Create new deployment environment' wizard. The left sidebar lists steps from Step 1 to Step 9, with Step 7 'Business Process Choreographer' highlighted. The main content area is titled 'Security' and contains the following information:

Security
 Edit the user names and passwords for the authentication aliases that are needed by this deployment environment.

| Component | User name | Password | Confirm Password | Description |
|--------------------------------|-----------|----------|------------------|-------------|
| Common Infrastructure | | | | |
| Service Core Architecture | | | | |
| Business Process Choreographer | | | | |

Below this table, there is a section for 'Business Process Choreographer' with a 'Security' subsection. It contains a table with the following data:

| Role | Users | Groups | Description |
|---------------|---------|--------|--|
| Administrator | s7admin | S7CFG | User names, group names, or both, separated by the " " symbol, for the business flow and human task administrator role. Users who are assigned to this role have all privileges. |
| Monitor | s7admin | S7CFG | User names, group names, or both, separated by the " " symbol, for the business flow and human task monitor role. Users who are assigned to this role can view the properties of all of the business process and task objects. |

At the bottom of the wizard, there are 'Previous', 'Next', and 'Cancel' buttons.

38

z/OS network deployment configuration

© 2010 IBM Corporation

The next two panels deal with security. There are authentication aliases required for the various components and Business Process Choreographer needs information for various security roles in addition to additional authentication roles and information for the Human Task Manager Mail Session. This screen is only partially shown.

Configure deployment environment: Context roots

The screenshot shows a wizard window titled "Create new deployment environment". On the left is a vertical list of steps: Step 1: Select Nodes, Step 2: Clusters, Step 3: System REST Service Endpoints, Step 4: Import database configuration, Step 5: Database, Step 6: Security, Step 7: Business Process Choreographer, and Step 8: Web Application Context Roots. Step 8 is highlighted with a yellow arrow. The main content area is titled "Web Application Context Roots" and contains the instruction "Modify the context roots for Web applications." Below this is a section labeled "Context Root" with three input fields: "Business Process Choreographer Explorer context root:" with the value "/bpc", "Business Rules Manager context root:" with the value "/br", and "Business Space context root:" with the value "/BusinessSpace".

The last bit of information needed before allowing the wizard to create your environment are some context roots. You can change the context roots for the business process choreographer or business rules application, if required. The defaults are shown.

Configure deployment environment: Generate

Step 1: Select Nodes
Step 2: Clusters
Step 3: System REST Service Endpoints
Step 4: Import database configuration
Step 5: Database
Step 6: Security
Step 7: Business Process Choreographer
Step 8: Web Application Context Roots
Step 9: Summary

Summary

This summary shows an overview of your new deployment environment. To save the deployment environment definition, click on "Finish". To save the definition and generate the deployment environment, click on "Finish and Generate Environment".

Overview

| Parameter | Value |
|--------------------------------|----------------|
| Deployment Environment Pattern | Single Cluster |
| Deployment environment name | s7sr01 |
| Features | WPS |
| Deployment Environment Status | Incomplete |

Deployment Targets

| Cluster | Nodes |
|-------------------------------|---------|
| Application Deployment Target | s7nodea |

Data Sources

| Component | Database Name | Schema | Database Provider | Database Host |
|-------------------------------------|---------------|--------|-------------------|---------------|
| Business Rules Manager | | | | |
| Parameter | | | | |
| Business Rules Manager context root | | | | |
| /br | | | | |

Buttons: Previous, Finish, **Finish and Generate Environment**, Cancel

Creates cluster and server with configured components

Once you have all the information as you require, you are given a summary of the information you have inputted. Once you have verified that it is setup as you intended, there is a 'finish and generate' button found at the bottom of the screen. Once you click that, you will see the progress as it deploys your environment. This will take a few minutes. When it is complete, you will have a cluster containing a server, or servers, with the various WebSphere Process Server components configured.

Configure deployment environment: Updates

The screenshot displays the configuration interface for WebSphere application server clusters. It shows the 'WebSphere application server clusters' page for 's7sr01.AppTarget' and the 'Application servers' page for 's7sr01.AppTarget.s7nodea.0'. The 'Ports' section is expanded, showing a table of resources with their respective hostnames and ports.

All defaults taken...may want to change

| Select | Port Name | Host | Port |
|--------------------------|------------------------|----------------------------|------|
| <input type="checkbox"/> | BOOTSTRAP_ADDRESS | mvs215.rtp.raleigh.ibm.com | 2809 |
| ■ ■ ■ | | | |
| <input type="checkbox"/> | SOAP_CONNECTOR_ADDRESS | mvs215.rtp.raleigh.ibm.com | 8880 |
| <input type="checkbox"/> | WC_adminhost | * | 9060 |
| <input type="checkbox"/> | WC_adminhost_secure | * | 9043 |
| <input type="checkbox"/> | WC_defaulthost | * | 9080 |
| <input type="checkbox"/> | WC_defaulthost_secure | * | 9443 |

Once the processing has finished and you have your deployment environment set up, you will most likely need to change some of the values that were used to match your installation's naming conventions. Shown here are some of the values you'll want to change, including the cluster name, server name and port values.

Severe errors seen during server start

Errors that can be ignored

- SECJ0384E: Trust Association Init Error. The Trust Association interceptor implementation `com.ibm.ws.security.web.TAMTrustAssociationInterceptorPlus` initialization failed. The error status/exception is -1. If you receive this error message in association with a trust association interceptor that you are not using, you can ignore this message.
- CWSPN0009E: SPNEGO Trust Association Interceptor configuration is not valid. Failure condition: `com.ibm.ws.security.spnego.isEnabled JVM` property is false or not set, no further processing is done.. If you are not using the SPNEGO TAI, you can ignore this message.
- CHFW0030E: Error starting chain `_InboundTCPProxyBridgeService` because of exception `com.ibm.wsspi.channel.framework.exception.RetryableChannelException`: An exception was thrown when attempting to start the `TCPProxyChannel`

A good way to start the verification process of your configuration is to look for 'SEVERE' messages in the job logs. These messages will often signal that you have a problem in your setup. Fix problems that you find which can include SQL errors or security errors. There are a few SEVERE errors that are expected however and can be ignored. These are listed on the slide.

Summary

- WebSphere Process Server for z/OS V7 and WebSphere Enterprise Service Bus for z/OS V7 network deployment configuration process
 - DB2 is a requirement
 - Deployment environment automates the cluster configuration

This presentation has stepped through the configuration process for WebSphere Process Server for z/OS or WebSphere Enterprise Service Bus for z/OS in a network deployment environment. DB2 becomes a requirement in this environment but scripts such as Db2DesignGenerator and createDB are used to make your job easier. Finally, the deployment environment automates the creation of a cluster, making it easier to get started in the network deployment environment.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WBPMv70_zOSNetworkDeploymentConfig.ppt

This module is also available in PDF format at: ..\\WBPMv70_zOSNetworkDeploymentConfig.pdf

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, DB2, Redbooks, System z, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the Web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

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

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2010. All rights reserved.