IBM WebSphere Application Server for z/OS, Version 8.0

Migrating, coexisting, and interoperating

GA32-0901-01

Note

Before using this information, be sure to read the general information under “Notices” on page 153.

Compilation date: July 26, 2011

© Copyright IBM Corporation 2011.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

How to send your comments v

Changes to serve you more quickly vii

Chapter 1. Migration, coexistence, and interoperability 1
Overview of migration, coexistence, and interoperability 1
Migrating and coexisting application servers 5
Premigration considerations 6
Migrating API and specifications 15

Chapter 2. How do I migrate, coexist, and interoperate? 17

Chapter 3. Migrating product configurations 19
Configuration mapping during product-configuration migration 20
Planning to migrate product configurations for z/OS 24
Migrating product configurations with migration tools 25
Using the z/OS Migration Management Tool to create and manage migration definitions 26
Using the zmmt command to create migration definitions on z/OS 32
clientUpgrade command 35
convertScriptCompatibility command 35
Migrating z/OS stand-alone application servers 38
Preparing to migrate a z/OS stand-alone application server 39
Migrating a z/OS stand-alone application server: Checklist 60
Migrating a stand-alone application server on the z/OS operating system 62
Migrating WebSphere Application Server, Network Deployment configurations on the z/OS operating system 65
Preparing to migrate a WebSphere Application Server, Network Deployment cell for z/OS 65
Migrating deployment managers for z/OS. 68
Migrating federated nodes for z/OS 91
Migrating IBM Cloudscape or Apache Derby databases 113
Migrating from the WebSphere Connect JDBC driver 115
Rolling back environments 117
Rolling back a WebSphere Application Server, Network Deployment cell 117
Rolling back a federated node 119
Rolling back stand-alone application servers 121

Chapter 4. Scenario 1: Migrating an administrative agent profile and its registered set of managed base application servers on z/OS 123

Chapter 5. Scenario 2: Migrating a job manager profile and its registered set of servers using the z/OS Configuration Migration Management Tool 127

Chapter 6. Migrating web server configurations 131

Chapter 7. Migrating administrative scripts 133
Migrating administrative scripts from a previously Version 5.1.x application server 133
Example: Migrating - Allowing configuration overwrite when saving configurations 134
Example: Migrating - Changing transaction log directory using wsadmin scripting 135
Migrating administrative scripts from Version 6.x or 7.x to Version 8.0 135

Chapter 8. Running multiple application server versions 137
Coexistence support 137

© Copyright IBM Corp. 2011

Chapter 9. Interoperating multiple application server versions 139

Chapter 10. Configuring port settings 141

Chapter 11. Troubleshooting migration 143

Appendix. Directory conventions 151

Notices 153

Trademarks and service marks 155

Index 157

How to send your comments

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

- To send comments on articles in the WebSphere Application Server Information Center
  1. Display the article in your Web browser and scroll to the end of the article.
  2. Click on the **Feedback** link at the bottom of the article, and a separate window containing an e-mail form appears.
  3. Fill out the e-mail form as instructed, and click on **Submit feedback** .
- To send comments on PDF books, you can e-mail your comments to: **wasdoc@us.ibm.com** or fax them to 919-254-5250.

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

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

© Copyright IBM Corp. 2011

Changes to serve you more quickly

Print sections directly from the information center navigation

PDF books are provided as a convenience format for easy printing, reading, and offline use. The information center is the official delivery format for IBM WebSphere Application Server documentation. If you use the PDF books primarily for convenient printing, it is now easier to print various parts of the information center as needed, quickly and directly from the information center navigation tree.

To print a section of the information center navigation:

1. Hover your cursor over an entry in the information center navigation until the **Open Quick Menu** icon is displayed beside the entry.
2. Right-click the icon to display a menu for printing or searching your selected section of the navigation tree.
3. If you select **Print this topic and subtopics** from the menu, the selected section is launched in a separate browser window as one HTML file. The HTML file includes each of the topics in the section, with a table of contents at the top.
4. Print the HTML file.

For performance reasons, the number of topics you can print at one time is limited. You are notified if your selection contains too many topics. If the current limit is too restrictive, use the feedback link to suggest a preferable limit. The feedback link is available at the end of most information center pages.

Under construction!

The Information Development Team for IBM WebSphere Application Server is changing its PDF book delivery strategy to respond better to user needs. The intention is to deliver the content to you in PDF format more frequently. During a temporary transition phase, you might experience broken links. During the transition phase, expect the following link behavior:

- Links to Web addresses beginning with http:// work
- Links that refer to specific page numbers within the same PDF book work
- The remaining links will *not* work. You receive an error message when you click them

Thanks for your patience, in the short term, to facilitate the transition to more frequent PDF book updates.

© Copyright IBM Corp. 2011

Chapter 1. Migration, coexistence, and interoperability

The goal of migration is to reconstruct the environment of your previous version of WebSphere® Application Server in a Version 8.0 environment. Coexistence allows you to create a mixed-version environment that is not in conflict and allows the nodes of all versions to start and run at the same time. Coexistence also facilitates rollback and allows one or the other version to run at one time. Interoperating is exchanging data between two coexisting product installations or between products on different systems.

Overview of migration, coexistence, and interoperability

The goal of migration is to reconstruct your earlier version of WebSphere Application Server in a Version 8.0 environment. Coexistence allows you to create a mixed-version environment that is not in conflict and allows the nodes of all versions to start and run at the same time. Coexistence also facilitates rollback and allows one or the other version to run at one time. Interoperating is exchanging data between two coexisting product installations or between products on different systems.

Frequently asked questions

Can I simply point to the new WebSphere Application Server for z/OS® Version 8.0 datasets and restart my servers?

No. WebSphere Application Server for z/OS Version 8.0 requires that you migrate your Version 6.x or 7.x configuration up to the Version 8.0 level.

Be aware of the following issues when migrating to Version 8.0:

- Any variables that belong to applications or products other than WebSphere Application Server are not migrated but are brought over to the new environment as is. Therefore, check any other product upgrades before migrating to ensure that all of these variables are still accurate after migration.
- Before performing migration from Version 6.x or 7.x to Version 8.0, verify that you do not have any region constraints (such as IEFUSI limits) in place. These constraints can cause unpredictable Java Virtual Machine (JVM) errors.

What is the basic migration process?

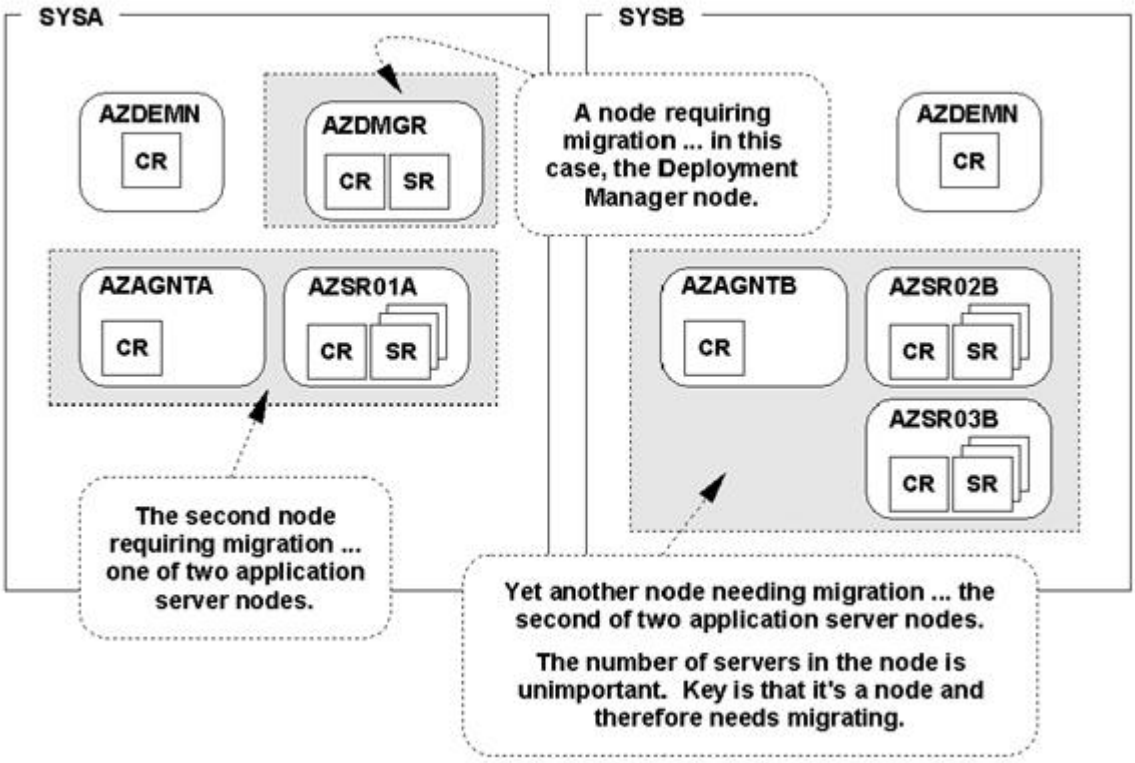
1. Install the SMP/E code for WebSphere Application Server for z/OS Version 8.0.
2. Use the z/OS Migration Management Tool or the zmmt command to create the migration utilities that you need to perform the migration.
3. Run these jobs.

   A new Version 8.0 configuration is created—separate from your existing Version 6.x or 7.x configuration—that is based on the Version 6.x or 7.x configuration information.

Is migration a node-by-node activity?

Yes. The process of migrating the configuration involves running the supplied utilities against each node in your configuration.

© IBM Corporation 2008



Although a stand-alone application server only has one node, that node needs to be migrated. The steps are essentially the same as the steps for migrating any other node, except that you do not have to have a deployment manager running. Read “Migrating a z/OS stand-alone application server: Checklist” on page 60 for a checklist of activities for migrating a stand-alone application server node.

What do the migration utilities do?

The migration utilities serve the following purposes:

Table 1. Migration utilities and their purposes. The table lists the various migration utilities and their purposes.

Utility	Purpose
BBOWMG1B (stand-alone application server migrations) BBOWMG1F (federated node migrations)	Enables all servers on the node being migrated to be configured to start in Peer Restart and Recovery (PRR) mode After this job completes, you must start all servers on the node being migrated and wait for them to stop. PRR processing mode resolves any outstanding transactions, clears the transaction logs, and stops the server. This job is not needed for a deployment manager migration, and it is optional for configurations that do not use distributed transaction (XA) connectors. This job is required only if you are using XA adapters and you need to migrate the XA logs. Check your resource providers in the Version Version 6.x or 7.x administrative console by going to **Resources > JDBC providers** and checking to see if you have chosen any XA providers such as DB2®, Apache Derby, and so on.

Table 1. Migration utilities and their purposes (continued). The table lists the various migration utilities and their purposes.

BBOWMG2B (stand-alone application server migrations) BBOWMG2F (federated node migrations)	Disables PRR mode and returns all servers to normal operating state You are not required to start all servers after this job completes. This job is not needed for a deployment manager migration, and it is optional for configurations that do not use XA connectors. This job is required only if you are using XA adapters and you need to migrate the XA logs. Check your resource providers in the Version 6.x or 7.x administrative console by going to **Resources > JDBC providers** and checking to see if you have chosen any XA providers such as DB2, Apache Derby, and so on.
BBOMBHFS or BBOMBZFS (stand-alone application server migrations) BBOMDHFS or BBOMDZFS (deployment manager migrations) BBOMMHFS or BBOMMZFS (federated node migrations)	**Optional:** Creates a file system and mount point for the Version 8.0 configuration root, and mounts the file system If you want to use an existing file system to contain the Version 8.0 configuration, you must manually create the mount point specified when you create the migration definition and verify that the file system is mounted rather than run this job. In either case, the configuration file system and mount point must be created and the file system must be mounted before proceeding with the migration.
For stand-alone application server migrations, the following utilities: • BBOWMG3B • BBOWBPRO • BBOWBPRE • BBOWBPOS For deployment manager migrations, the following utilities: • BBOWMG3D • BBOWDPRO • BBOWDPRE • BBOWDPOS For federated node migrations, the following utilities: • BBOWMG3F • BBOWMPRO • BBOWMPRE • BBOWMPOS	BBOWMG3*x* runs the complete migration of the node from Version 6.x or 7.x to Version 8. BBOW*x*PRO just creates the WebSphere Application Server home and default profile. BBOW*x*PRE just runs the migration pre-upgrade process. BBOW*x*POS just runs the migration post-upgrade and finish-up (change file permission) processes.
BBOMBCP (stand-alone application server migrations) BBOMDCP (deployment manager migrations) BBOMMCP (federated node migrations)	Copies the generated Job Control Language (JCL) procedures to start the servers to the specified procedure library If you choose to have your Version 8.0 configuration make use of different JCL start procedure names, this utility updates the new Version 8.0 configuration, substituting your new JCL names for the names that existed in your original Version 6.x or 7.x configuration.

Where should you run the migration jobs?

Run the jobs on the same system on which the node being migrated resides.

What happens when a node is migrated?

The migration utilities transform the contents of your present WebSphere Application Server Version 6.x or 7.x configuration file system and merge them into a new, separate Version 8.0 configuration file system.

Are my existing configuration lost during migration?

During the migration, the original WebSphere Application Server Version 6.x or 7.x configuration tree is unaffected. If for some reason the migration fails before completing, your previous configuration still exists.

If my node has multiple application servers, are all of them migrated?

Yes. The utility detects all servers and migrate all, including the node agent. One invocation of the migration utilities against the node migrates all the servers in the node.

Must I stop the servers in a node to perform the migration?

Yes. In a multinode configuration it is possible to have the other nodes still running. But any node that you want to migrate must have its servers stopped.

When an application server node that is part of a WebSphere Application Server, Network Deployment configuration is being migrated, the previously migrated Version 8.0 deployment manager for that cell must be running. This is because part of the migration involves the use of the wsadmin scripting function to synchronize the newly migrated application server node with the deployment manager. The deployment manager must be running in order to perform that synchronization.

Is it possible to have a cell operating with only some of the nodes migrated and others not?

Yes, that is possible. WebSphere Application Server Version 6.x or 7.x can coexist with Version 8.0 in the same cell and on the same logical partition (LPAR).

Can my newly migrated WebSphere Application Server for z/OS Version 8.0 deployment manager still communicate with the Version 6.x or 7.x nodes?

Yes. A deployment manager that is migrated to the Version 8.0 level of code can manage a Version 6.x or 7.x node. Changes made through the administrative console are applied to the node. Remember the following points:

- When a deployment manager is migrated to Version 8.0, a new Version 8.0primary configuration is created. The Version 6.x or 7.x primary configuration still exists. But when the Version 8.0 deployment manager makes changes to the configuration, the changes are made to the new Version 8.0 primary configuration. While it is still possible to use the Version 6.x or 7.x code, therefore, any changes made in Version 8.0 are not seen when the older code is restored.
- A Version 6.x or 7.x deployment manager has no ability to manage a Version 8.0 node.
- A WebSphere Application ServerVersion 8.0 WebSphere Application Server, Network Deployment cell can contain mixed releases of Version 6.x nodes, but there is no mixed-node management support for Version 6.0.1.x.

  The Version 8.0 migration tools still migrate these nodes during deployment-manager migration, but they issue a warning message that the nodes cannot be managed by the Version 8.0 deployment manager. You can then perform one of the following actions based on your needs.

  – Upgrade all Version 6.0.1.x nodes to at least Version 6.0.2.

This action allows the nodes to be administered by a Version 8.0 deployment manager.

– Migrate these nodes to Version 8.0.

Is there a sequence to performing a multinode migration?

Yes. Migrate according to the following sequence:

1. Always migrate the deployment manager first.
2. Application server nodes on the same system as the deployment manager or on other Multiple Virtual Storage (MVS™) images can then be migrated.

Is it possible to have cells at WebSphere Application Server for z/OS Version 8.0 coexist with other cells at Version 6.x or 7.x?

Yes. It possible to have cells at WebSphere Application Server for z/OS Version 8.0 coexist with other cells at Version 6.x or 7.x for a sysplex or any given MVS image. The following restrictions exist:

- A cell can contain servers at Version 6.x or 7.x levels.
- A cell can contain z/OS and non-z/OS nodes; however, the deployment manager must be at the highest version level in the cell and any nodes on platforms other than that on which the deployment manager is located must be at Version 6.0 or later.
- A server on a z/OS node cannot be clustered with a server on a non-z/OS node.
- An LPAR can contain more than one node from the same cell.
- Each LPAR has at most one daemon per cell with servers on that LPAR regardless of how many nodes from that cell are configured for that LPAR.
- For a given LPAR, a daemon must be at or above the version level of all servers on that LPAR that are in the daemon's cell, regardless of node.
- All servers in the same node must be at the same version level.
- The deployment manager must be at or above the version level of any server in the cell.
- The controller and its servants must be at the same version level.
- No two cells can have the same cell short name.
- Other considerations exist for separate cells, regardless of whether they are at different versions of the code. For example, you must have a separate configuration file system mount point and separate JCL procedures.

Migrating and coexisting application servers

Migrating involves collecting the configuration information from a previous release of a WebSphere Application Server and merging it into a configuration for a new release. Coexisting involves running a new release of a WebSphere Application Server and an earlier release simultaneously on the same machine.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6. For resources to help you plan and perform your migration, visit Knowledge Collection: Migration planning for WebSphere Application Server.

The migration tools basically save the existing WebSphere configurations and user applications in a backup directory and then process the contents of this backup directory to migrate the configurations and your applications from previous WebSphere Application Server releases to the latest release.

If you have a previous version of WebSphere Application Server, you must decide whether to migrate the configuration and applications of the previous version to the new version.

Migration does not uninstall the previous version.

- For stand-alone application server migrations and for deployment-manager migrations in which you do not choose to disable the previous deployment manager during migration, the earlier release is still functional.
- For federated-node migrations and for deployment-manager migrations in which you do choose to disable the previous deployment manager during migration, the earlier release is disabled after migration completes successfully. You can re-enable the earlier version using the migrationDisablementReversal.jacl script.

If you run two different versions of the application server at the same time, the two versions are *coexisting*. For example, if Version 6.1 and 8.0 application servers are running on the same machine, they are coexisting.

For help in troubleshooting problems when migrating, read Chapter 11, “Troubleshooting migration,” on page 143.

About this task

For information on migrating to Version 8.0, read Chapter 3, “Migrating product configurations,” on page 19. For more information on coexistence among releases, read “Coexistence support” on page 137.

Procedure

1. Update product prerequisites and corequisites to supported versions.

   Refer to the IBM® WebSphere Application Server supported hardware, software, and APIs site for current requirements.
2. Install the Version 8.0 product.
3. Migrate your WebSphere Application Server Version 6.x product configuration to Version 8.0.
4. Migrate web server plug-ins as described in Chapter 6, “Migrating web server configurations,” on page 131.
5. Optional: Set up multiple versions of WebSphere Application Server to coexist.

Premigration considerations

Before you begin the process of migrating to WebSphere Application Server Version 8.0, there are some considerations of which you need to be aware.

Considerations for z/OS

Consider the following before migrating Application Server for z/OS:

- The migration progress usually carries forward the values for user-defined variables or variables that are defined by Application Server. There are some special variables, however, for which the values are not carried forward. These values are:
  - WAS_INSTALL_ROOT
  - USER_INSTALL_ROOT
  - WAS_PRODUCT_ROOT
  - WAS_LIBS_DIR
  - WAS_PROPS_DIR
  - WAS_TEMP_DIR
  - APP_INSTALL_ROOT
  - DRIVER_PATH
  - WAS_INSTALL_LIBRARY

- JAVA_HOME
- DEPLOY_TOOL_ROOT
- CONNECTOR_INSTALL_ROOT
- TRANLOG_ROOT
- MQJMS_LIB_ROOT
- WAS_ETC_DIR
- CLOUDSCAPE_JDBC_DRIVER_PATH
- WEMPS_USER_ROOT
- MQ_INSTALL_ROOT
- WAS_DAEMON_ONLY_ICU_DATA
- WAS_DAEMON_ONLY_CONFIG_ROOT
- WAS_DAEMON_primordial_root
- WAS_DAEMON_daemon_was_env_file

The values for the these variables are picked up from the profile template when migration creates profiles. If you defined custom values for these variables, they are not maintained in the Application Server release to which you are migrating. These variables are not migrated, because they are Application Server properties that are profile attributes. If you need to override their values with entries that are not the default, you must change the value outside of the product install, profile creation, and migration processes.

Note: Be particularly aware of the APP_INSTALL_ROOT variable. Even if the APP_INSTALL_ROOT had a custom location that you defined, by default the migration process will install applications in the following location:

```
${USER_INSTALL_ROOT}/installedApps
```

To have the migration process honor a an application install location that is not the default:

- Specify the location in the Application Install Directory field of zMMT.
- Select the **Migrate applications and use the previous application installation directory** option during migration to honor the install path without the need to enter it again. If the install path for the previous release contains variable references, Application Server will resolve those variables with the migrated values.

- After you install WebSphere Application Server Version 8.0 on the z/OS operating system, you might want to build a complete WebSphere Application Server, Network Deployment cell configuration and verify that it works correctly before you attempt to migrate an existing cell or node.

  This process ensures that your system has all of the necessary prerequisites and supports the new level of the product.
- Before you perform the migration, evaluate the items deprecated in WebSphere Application Server Version 8.0.

  For more information, read the "Deprecated and removed features" article in the information center.
- Before you migrate from WebSphere Application Server Version 6.x to Version 8.0, connect the application servant region ID to a keyring and verify that the keyring has the WebSphereCA certificate associated with it. Otherwise, security errors occur if you have global security turned on.
- High availability manager and core group functionality are included in WebSphere Application Server Version 6.0 and later.

  Read the "Core group migration considerations" article in the information center for core-group configuration and topology considerations that might impact your migration from Version 6.x to Version 8.0.
- If you are migrating from WebSphere Application Server Version 6.0.1 to Version 8.0 and security is enabled for Version 6.0.1, you must upgrade to service level 6.0.2.12 or later before migrating to Version 8.0.

If security is not enabled for WebSphere Application Server Version 6.0.1, this requirement does not apply and there is no specific service-level requirement for the migration.

- If you are migrating from WebSphere Application Server Version 6.0.2.x to Version 8.0 and your configuration uses XA connections, you must have Version 6.0.2.9 or later before migrating.
- If you plan on running an Internet Inter-ORB Protocol (IIOP) client that is earlier than Version 6.1 and it will be interacting with a Version 8.0 server on the same logical partition (LPAR), the daemon procedure library for the Version 8.0 daemon must include the earlier release's SBBOLD2 and SBBOLPA libraries in its STEPLIB.
- Migration support requires that both the source and target WebSphere Application Server systems be on the LPAR.

  Therefore, you cannot migrate an existing configuration to a different z/OS LPAR. You also cannot migrate to or from a non-z/OS operating system using the WebSphere Application Server Version 8.0 migration utilities.
- Migrating cells that span Sysplex environments or operating systems should not present any unique migration issues. You migrate at the node level, and you use the tools provided based on the platform of the node that you are migrating.

  For information on setting up a mixed-platform cell, read the WebSphere for z/OS -- Heterogeneous Cells white paper.
- WebSphere Application Server on the z/OS operating system does not support the WASPreUpgrade, WASPostUpgrade, and manageprofiles command-line tools that are supported by the distributed and i5/OS® versions of the product.

  You must generate the migration jobs using the z/OS Migration Management Tool or the zmmt command and then submit them according to the generated instructions.
- The amount of storage that your system requires during migration to Version 8.0 depends on your environment.
  - The size of the file system is provided by the value of the primary allocation in cylinders specified in the z/OS Migration Management Tool or the zmmt command when you generate a migration definition. In addition, this file system is automatically expandable based on the secondary allocation that you specify. The default values provided for these allocations by the migration tools is normally sufficient for the configuration data.
  - The migration backup directory requires a considerable amount of temporary space. The exact amount needed depends on several factors, but 100 cylinders is sufficient in most situations (including tracing if needed). If you are in doubt about the space available for your temporary directory, use the option in the z/OS Migration Management Tool or the zmmt command to move the temporary directory to a custom location with more space and mount your own temporary file system there. Failure to have adequate temporary space can lead to a premature end to the migration process.
- Before you migrate to Java Standard Edition (SE) Development Kit (JDK) 6 from JDK 5 or JDK 1.4 , review your applications for necessary changes based on the Sun Microsystems Java specification.

  Read “Migrating API and specifications” on page 15 for more information.
- When migrating a cell with multiple nodes, the applications must remain at lowest JDK level until all nodes are migrated.
- The migration articles assume that WebSphere Application Server Version 8.0 is being installed in an environment where it must coexist with prior levels of WebSphere Application Server.

  Consider the following items while planning to enable coexistence:
  - Update prerequisites to the levels required by WebSphere Application Server Version 8.0.

    Prior levels of WebSphere Application Server continue to run aWebSphere Application Servert the higher prerequisite levels.
  - Review the ports that have been defined to ensure that the WebSphere Application Server Version 8.0 installation does not conflict.

In particular, note that the default daemon port definition for both versions is the same when installing to coexist with WebSphere Application Server Version 6.x.

Read the "Default port assignments" article in the information center for default port information.

Read Chapter 8, “Running multiple application server versions,” on page 137 for more information.

- Consider the following information if you are planning to have any mixed-release cells:
  - You can upgrade a portion of the nodes in a cell to WebSphere Application Server Version 8.0 while leaving others at the previous release level. This means that, for a period of time, you might be administering servers that are at the previous release level and servers that are running the newer release in the same cell.

    In this mixed-release environment, some restrictions on what you can do with servers at the previous release level might exist. For details, read the "Creating application servers" article in the information center. Restrictions on creating clusters and cluster members might also exist. For details, read the "Creating clusters" article in the information center.
  - A WebSphere Application Server Version 8.0 WebSphere Application Server, Network Deployment cell can contain mixed releases of Version 6.x nodes; however, there is no mixed-node management support for Version 6.0.1.x.

    The Version 8.0 migration tools still migrate these nodes during deployment-manager migration; however, these tools issue a warning message that the nodes cannot be managed by the Version 8.0 deployment manager. You can then perform one of the following actions based on your needs.
    - Upgrade all Version 6.0.1.x nodes to at least Version 6.0.2. This upgrade allows them to be administered by a Version 8.0 deployment manager.
    - Migrate these nodes to Version 8.0.
- During migration, Version 8.0 cluster information is distributed throughout the cell. Version 6.0.x nodes that are not at Version 6.0.2.11 or later fail to read this information and the cluster function might fail. Therefore, upgrade all Version 6.0.x nodes that will be contained in or interoperating with a Version 8.0 cell to Version 6.0.2.11 or later before migrating your deployment managers to Version 8.0.
- WebSphere Application Server Version 8.0 migration converts HTTP transports to channel-framework web container transport chains.

  For more information on WebSphere Application Server Version 8.0 transport support, read the following articles in the information center:
  - Configuring transport chains
  - HTTP transport channel settings
  - Transport chains
- Include maintenance considerations when developing your configuration file system strategy.

  If you configure your WebSphere Application Server, Network Deployment environment using the default value for the product file system path in the z/OS Migration Management Tool, all the nodes point directly at the mount point of the product file system. This configuration makes rolling maintenance in a nondisruptive way almost impossible. If a cell is configured in this way, applying service to the product file system affects all the nodes at the same time. If multiple cells are configured in this way, applying service to the product file system affects all the cells at the same time.

  You might want to specify what is referred to as an "intermediate symbolic link" between each node's configuration file system and the actual mount point of the product file system. This strategy is described in the WebSphere Application Server for z/OS V5 - Planning for Test, Production and Maintenance white paper.

  Read the Washington Systems Center Sample WebSphere for z/OS ND Configuration white paper for more information about this issue and its relationship to applying maintenance. Read the WebSphere for z/OS: Updating an Existing Configuration HFS to Use Intermediate Symbolic Links instructions for information on obtaining and using a utility that you can use to update an existing configuration Hierarchical Hile System (HFS) to use intermediate symbolic links.

- The migration tools create a migration backup directory containing a backup copy of the configuration from the previous version. The space available for this directory must be at least the size of the configuration directory and applications from the previous version plus the size of the batch-job output from the migration.

  Typically, the batch-job output from the migration is very small unless you enable trace. The trace output size varies depending on the parts of migration for which you have enabled trace. The largest trace producer is the WASPostUpgrade phase of migration. You can typically see trace output of approximately 30 MB for this phase.
- WebSphere Application Server Version 8.0 does not support the DB2 for zOS Local JDBC Provider (RRS).

  If you use the DB2 for zOS Local JDBC Provider (RRS) in the Version 6.x configuration that you are going to migrate, you must change your configuration to use the DB2 Universal JDBC Driver Provider before or immediately after you migrate to Version 8.0. The Version 8.0 migration tools do not migrate the provider for you.

  If you use the DB2 for zOS Local JDBC Provider (RRS) in the version to be migrated and do not change your configuration to use the DB2 Universal JDBC Driver Provider before migrating to Version 8.0, the following events occur:

  – When running the migration tools, you receive the following message:

    ```
    MIGR0442W: Not migrating DB2 for zOS Local JDBC Provider (RRS) jdbc provider.
    Manually create a DB2 Universal Driver provider as a replacement. See DB2
    documentation for further details.
    ```

  – After migration, DB2 access is broken and you receive the following runtime message:

    ```
    DSRA8213W: JDBC provider, DB2 for zOS Local JDBC Provider (RRS), is no longer
    supported by WebSphere Application Server. Applications should use DB2 Universal
    JDBC Driver Provider Type 2.
    ```

  If you determine that you must change your configuration to use the DB2 Universal JDBC Driver Provider, you can do so by completing one of the following tasks:

  – Before migrating to Version 8.0, change your Version 6.x configuration to use the DB2 Universal JDBC Driver Provider.

    If you do this, the Version 8.0 migration tools handle the migration to the DB2 Universal JDBC Driver Provider and no post-migration activity id required.

    Perform one of the following actions:

    - Manually change your configuration to use the DB2 Universal JDBC Driver Provider.

      Search the information center for your Version 6.0.2 product to find information on configuring a DB2 Universal JDBC Driver Provider for WebSphere Application Server running on the z/OS operating system.
    - If you are migrating from Version 6.1.0 or later, use the JDBC Migration Utility for DB2 on z/OS to migrate from the DB2 for zOS Local JDBC Provider (RRS) to the DB2 Universal JDBC Driver Provider.

      This tool is a script that migrates DB2 for zOS Local JDBC Providers (RRS) to DB2 Universal JDBC Driver Providers one node at a time. A white paper that accompanies the tool explains how to install and configure the DB2 Universal JDBC Driver before running the tool to migrate your configuration. The tool and white paper are available at http://www.ibm.com/support/docview.wss?uid=swg27007826.

  – After migrating to Version 8.0, perform one of the following actions:

    - Manually change your configuration to use the DB2 Universal JDBC Driver Provider.
    - Use the JDBC Migration Utility for DB2 on z/OS to migrate from the DB2 for zOS Local JDBC Provider (RRS) to the DB2 Universal JDBC Driver Provider.

      This tool is a script that migrates DB2 for zOS Local JDBC Providers (RRS) to DB2 Universal JDBC Driver Providers one node at a time. A white paper that accompanies the tool explains how

to install and configure the DB2 Universal JDBC Driver before running the tool to migrate your configuration. The tool and white paper are available at http://www.ibm.com/support/docview.wss?uid=swg27007826.

- After you migrate a base application server to WebSphere Application Server Version 8.0 running on the z/OS operating system, the administrative and user applications continue to be defined under the virtual host default_host as they were in the previous release. However, a migrated deployment manager is defined under the virtual host admin_host that was introduced in Version 6.1.
- If you use isolated data repositories—specifically, nonshared data repositories such as transaction logs for SIB and IBM Cloudscape or Apache Derby databases—and you migrate from a previous release, your existing databases and transaction logs are saved. If you have mission-critical information that is stored in these local data repositories, you should safely shut down all servers that interact with those repositories before attempting migration. Those servers should remain offline until the migration has been successfully completed or rolled back.

  After the migration is complete or you have rolled back to the previous version, you can restart the servers that interact with these isolated data repositories.
- Before you migrate an Apache Derby database, ensure that any application servers hosting applications that are using the Apache Derby database are closed. Otherwise, the Apache Derby migration fails.
- You should be aware of the following rules related to migrating security domains:
  - If you migrate a deployment manager that has a security domain with a cell-level scope, the migration tools take the following actions:
    - Migration creates a domain in the new configuration called *PassThroughToGlobalSecurity* if it does not already exist.
    - Migration adds a cluster mapping to the new configuration for all clusters that existed in the old configuration.
      - Clusters that only existed in the Version 8.0 deployment-manager configuration before migration do not have their mappings to PassThroughToGlobalSecurity changed.
        - If mappings for the Version 8.0 clusters exist before migration, they still exist after migration.
        - If no mappings for the Version 8.0 clusters exist before migration, they still do not exist after migration.
      - If a cluster exists in both the previous configuration and the Version 8.0 configuration before migration, the cluster in the new configuration is added to the PassThroughToGlobalSecurity domain and behaves like the cluster in the previous release.
    - Migration adds a bus mapping for any busses that exist in a migrated Version 6.1.x configuration.

      Bus mappings are updated following the same rules as those for cluster mapping that are described above.
    - Administrative servers (deployment manager) are not added to the PassThroughToGlobalSecurity domain.
  - If you migrate a federated node that has a security domain with a cell-level scope, the migration tools take the following actions:
    - Migration creates a domain in the new configuration called *PassThroughToGlobalSecurity* if it does not already exist.
    - Migration adds a server-level mapping to the PassThroughToGlobalSecurity domain for all non-clustered servers in the old node's configuration.
      - Servers on the node that is being migrated that are part of a cluster do not receive entries in the PassThroughToGlobalSecurity domain because this was addressed through a cluster mapping during deployment-manager migration.

        If you have removed that mapping, migration maintains that behavior.
      - Administrative servers (node agents) are not added to the PassThroughToGlobalSecurity domain.

Read the "Security domains in a mixed-version environment" section of the "Multiple security domains" article in the information center.

- Modifications to the java.security file are lost during migration. If you've updated your java.security file in the previous version of WebSphere Application Server, the new version will not have your updates. For example, if you modified your provider list to use FIPS or Hardware crypto, your migrated cell will not use these providers, and therefore will not use FIPS or Hardware Crypto. To migrate your modified java.security file, use the following steps:
  1. Before migrating, copy the java.security file from the old `WAS_HOME/java/lib/security` to a temporary directory.
  2. Migrate WebSphere Application Server to Version 8.0.
  3. Compare the old java.security file with the new Version 8.0 java.security file that is located in `V8WAS_HOME/properties/java.security`.
  4. Update `V8WAS_HOME/properties/java.security` with changes to the provider list from the old java.security file.
  5. Repeat this procedure for each installation of WebSphere Application Server.
- After you use the migration tools to migrate to WebSphere Application Server Version 8.0, you might need to perform some actions that are not performed automatically by the migration tools.
  – Examine any Lightweight Third-Party Authentication (LTPA) security settings that you might have used in WebSphere Application Server Version 6.x, and verify that Version 8.0 security is set appropriately.

    Read the "Lightweight Third Party Authentication" article in the information center for more information.
  – If necessary, create new System Authorization Facility (SAF) profiles before you start your migrated servers on WebSphere Application Server Version 8.

    Beginning with Version 6.1, certain security facilities are controlled using SAF profiles.

    - In Version 6.1 and later, the **Enabling trusted applications** setting is controlled with a SAF security profile instead of an internal WebSphere variable as in previous releases.

      The **Enabling trusted applications** option, which permits the WebSphere Application Server run time to perform certain privileged operations on behalf of application code, is required for all servers that use the LocalOS registry or SAF authorization.
    - In Version 6.1 and later, the **Sync to OS thread allowed** feature (which allows an application to access resources using an operating system identity other than the server identity) is controlled with a SAF security profile and the com.ibm.websphere.security.SyncToOSThread variable.

      This implementation allows both the administrator and the system security administrator to determine whether or not the feature is used. This implementation also allows restrictions on which identities the application can assume.

    If you migrate from a previous version of WebSphere Application Server and need these features, you must create the required SAF profiles. If these profiles are not present and correctly set up, a cell using the LocalOS user registry or SAF authorization fails when started on Version 8.0.

    If you use Resource Access Control Facility (RACF®) for your security system, use the following instructions. If you use another SAF-compliant security system, contact the security system vendor for appropriate information.

    - Check your Multiple Virtual Storage (MVS) system log or use the administrative console to determine whether or not **Enable trusted applications** is enabled for your server.

      Look for `control_region_security_enable_trusted_applications` in the startup log. If this value is set to 1, **Enabled trusted applications** is enabled. If this option is enabled, create the following SAF profile and grant READ access to the application server control region user ID:

      `BBO.TRUSTEDAPPS.`*`cell_shortname.cluster_transition_name`*

      Use the following RACF commands to complete this action:

```
RDEFINE FACILITY
  BBO.TRUSTEDAPPS.cell_shortname.cluster_transition_name
  UACC(NONE)
PERMIT FACILITY
  BBO.TRUSTEDAPPS.cell_shortname.cluster_transition_name
  ID(controller_userid) ACCESS(READ)
SETROPTS RACLIST(FACILITY) REFRESH
```

The *cluster_name* SAF facility profile is replaced by the cluster transition name for unclustered servers. If you want all servers in the cell to have **Enabling trusted applications** enabled, replace the cluster name with a wild card (*).

For more information, read the "System Authorization Facility classes and profiles" article in the information center.

- Check your Multiple Virtual Storage (MVS) system log or use the administrative console to determine whether or not **Sync to OS thread allowed** is enabled for your server.

  If this option is enabled, create the following SAF profile and grant either READ or CONTROL access to the application server control region user ID:

  ```
  BBO.SYNC.cell_shortname.cluster_transition_name
  ```

  The following example contains RACF commands that you might use to complete this action:

  ```
  RDEFINE FACILITY
    BBO.SYNC.cell_shortname.cluster_transition_name
    UACC(NONE)
  PERMIT FACILITY
  BBO.SYNC.cell_shortname.cluster_transition_name
    ID(controller_userid) ACCESS(CONTROL)
  SETROPTS RACLIST(FACILITY) REFRESH
  ```

  The cluster name is replaced by the cluster transition name for unclustered servers. If you want all servers in the cell to have **Sync to OS thread allowed** enabled, replace the cluster name with a wild card (*).

  Important:

  - Granting the control region READ access to the application server control region user ID restricts the user IDs to which the thread identity can be changed based on SAF SURROGAT profiles.

    If the controller user ID has READ access to the BBO.SYNC profile and the com.ibm.websphere.security.SyncToOSThread variable is set to true, an application might request Sync to the OS Thread. The application might assume the identity of either the caller or a role-related user ID to access resources. In order for the servant to run with a different application ID, however, the servant needs READ access to the SURROGAT profile BBO.SYNC.*application_userid*.
  - Granting the control region CONTROL access to the application server control region user ID allows the thread identity to be switched to any user ID that requests Sync to OS thread.

    If the controller user ID has CONTROL access to the BBO.SYNC profile and the com.ibm.websphere.security.SyncToOSThread variable is set to true, then an application might request Sync to OS Thread. The application might assume the identity of either the caller or any role-related user ID to access resources. SURROGAT profiles are not checked.

  For more information, read the "Application Synch to OS Thread Allowed" article in the information center.

– If you use SAF EJBROLE profiles for role-based authorization, create EJBROLE profiles for the two administrative roles—the deployer and admin security manager roles—that were introduced in Version 6.1.

– Review your Java Virtual Machine (JVM) settings to verify that you are using a heap size of at least 50 for improved startup performance.

Read the "Java virtual machine settings" article in the information center for more information.

If you have used a smaller heap size before, you can use the default heap size of 50.

– Verify the results of the automatic Apache Derby database migration, and manually migrate any Apache Derby databases that are not automatically migrated by the tools.

  Read “Migrating IBM Cloudscape or Apache Derby databases” on page 113 for more information.

– WebSphere Application Server Version 8.0 does not include the WebSphere Connect JDBC driver for SQL Server. The WebSphereConnectJDBCDriverConversion tool is provided to convert data sources from the WebSphere Connect JDBC driver to the DataDirect Connect JDBC driver or the Microsoft SQL Server JDBC driver.

  Read “Migrating from the WebSphere Connect JDBC driver” on page 115 for more information.

– During cell migrations to Version 8.0, the Version 6.0.x server user ID is not migrated or mapped to the Version 8.0 **Primary administrative user name** field as shown in the administrative console at **Security > Global security > *user_account_repository* > Configure**. The server user identity value is transferred, and the proper user ID from the Version 6.0.x cell is migrated along with a password.

  The primary administrative user name is required when defining Lightweight Directory Access Protocol (LDAP) as the security repository in Version 8.0.

  The **Primary administrative user name** field and server user identity value are used by the product when you use administrative scripts (wsadmin) to update the configuration. Version 6.0.x and earlier versions of the product store a valid username-and-password pair in the authenticating registry inside the local product configuration files. Beginning with Version 6.1, an administrative user ID can be set but the authentication information is not stored in the local product configuration files (thereby improving security).

  If you migrate from an older release of the product to Version 6.1 or later, the old model is still used in order to maintain backward compatibility and to allow the mixed cell to function properly. When you are ready to upgrade to the new model, which requires that all nodes in the cell be at Version 6.1.x or later, you can perform the following actions:

  - Input a Primary administrative user name that represents a valid user identity in the authentication authority (user registry).
  - Change the server user identity value from **Server identity that is stored in the repository** to **Automatically generated server identity**.

  If you are still using a mixed cell in Version 8.0 that contains Version 6.0.x nodes, you must continue to use the old model or the older nodes will not be able to communicate with the new nodes. You should not make any changes to the security settings in the mixed cell. If editing the user registry becomes necessary in a mixed cell that is not ready to be upgraded to the new model, however, select a valid user identity in the authentication authority (user registry) and put that value into the **Primary administrative user name** field but do **not** change the value of the server user identity. On all platforms, it is important to select a user ID that is valid for the scripting tools (wsadmin).

– If you have multiple node agents on the same logical partition (LPAR), IPC_CONNECTOR_ADDRESS port conflicts might exist after running the migration jobs. Reconfigure the conflicting ports.

– If you have applications that attempt to send a request to a Session Initiation Protocol (SIP) Uniform Resource Identifier (URI) over Transport Layer Security (TLS), you should be aware of a difference in behavior between WebSphere Application Server Version 6.1 and Version 8.0.

  When a SIP application sends a request to a SIP URI over TLS in Version 6.1, the request URI scheme changes from "sip" to "sips". In Version 8.0, the scheme remains without a change.

  This difference is seen when your applications attempt to send a SIP request in which the request URI contains a "sip" scheme and a "tls" transport parameter. For example, if an application in Version 6.1 creates a request with the following request URI:

  ```
  sip:alice@atlanta.com;transport=tls
  ```

and sends it to the network, the SIP container changes both the scheme and the transport parameter to produce the following request URI:

```
sips:alice@atlanta.com;transport=tcp
```

In Version 8.0, the SIP container does not change the scheme.

If you want to retain the old behavior after migrating the Version 6.1 application server to Version 8.0, change the application code. If the application intends to send a "sips" URI, it should create the URI that way before requesting to send out the message. With a "sips" URI, the behavior remains the same as in Version 6.1.

Migrating API and specifications

Migrating application programming interfaces (APIs) and specifications involves moving to the current Java component level as well as to other technologies that WebSphere Application ServerVersion 8.0 supports. If your existing applications currently support different specification levels than are supported by this version of the product, it is likely that you must update at least some aspects of the applications to comply with the new specifications.

In many cases, IBM provides additional features and customization options that extend the specification level even further. If your existing applications use IBM extensions from earlier product versions, it might be necessary for you to perform mandatory or optional migration to use the same kinds of extensions in Version 8.0.

Notes on the use of Java SE Development Kit 6:

- WebSphere Application Server began supporting Java SE Development Kit (JDK) 6 in Version 8.0.

  Read JSR 270: Java SE 6 Release Contents and Java SE 6 for more information about JDK 6.
- In general, existing Version 6.x application binaries that were developed using JDK 1.4 and 5 are highly compatible and typically do not require modifications to run. However, recompilation of the JDK 1.4 or 5 applications at the JDK 6 level might necessitate modifications of the source code to conform to incompatible changes that are present in JDK 6. As part of your migration planning, you should review the JDK compatibility restrictions that are documented by Sun Microsystems at Java SE 6 Release Notes®: Compatibility.
- A mixed cell containing Version 6.x and Version 7.0 nodes requires that all application binaries deployed on Version 6.x remain at the lowest JDK level associated with the Version 6.x nodes. Although you can successfully migrate Version 6.x applications to Version 8.0, this is only meant to be a temporary state as you transition to Version 8.0. After you begin migration to Version 8.0, plan to complete the migration of the entire cell, update your tooling to Version 8.0, and update your applications to conform to JDK 6 requirements. Complete this action before any further application changes. After you have completely migrated your cell to Version 8.0, upgrade your application binaries to the JDK 6 level the next time that you

make application modifications that require recompiling. This action might require source code changes to your application to conform to the JDK 6 API changes as documented by Sun Microsystems.

Note: The Java Virtual Machine Debug Interface (JVMDI) and the Java Virtual Machine Profiler Interface (JVMPI) were deprecated in JDK 5 and removed in JDK 6. Read Java SE 6 Release Deprecated API for more information.

Read the "Specifications and API documentation" article in the information center for a summary of the specifications and API documentation supported in current and prior product releases.

For more information on the items deprecated in WebSphere Application ServerVersion 8.0, read the "Deprecated, stabilized, and removed features" article in the information center.

Chapter 2. How do I migrate, coexist, and interoperate?

Use the documentation provided to answer your questions about migration, coexistence, and interoperability.

Follow these shortcuts to get started quickly with popular tasks. When you visit a task, look for the **IBM Suggests** feature at the bottom of the page. Use it to find available tutorials, demonstrations, presentations, articles, IBM Redbooks, support documents, and more.

“Configuration mapping during product-configuration migration” on page 20

Chapter 3, “Migrating product configurations,” on page 19

Chapter 6, “Migrating web server configurations,” on page 131

Review the software and hardware prerequisites

Chapter 8, “Running multiple application server versions,” on page 137

Chapter 10, “Configuring port settings,” on page 141

Chapter 9, “Interoperating multiple application server versions,” on page 139

Chapter 11, “Troubleshooting migration,” on page 143

© IBM Corporation 2004, 2008

Chapter 3. Migrating product configurations

Use the WebSphere Application Server Version 8.0 migration tools to migrate your product configurations. These migration tools support migration from Version 6.x or 7.x.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6. For resources to help you plan and perform your migration, visit Knowledge Collection: Migration planning for WebSphere Application Server.

The following configuration upgrades of WebSphere Application Server versions and offerings are directly supported.

Table 2. Profiles that can be migrated from Version 6.x or Version 7.x to Version 8.0. The table lists the supported profiles that can be migrated to WebSphere Application Server Version 8.0.

Migration Source	Migration to WebSphere Application Server for z/OS Version 8.0 Target supported?	
	Version 6.x	Version 7.x
Stand-alone application server profile	Supported	Supported
Deployment Manager profile	Supported	Supported
Managed (custom) application server profile	Supported	Supported
Administrative agent	Not supported	Supported
Job Manager	Not supported	Supported
* WebSphere Application Server for z/OS Version 8.0 supports the migration of a subset of programming model extensions (PMEs) from WebSphere Business Integration Server Foundation.		

Before using the migration tools, consult the IBM WebSphere Application Server supported hardware, software, and APIs website to understand what fixes you must apply to earlier versions. Applying fixes to an earlier version might also apply fixes to files that have a role in the migration. Apply any fixes to ensure the most effective migration of configurations and applications.

Note: The deployment manager maintains the master configuration data for all of the nodes that it manages. This configuration data is updated through the configuration manager. When the configuration manager detects that the updates to the configuration data were not made against the latest saved copy, it reject the updates and creates an exception. To avoid this situation, following these best practices:

- Migrate each node independently. For example, let the first node complete the migration process before starting the process for the second node, and so on.
- Ensure that the administrative console for the deployment manager is not running when the migration process is in progress.

If you need to migrate federated nodes concurrently, use the following steps to minimize the potential failures:

1. Run the backupConfig command for the deployment manager and each federated node before you begin. For more information, see the documentation about the backupConfig command.
2. Stagger the start of the migration process for each node by 3 to 5 minutes.
3. Run the restoreConfig command on that node and rerun the migration process if a failure occurs. For more information, see the documentation about the restoreConfig command.

© IBM Corporation 2002, 2008

Configuration mapping during product-configuration migration

Various configurations are mapped during product-configuration migration.

Migration involves the copying of your configuration from a previous release of a WebSphere Application Server into a new release.

Many migration scenarios are possible. The migration tools map objects and attributes existing in the version from which you are migrating to the corresponding objects and attributes in the Version 8.0 environment.

Bootstrap port

The migration tools carry the old release value into the Version 8.0 environment.

Command-line parameters

The migration tools convert appropriate command-line parameters to Java Virtual Machine (JVM) settings in the server process definition. Most settings are mapped directly. Some settings are not migrated because their roles in theWebSphere Application Server Version 8.0 configuration do not exist, have different meanings, or have different scopes.

For information on how to change the process-definition settings, read the "Process definition settings" article in the information center. For information on how to change the JVM settings, read the "Java virtual machine settings" article in the information center.

Generic server

In Version 6.x and later, a generic server has its own type, called GENERIC_SERVER. Migration will perform this conversion, but migration cannot accurately migrate the external resources that the generic server references. After migration has completed migrating the generic server settings, you might need to perform additional tasks. If the old resource that the generic server was managing is located under the old WebSphere Application Server installation, perform the following tasks:

1. Copy any related files to the new installation.
2. Run any setup required to put the external application back into a valid and working state.

   It is best that you reinstall the resource into the new WebSphere Application Server directory. Whatever you choose to do, the final step is to reset the reference to the new location of the application.

If the old resource that the generic server was managing is not installed under the oldWebSphere Application Server installation, nothing further is required.

Migration of a Version 6.x or above node to a Version 8.0 node

You can migrate a WebSphere Application Server Version 6.x or above node that belongs to a cell without removing the node from the cell.

Migrate the deployment manager first, before migrating any base nodes in the cell.

Important: Use the same cell name when migrating WebSphere Application Server, Network Deployment from Version 6.x or above to Version 8.0. If you use a different cell name, federated nodes cannot successfully migrate to the WebSphere Application Server, Network Deployment Version 8.0 cell.

Migrating a base WebSphere Application Server node that is within a cell to Version 8.0 also migrates the node agent to Version 8.0. A cell can have some Version 8.0 nodes and other nodes that are at Version 6.x or above levels. Read “Coexistence support” on page 137 for information on restrictions on using mixed-release cells.

Policy files

WebSphere Application Server Version 8.0 migrates all the policy files that are installed with Version 6.x or above by merging settings into the Version 8.0 policy files with the following characteristics:

- Any comments located in the Version 8.0 policy files will be preserved. Any comments contained in the Version 6.x or above policy files will not be included in the Version 8.0 file.
- Migration will not attempt to merge permissions or grants; it is strictly an add-type migration. If the permission or grant is not located in the Version 8.0 file, the migration will bring it over.
- Security is a critical component; thus, the migration makes any additions at the end of the original `.policy` files right after the comment `MIGR0372I: Migrated grant permissions follow`. This is done to help administrators verify any policy-file changes that the migration has made.

Properties directories

Migration copies files from prior version directories into the WebSphere Application Server Version 8.0 configuration.

Property files

WebSphere Application Server Version 8.0 migrates all the property files that are installed with Version 6.x or above by merging settings into the Version 8.0 property files.

Resource adapter archives (RARs) referenced by J2C resources

RARs that are referenced by J2C resources are migrated if those RARs are in the old WebSphere Application Server installation. In this case, the RARs are copied over to the corresponding location in the new WebSphere Application Server installation. Relational Resource Adapter RARs will not be migrated.

Migrating cluster-level resources:

WebSphere Application Server Version 6.0 introduced the concept of cluster-level resources. These are configured in `resource`*`xxx`*`.xml` files under the cluster directories. For example:

```
<resources.j2c:J2CResourceAdapter xmi:id="J2CResourceAdapter_1112808424172"
  name="ims" archivePath="${WAS_INSTALL_ROOT}\installedConnectors\x2.rar">
  ...
</resources.j2c:J2CResourceAdapter>
```

If you have a cluster-level resource, this resource must be in the same location on each cluster member (node). Using the above example, therefore, each cluster member must have the RAR file installed at location `${WAS_INSTALL_ROOT}\installedConnectors\x2.rar`. `${WAS_INSTALL_ROOT}` is resolved on each cluster member to get the exact location.

In the migration of a deployment manager, the tools migrate the cluster files on the deployment manager, including the `resource`*`xxx`*`.xml` files.

In the migration of a federated node, the tools process each J2C adapter.

RAR files in a Version 6.x to Version 8.0 migration:

Migration from Version 6.x to Version 8.0 copies files such as RAR files from WAS_INSTALL_ROOT to WAS_INSTALL_ROOT and from USER_INSTALL_ROOT to USER_INSTALL_ROOT.

If you have a RAR file in the WAS_INSTALL_ROOT for Version 6.x, for example, the migration tools do not automatically copy the file from WAS_INSTALL_ROOT to USER_INSTALL_ROOT. This maintains the integrity of the cluster-level J2C resources. If you hardcoded a path to a RAR file (`archivePath="C:/WAS/installedConnectors/x2.rar"` for example) in Version 6.x, however, the Version 8.0 migration tools cannot change the archivePath attribute to reflect this because that would break all of the other cluster members that have not been migrated.

Samples

No migration of samples from previous versions is available. There are equivalent WebSphere Application ServerVersion 8.0 samples that you can install.

Security

Java 2 security is enabled by default when you enable security in WebSphere Application Server Version 8.0. Java 2 security requires you to grant security permissions explicitly.

When migrating to WebSphere Application ServerVersion 8.0, your choice of whether or not to migrate to support script compatibility results in one of two different outcomes.

- If you choose to migrate to support script compatibility, your security configuration is brought over to Version 8.0 without any changes.

  This is the default.
- If you choose not to migrate to support script compatibility, the security configuration is converted to the default configuration for WebSphere Application Server Version 8.0. The default security configuration for Version 6.1 and later acts almost the same as in the previous versions, but there are some changes.

  For example, existing keyfiles and trustfiles are moved out of the SSLConfig repertoire and new keystore and truststore objects are created.

  All SSL configuration repertoires of the System Secure Sockets Layer (SSSL) type, except those that belong to the daemon, are converted to the Java Secure Socket Extension (JSSE) type.

For more information on migrating your security configurations to Version 8.0, read the "Migrating, coexisting, and interoperating – Security considerations" article in the information center.

Stdin, stdout, stderr, passivation, and working directories

In WebSphere Application Server for z/OS, outputs for `stdin`, `stdout`, and `stderr` are directed to SYSOUT by default. If they are redirected to the configuration directory of a previous version, you might need to change this in the Version 8.0 JCL.

The migration tools attempt to migrate existing passivation and working directories. Otherwise, appropriate Version 8.0 defaults are used.

If WebSphere Application Server for z/OS user IDs have home directories in the configuration directory of a previous version, you should update them before migration to reside in another location.

Note: In a coexistence scenario, using common directories between versions can create problems.

Transport ports

The migration tools migrate all ports. You must resolve any port conflicts before you can run servers at the same time.

Note: If ports are already defined in a configuration being migrated, the migration tools fix the port conflicts in the Version 8.0 configuration and log the changes for your verification .

You must manually add virtual host alias entries for each port. For more information, read the "Configuring virtual hosts" article in the information center.

Web modules

The specification level of the Java Platform, Enterprise Edition (Java EE) implemented in WebSphere Application Server Version 6.0.x required behavior changes in the web container for setting the content type. If a default servlet writer does not set the content type, not only does the web container no longer default to it but the web container returns the call as "null." This situation might cause some browsers to display resulting web container tags incorrectly. To prevent this problem from occurring, migration sets the autoResponseEncoding IBM extension to "true" for web modules as it migrates enterprise applications.

JVM system properties

If you migrate a Version 6.1 configuration that has feature packs installed, the migration tools might add one or two JVM system properties for each Java server in your configuration, including your administrative servers. Web servers are not affected. The properties are set to indicate to the JVM that the configuration should use a Java annotation scan policy other than the Version 8.0 default scan policy.

- If you migrate a Version 6.1 profile that has the Feature Pack for EJB 3.0 installed, the migration tools add the following system property to the JVM definitions for all Java servers defined on that node:

  ```
  com.ibm.websphere.ejb.UseEJB61FEPScanPolicy = true
  ```

- If you migrate a Version 6.1 profile that has the Feature Pack for Web Services installed, the migration tools add the following system property to the JVM definitions for all Java servers defined on that node:

  ```
  com.ibm.websphere.webservices.UseWSFEP61ScanPolicy = true
  ```

- If you migrate a Version 6.1 profile that has both the Feature Pack for EJB 3.0 and the Feature Pack for Web Services installed, the migration tools add both of the system properties to the JVM definitions for all Java servers defined on that node:

  ```
  com.ibm.websphere.ejb.UseEJB61FEPScanPolicy = true
  com.ibm.websphere.webservices.UseWSFEP61ScanPolicy = true
  ```

  A WebSphere Application Server, Network Deployment configuration requires that the deployment manager profile be augmented with all of the feature packs used in the cell. This means that the deployment-manager profile can potentially have both feature packs installed even if none of its federated nodes have both installed.

If these properties are set, the following two changes take place in the default Version 8.0 behavior:

- Application installation generates classes based on the annotation scan policy associated with the settings for those two properties.

  This means that you can potentially use the following four annotation scan policies:

  – Version 8 default behavior
  – Feature Pack for EJB 3.0 behavior
  – Feature Pack for Web Services behavior
  – Net behavior from having both the Feature Pack for EJB 3.0 and the Feature Pack for Web Services installed

- The servers use the generated annotation classes based on the properties set, resulting in four potential behaviors.

You can change the scan policy behavior by adding or removing the custom JVM system properties from your `server.xml` files.

Important: To evoke the correct installApp behavior, the `server.xml` file for the deployment manager must retain any property specified for any node in the cell.

After changing the properties, you must reinstall or update your applications and then resynchronize the cell to implement the change.

Planning to migrate product configurations for z/OS

You must prepare your system to migrate from WebSphere Application Server for z/OS Version 6.x and Version 7.x to Version 8.0.

Before you begin

WebSphere Application Server for z/OS supports migration paths to Version 8.0 from the following releases:

- Version 6.x
- Version 7.x

In addition to migration, it is possible for these WebSphere Application Server versions to coexist with Version 8.0.

To learn more about the migration utilities, read “Overview of migration, coexistence, and interoperability” on page 1. When you are getting ready to migrate to Version 8.0, read “Premigration considerations” on page 6 and “Migrating z/OS stand-alone application servers” on page 38, “Migrating WebSphere Application Server, Network Deployment configurations on the z/OS operating system” on page 65, or “Migrating federated nodes for z/OS” on page 91 for planning and migration information.

Procedure

1. Read the "Task overview: Installing" article in the information center.

   This document will give you a basic understanding of what is involved with installing the product.
2. Install the Version 8.0 product.

   Read the following articles in the information center for details:
   - Planning for installation
   - Installing the product and additional software
3. To learn about preparing your environment for migration, read the "Preparing the base operating system" article in the information center.

   You should also plan for any new features that you might want to include. Use the instructions and tips:
   - For system requirements, read the "Hardware and software requirements on z/OS" article in the information center.
   - For server security (RACF), read the "Security planning overview" article in the information center.
   - For workload management (WLM), read the "Workload management (WLM) for z/OS" article in the information center.
4. Review the "Ensuring problem avoidance" article in the information center, making sure that each task has been completed before you begin.
5. Review “Migrating and coexisting application servers” on page 5 to determine whether you want the two versions of the product to coexist in the same environment or you want to perform a complete migration.
6. Capture the variables used in the current installation.

   This information will be useful in planning for the new installation, ensuring that you define unique ports, and planning for the configuration file system structure.

What to do next

When you are ready to begin migrating, go to Chapter 3, “Migrating product configurations,” on page 19.

Migrating product configurations with migration tools

Migration support consists of tools that are included with WebSphere Application Server. These tools primarily provide support for saving the configuration and applications from a previous version of the product into a migration-specific backup directory and then importing that configuration into the latest version of the application server.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6. For resources to help you plan and perform your migration, visit Knowledge Collection: Migration planning for WebSphere Application Server.

Important: Use the migration tools for the version of WebSphere Application Server that you are installing. The tools change over time. If you use migration tools from an earlier release of WebSphere Application Server, you are likely to encounter a problem with the migration.

The migration scripts are located in the *app_server_root*/bin directory after installation.

Procedure

Select the appropriate migration tools to migrate your product configurations.

z/OS Migration Management Tool

newfeat: You can use the z/OS Migration Management Tool to collect the information and generate the Job Control Language (JCL) jobs (CNTL and DATA datasets) for migrating a stand-alone application server, deployment manager, or federated node. You must create these jobs before you can begin the actual migration.

Read “Using the z/OS Migration Management Tool to create and manage migration definitions” on page 26 for more information.

zmmt command

newfeat: You can use the zmmt command with a response file to generate the JCL jobs (CNTL and DATA datasets) for migrating a stand-alone application server, deployment manager, or federated node. You must create these jobs before you can begin the actual migration.

Read “Using the zmmt command to create migration definitions on z/OS” on page 32 for more information.

clientUpgrade tool

You can use the clientUpgrade tool to upgrade the client application to a new release level.

Read “clientUpgrade command” on page 35 for more information.

convertScriptCompatibility tool

Administrators use the convertScriptCompatibility tool to convert their configuration from a mode that supports backward compatibility of Version 6.x or 7.x administration scripts to a mode that is fully Version 8.0.

Read “convertScriptCompatibility command” on page 35 for more information.

convertSelfSignedCertificatesToChained task

Chained certificates are the default certificate type in WebSphere Application Server Version 8.0. Administrators can use the convertSelfSignedCertificatesToChained task with the wsadmin tool to convert self-signed certificates to chained certificates.

Read the "SSLMigrationCommands command group for the AdminTask object" article for more information.

Tip: For help, read Chapter 11, “Troubleshooting migration,” on page 143.

What to do next

Use the selected tools to migrate your product configuration.

Using the z/OS Migration Management Tool to create and manage migration definitions

Before you migrate a WebSphere Application Server for z/OS node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you run during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the migration jobs.

Before you begin

1. Install WebSphere Application Server Version 8.0 on a z/OS system containing the earlier configuration.
2. Install the most current release of WebSphere Customization Toolbox Version 8.0 on a workstation running the Windows or Linux Intel operating system.
   - If the latest release of WebSphere Customization Toolbox Version 8.0 is on your product disk, install it.
   - If the latest release of WebSphere Customization Toolbox Version 8.0 is not on your product disk, perform the following tasks:
     a. Go to Recommended fixes for WebSphere Application Server.
     b. Look under **Version 8.0** for a link to the WebSphere Customization Toolbox Version 8.0 download page.
     c. Go to that web page, and download the latest WebSphere Customization Toolbox Version 8.0 package.
     d. **Optional:** Uninstall any earlier releases of WebSphere Customization Toolbox Version 8.0 from your system.

        Tip: You can have multiple copies of WebSphere Customization Toolbox concurrently installed. If you want to do this (to make sure that the new version is working before deleting the old version for example), the only requirement is that you install each version at a different location.
     e. Install the new WebSphere Customization Toolbox Version 8.0 package.
3. Make sure that an FTP server is up and running on the z/OS target system.

Note: The z/OS Migration Management Tool is intended for use by a systems programmer or administrator who is familiar with the z/OS target system on which the migrated Version 8.0 nodes run.

About this task

The z/OS Migration Management Tool itself does not perform the migration; it creates batch jobs and instructions that you can use to perform the actual migration tasks. The jobs and instructions that you

generate using the z/OS Migration Management Tool form a migration definition on your workstation that you then upload to your z/OS system and use for the migration.

In WebSphere Application Server for z/OS Version 8.0, you can use the z/OS Migration Management Tool and the jobs as well as instructions that it generates to migrate to Version 8.0 nodes. After you actually migrate a node, however, you must use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer it.

Tip: The z/OS Migration Management Tool creates a response file to hold the various values that you enter when you generate a WebSphere Application Server for z/OS migration definition. This response file remains on the workstation on which the z/OS Migration Management Tool is run, and you can use this file as a basis for creating similar migration definitions later. The z/OS Migration Management Tool allows you to put these response files on network drives where they can be shared with other users.

Procedure

1. Start the z/OS Migration Management Tool.

   Read “Starting the z/OS Migration Management Tool” on page 28 for more information.
2. Create your migration definitions.

   Read “Creating a migration definition using the z/OS Migration Management Tool” on page 28 for more information.
3. Optional: Modify the variables in your migration definitions.

   Read “Modifying a migration definition using the z/OS Migration Management Tool” on page 30 for more information.
4. Optional: Delete any existing migration definitions that you want to remove.
   a. In the WebSphere Application Server for z/OS migration definition table, select the migration definition that you want to delete.
   b. Click **Delete**.
   c. Click **Yes**.
5. Optional: Review your migration definitions.

   Read “Reviewing a migration definition using the z/OS Migration Management Tool” on page 31 for more information.
6. Upload your migration jobs to a target z/OS system, or export the jobs to the local file system.

   Read “Processing a migration definition using the z/OS Migration Management Tool” on page 31 for more information.

What to do next

1. To run the migration jobs on the target z/OS system, follow the instructions in the **Migration Instructions** view of the z/OS Migration Management Tool or in the migration jobs that you uploaded to the target z/OS system.

   Read the following articles for details:
   - “Migrating a stand-alone application server on the z/OS operating system” on page 62
   - “Migrating a z/OS deployment manager” on page 88
   - “Migrating a z/OS federated node” on page 108
2. After you migrate the nodes, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer them.

Starting the z/OS Migration Management Tool

Before you migrate a WebSphere Application Server for z/OS node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs.

Before you begin

1. Install WebSphere Application Server Version 8.0 on a z/OS system containing the earlier configuration.
2. Install the most current release of WebSphere Customization Toolbox Version 8.0 on a workstation running the Windows or Linux Intel operating system.
   - If the latest release of WebSphere Customization Toolbox Version 8.0 is on your product disk, install it.
   - If the latest release of WebSphere Customization Toolbox Version 8.0 is not on your product disk, perform the following tasks:
     a. Go to Recommended fixes for WebSphere Application Server.
     b. Look under **Version 8.0** for a link to the WebSphere Customization Toolbox Version 8.0 download page.
     c. Go to that web page, and download the latest WebSphere Customization Toolbox Version 8.0 package.
     d. **Optional:** Uninstall any earlier releases of WebSphere Customization Toolbox Version 8.0 from your system.

        Tip: You can have multiple copies of WebSphere Customization Toolbox concurrently installed. If you want to do this (to make sure that the new version is working before deleting the old version for example), the only requirement is that you install each version at a different location.
     e. Install the new WebSphere Customization Toolbox Version 8.0 package.

Procedure

1. Open the WebSphere Customization Toolbox.
   - On a Windows operating system, go to **Start > Programs > IBM WebSphere > WebSphere Customization Toolbox V8.0** and click **WebSphere Customization Toolbox**.
   - On a Linux operating system, use the menus used to start your programs.

     For example, click ***operating_system_menus_to_access_programs*** **> IBM WebSphere > WebSphere Customization Toolbox V8.0**.
2. If the z/OS Migration Management Tool is not already open, perform the following actions:
   a. Open the **Welcome** tab, and select **z/OS Migration Management Tool**.
   b. Read the Welcome information, and then click **Launch Selected Tool**.

What to do next

You can now create or work with a WebSphere Application Server for z/OS Version 8.0 migration definition.

Creating a migration definition using the z/OS Migration Management Tool

Before you migrate a WebSphere Application Server for z/OS Version 6.1.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create migration definitions and upload the appropriate migration jobs.

Before you begin

Print and complete one of the following worksheets:

- "Specifying a stand-alone application server migration configuration in the z/OS Migration Management Tool: Worksheet" on page 48
- "Specifying a deployment manager migration configuration in the z/OS Migration Management Tool: Worksheet" on page 76
- "Specifying a federated node migration configuration in the z/OS Migration Management Tool: Worksheet" on page 97

Refer to the definitions of the variables in the following articles:

- "Configuration variables for migrating a stand-alone application server using the z/OS Migration Management Tool" on page 42
- "Configuration variables for migrating a deployment manager using the z/OS Migration Management Tool" on page 69
- "Configuration variables for migrating a federated node using the z/OS Migration Management Tool" on page 93

Use the completed worksheet as a reference when you create your migration definition.

Procedure

1. Start the z/OS Migration Management Tool.

   Read "Starting the z/OS Migration Management Tool" on page 28 for more information.
2. To initially specify a location where you want migration definition files to be stored on your workstation or to add another migration location to the **Migration Locations** table, perform the following actions:
   a. Click **Add**.
   b. Enter the path name of the location where you want to store the migration data.

      Important: The migration location directory must be empty when you create a new migration location.
   c. Enter a name to be associated with the table entry.
   d. Select the version of WebSphere Application Server to which you are migrating.
   e. Click **Finish**.
3. In the **Migration Locations** table, select the location where you want to save the migration definition.
4. Click **Migrate**.
5. Complete the fields in the panels using the values that you entered for the variables on the configuration worksheet that you created, clicking **Back** and **Next** as necessary.

   Tips:

   - Hover your cursor over a field for help information.
   - Click **Cancel** at any time to leave the creation process without generating a migration definition.

   When you have successfully entered all of the necessary information on the panels for this type of migration definition, the z/OS Migration Management Tool displays the definition type, location, and name on the **Migration Summary** panel.
6. Click **Create** to build the migration definition on your workstation.
7. Read the information on the **Migration Creation Summary** panel.

   Tip: You might want to make a note of the name and response-file location for future reference.
8. Click **Finish**.

Modifying a migration definition using the z/OS Migration Management Tool

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. This article leads you through the steps involved in modifying a migration definition that you created using the z/OS Migration Management Tool.

Before you begin

Print and complete one of the following worksheets:

- "Specifying a stand-alone application server migration configuration in the z/OS Migration Management Tool: Worksheet" on page 48
- "Specifying a deployment manager migration configuration in the z/OS Migration Management Tool: Worksheet" on page 76
- "Specifying a federated node migration configuration in the z/OS Migration Management Tool: Worksheet" on page 97

Refer to the definitions of the variables in the following articles:

- "Configuration variables for migrating a stand-alone application server using the z/OS Migration Management Tool" on page 42
- "Configuration variables for migrating a deployment manager using the z/OS Migration Management Tool" on page 69
- "Configuration variables for migrating a federated node using the z/OS Migration Management Tool" on page 93

Use the completed worksheet as a reference when you modify your migration definition.

Procedure

1. Start the z/OS Migration Management Tool.

   Read "Starting the z/OS Migration Management Tool" on page 28 for more information.
2. In the **Migration Locations** table, select the location of the migration definition that you want to modify.
3. In the **Migration Definitions** table, select the migration definition that you want to modify.
4. Click **Regen**.
5. Modify the fields in the panels using the values that you entered for the variables on the configuration worksheet that you created.

   Tips:

   - Click **Back** and **Next** as necessary.
   - Hover your cursor over a field for help information.
   - Click **Cancel** at any time to leave the process without generating a modified migration definition.

   When you have successfully entered all of the necessary information on the panels for this type of migration definition, the z/OS Migration Management Tool displays the definition type, location, and name on the **Migration Summary** panel.
6. Click **Create** to build the modified migration definition on your workstation and replace the old definition.
7. Read the information on the **Migration Creation Summary** panel.
8. Click **Finish**.

Reviewing a migration definition using the z/OS Migration Management Tool

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. This article leads you through the steps involved in reviewing a migration definition that you created using the z/OS Migration Management Tool.

Before you begin

Procedure

1. Start the z/OS Migration Management Tool.

   Read “Starting the z/OS Migration Management Tool” on page 28 for more information.
2. In the **Migration Locations** table, select the location of the migration definition that you want to review.
3. In the **Migration Definitions** table, select the migration definition that you want to review.
4. Click the appropriate tabs for information about the migration definition.
   - Click the **Migration Summary** tab for general information about the migration definition.
   - Click the **Migration Instructions** tab for a copy of the customized instructions that were generated when the migration definition was created.

     These are the instructions that you use to perform the actual migration after you upload the migration definition to the z/OS target system.
   - Click the **Migration Response File** tab for a copy of the response file that was generated when the migration definition was created.
   - Click the **Migration Definitions** tab to return to a list of existing definitions, perform operations on these existing definitions, or create a new migration definition.

Processing a migration definition using the z/OS Migration Management Tool

Before you migrate a WebSphere Application Server for z/OS Version 6.1.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. This article leads you through the steps involved in uploading migration jobs to a target z/OS system using the z/OS Migration Management Tool. The article also describes how to export the jobs for a migration definition to a directory on the local file system using the z/OS Migration Management Tool.

Before you begin

Create the migration definition for which you want to upload jobs to a target z/OS system. Read “Creating a migration definition using the z/OS Migration Management Tool” on page 28 for more information.

Procedure

1. Start the z/OS Migration Management Tool.

   Read “Starting the z/OS Migration Management Tool” on page 28 for more information.
2. In the **Migration Locations** table, select the location of the migration definition that you want to process.
3. In the **Migration Definitions** table, select the migration definition that you want to process.
4. Click **Process**.
5. On the **Select Process Type** panel, select the type of processing that you want to perform on the migration definition.
6. Click **Next**.
7. Depending on the type of process that you selected, perform one of the following actions:
   - On the **Upload Migration Definition** panel, specify the necessary upload information.

a. In the **Target z/OS system** field, enter the IP name or address of the z/OS system to which you want to upload the migration jobs.
b. In the **User ID** field, enter the user ID that you want to use to log on to the FTP server on the target z/OS system.
c. In the **Password** field, enter the password for the user ID that you want to use to log on to the FTP server on the target z/OS system.
d. In the **Server port** field, enter the port number of the FTP server on the target z/OS system.
e. In the **Timeout** field, enter the number of seconds that can elapse without any I/O operation completing before the upload is stopped.
f. If you want to allocate the target z/OS datasets, check the box beside **Allocate target z/OS datasets** and complete the two fields that are activated.

Note: If you are uploading the migration jobs to a system different from the system on which you want to run the jobs, you must target a volume that is shared by those systems.
1) In the **Volume** field, enter the volume for the target datasets.
2) In the **Unit** field, enter the unit for the target datasets.

- On the **Export Migration Definition** panel, specify the directory to which you want to export migration jobs.

Tips:

- Hover your cursor over a field for help information.
- Click **Cancel** to leave the process without uploading or exporting the migration jobs.

8. Click **Finish**.

Using the zmmt command to create migration definitions on z/OS

Before you migrate a WebSphere Application Server Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run during the actual migration. You can use the zmmt command to create the appropriate jobs and instructions. This article provides information on how you can use the zmmt command in the migration process.

Before you begin

Install WebSphere Application Server Version 8.0 on a z/OS system containing a Version 6.1.x or Version 7.x configuration.

Note: The zmmt command is intended for use by a systems programmer or administrator who is familiar with the z/OS target system on which the migrated Version 8.0 nodes run.

About this task

Overview

You can use the zmmt command to create migration jobs for migrating WebSphere Application Server Version 6.1.x or Version 7.x nodes to Version 8.0. The zmmt command itself does not perform the migration; it creates batch jobs and instructions that you can then use to perform the actual migration tasks.

The zmmt command is an alternative to the z/OS Migration Management Tool. You might want to use this command rather than the tool to create migration jobs if you have no workstations with the WebSphere Customization Toolbox available or need to automate the generation of your migration jobs.

You run the zmmt command using a migration response file that you create manually.

Tip: Migrating a WebSphere Application Server application-serving environment requires a fair amount of planning and coordination. If you have not previously migrated an application server, you should first migrate a "practice" standalone application server using a sample response file.

Note: In WebSphere Application Server Version 8.0, you can use the z/OS Migration Management Tool or the zmmt command and the jobs that they generate to migrate from Version 6.1.x or Version 7.x to Version 8.0 nodes. After you actually migrate a node, however, you use the WebSphere Application Server Version 8.0 administrative console or scripting to administer it.

Location of the shell script

The `zmmt.sh` shell script is located in the *smpe_root*`/bin` directory.

Parameters

-responseFile

This parameter is used to specify the path to your response file.

This cannot be used with the -profilePath parameter.

-profilePath

This parameter is used to specify a fully qualified path name to an existing set of generated jobs.

This cannot be used with the -responseFile parameter.

-workspace

This required parameter is used to specify the Eclipse workspace directory.

This must match the path specified in the response file before the .metadata directory for the profilePath and templatePath variables.

-transfer

This optional parameter is used to copy generated jobs from a UNIX System Services (USS) file system to a pair of partitioned datasets.

The zmmt command first writes the migration jobs to a USS file system.

-allocate

This optional parameter is used to attempt to allocate the target datasets.

This parameter cannot be used without the -transfer parameter.

Datasets are determined by appending the values ".CNTL" and ".DATA" to the zTargetHLQ value for the profile containing the jobs that are being copied. This operation overwrites existing files of the same name in those datasets.

Examples

The following examples describe typical command lines with attributes for the zmmt command. In these examples, /*xxx* can be any directory to which the user invoking the `zmmt.sh` script has read and write access.

- `zmmt.sh -workspace /`*xxx*` -responseFile /`*xxx*`/ZAppSrvcmd.responseFile`

  This example generates the migration jobs in the location specified by the profilePath option in the response file.

- `zmmt.sh -workspace /`*xxx*` -responseFile /`*xxx*`/ZAppSrvcmd.responseFile -transfer -allocate`

  . This example performs the following actions:

  – Generates the migration jobs in the location specified by the profilePath option in the response file

  – Transfers the jobs from the file system to the CNTL and DATA datasets

– Allocates the target CNTL and DATA datasets using the high-level qualifier specified by the targetHLQ option in the response file

- `zmmt.sh -workspace /xxx -profilePath /xxx/ZAppSrvcmd -transfer -allocate`

  This example performs the following actions:

  – Transfers the generated jobs at the location specified by the -profilePath parameter to those datasets

  – Allocates the target CNTL and DATA datasets using the high-level qualifier specified by the targetHLQ option in the response file

  Note: This usage assumes that migration jobs have already been generated with a previous invocation of the `zmmt.sh` script.

- `zmmt.sh -workspace /xxx -transfer -responseFile /xxx/ZAppSrvcmd.responseFile`

  This example transfers the generated jobs from the location specified by the profilePath option in the response file to the generated CNTL and DATA datasets.

  Note: This usage assumes that the jobs have already been generated with a previous invocation of the `zmmt.sh` script and that the target CNTL and DATA datasets have already been allocated.

Procedure

1. Prepare the type of response file that you need.
   - To prepare a response file for creating jobs for migrating a standalone application server, see “Specifying a z/OS stand-alone application server migration configuration in the zmmt command response file: Worksheet” on page 58.
   - To prepare a response file for creating jobs for migrating a deployment manager, see “Specifying a z/OS federated node migration configuration in the zmmt command response file: Worksheet” on page 105.
   - To prepare a response file for creating jobs for migrating a federated node, see “Specifying a z/OS deployment manager migration configuration in the zmmt command response file: Worksheet” on page 86.
2. On your target z/OS system, run the `zmmt.sh` shell script using your prepared response file.

   The command creates the CNTL and DATA files needed to run the required jobs. The response file needs to be located in the UNIX (USS) file system.

Results

After you run the `zmmt.sh` script, you should see something similar to the following output:

```
smpe_root/bin:> zmmt.sh  /zmmt/workspace
-responseFile  /ZMigDmgr01.responseFile  -allocate  -transfer<

osgi> Customization definition successfully written to
  /zmt/workspace/.metadata/.plugins/com.ibm.ws390.mmt.config/profiles/ZMigDmgr01

Attempting to allocate dataset: ZMMT.CNTL
Allocation successful.
Attempting to allocate dataset: ZMMT.DATA
Allocation successful.
Copying CNTL files to ZMMT.CNTL...
Copy successful.
Copying DATA files to ZMMT.DATA...
Copy successful.
```

The migration jobs have been copied to the CNTL and DATA datasets. Follow the generated instructions provided in the CNTL dataset to perform the actual migration.

clientUpgrade command

Use the **clientUpgrade** command to migrate previous versions of client resources to Version 8 level resources.

Use the **clientUpgrade** command to migrate Version 6.x and Version 7 client resources to Version 8 level resources. In the process of migrating these resources, the `client-resources.xmi` file that is located in the client JAR files is migrated to the latest level. A backup of the `client-resources.xmi` file is also located in the client JAR file. If this command is not launched against the client EAR files before they are installed on Version 8, the client EAR files do not operate or install correctly.

Locate the command file in the *app_server_root*/bin directory.

```
clientUpgrade EAR_file [-clientJar client_jar ][-logFileLocation logFileLocation]
[-traceString trace_spec [-traceFile file_name ]]
```

Parameters

Supported arguments include the following:

EAR_file
Use this parameter to specify the fully qualified path to the EAR file that contains client JAR files to process.

-clientJar
Use this optional parameter to specify a JAR file for processing. If not specified, the program transforms all client JAR files in the EAR file.

-logFileLocation *log_file_location*
Use this optional parameter to specify an alternate location to store the log output.

-traceString *trace_spec* **-traceFile** *file_name*
Use these optional parameters to gather trace information for IBM Service personnel. Specify a trace_spec of "*=all=enabled" (with quotation marks) to gather all trace information.

The following example demonstrates correct syntax:

```
clientUpgrade EAR_file -clientJar ejbJarFile
```

convertScriptCompatibility command

The convertScriptCompatibility command is used by administrators to convert their configurations from a mode that supports backward compatibility of WebSphere Application Server Version 6.x or Version 7.0.x administration scripts to a mode that is fully in the Version 8.0 configuration model.

The scope of the configuration changes depend on the type of profile that is being processed.

- For stand-alone configurations, the default is to convert all servers owned by the node in that configuration.

  Use the -serverName parameter for more granular control.
- For WebSphere Application Server, Network Deployment configurations, the default behavior is to convert all nodes and all servers owned by those nodes.

  Use the -nodeName and -serverName parameters for more granular control.

Nodes are checked to verify that they are at a WebSphere Application Server Version 8.0 level before they are processed in order to support mixed-node configurations. Client environments are not processed.

The following conversions take place with this tool:

- transports to channels

Existing transport entries in the configuration from WebSphere Application Server Version 6.x are mapped to channel support. This affects `server.xml` and `serverindex.xml` files. The values of the transport settings are used to create new channel entries.

- SSL configuration

  WebSphere Application Server Version 8.0 contains enhancements to SSL configuration that result in refactoring the existing SSL configuration model. Both the old and the new model are supported. The default is to map to the WebSphere Application Server Version 6.x or Version 7.x SSL configuration model.

- `bootstrapAddress` to `bootstrapAddresses`

  Each single bootstrap address configuration is converted to a new bootstrap address list configuration containing that single bootstrap address.

- `ObjectRequestBroker` from not using the server thread pool to using it

  For example, `<ObjectRequestBroker useServerThreadPool="false"...>` is changed to `<ObjectRequestBroker useServerThreadPool="true">`.

Location

The convertScriptCompatibility command is located in the following directory.

- *profile_root*/bin

Syntax

The syntax is as follows:

```
convertScriptCompatibility.sh  [-help]

convertScriptCompatibility.sh  [-backupConfig true | false]
                    [-nodeName node_name [-serverName server_name]]
                    [-traceString trace_spec [-traceFile file_name]]
```

Parameters

Supported arguments include the following parameters:

-help
This displays help for this command

-backupConfig
This is an optional parameter that is used to back up the existing configuration of the current profile. The default is true—that is, to use the backupConfig command to save a copy of the current configuration into the *profile_name*/temp directory.

Use the restoreConfig command to restore that configuration as required.

Read the "restoreConfig command" article in the information center for more information.

-nodeName
This is an optional parameter that is used to specify a particular node name be processed rather than every node in the configuration. If this is not specified, all nodes in the configuration are converted.

-serverName
This is an optional parameter that is used to specify a particular server name to be processed rather than every server in the configuration. It can be used on all profile types and can be used in conjunction with the -nodeName parameter when processing WebSphere Application Server, Network Deployment configurations. If this parameter is not specified, all servers in the configuration are converted. If it is used in conjunction with the -nodeName parameter, all processing is limited to the specified node name.

-traceString
This is an optional parameter. The value *trace_spec* specifies the trace information that you want to collect. To gather all trace information, specify "*=all=enabled" (with quotation marks). The default is to not gather trace information. If you specify this parameter, you must also specify the -traceFile parameter.

-traceFile
This is an optional parameter. The value *file_name* specifies the name of the output file for trace information. If you specify the -traceString parameter but do not specify the -traceFile parameter, the command does not generate a trace file.

Usage

Stand-alone application server profile

Example scenario

1. Choose to migrate to support script compatibility when you migrate the stand-alone application server.
2. Follow these steps to convert all servers under this stand-alone profile:
   a. Open a command window.
   b. Change to the Version 8.0 profile's *profile_root*/bin directory.
   c. Run the following command:
      convertScriptCompatibility.sh

Deployment manager with federated nodes

Example scenario 1

1. Choose to migrate to support script compatibility when you migrate the deployment manager and all of its federated nodes.
2. Follow these steps to convert all nodes and servers in this cell.

   Important: The following steps should be taken against the deployment-manager management profile. If you run the convertScriptCompatibility command against a federated profile, the changes will be removed the next time the deployment manager synchronizes with the federated node.

   a. Open a command window.
   b. Change to the Version 8.0 deployment-manager management profile's *profile_root*/bin directory.
   c. Run the following command:
      convertScriptCompatibility.sh
3. Perform a full resynchronization of the deployment manager's configuration with each federated node to produce a consistent configuration.

Example scenario 2

1. Choose **not** to migrate to support script compatibility when you migrate the deployment manager.
2. Choose to migrate to support script compatibility when you migrate the deployment manager's federated nodes.
3. Follow these steps to convert all non-converted nodes and servers in the cell.

   Important: The following steps should be taken against the deployment-manager management profile. If you run the convertScriptCompatibility command against a federated profile, the changes will be removed the next time the deployment manager synchronizes with the federated node.

   a. Open a command window.

b. Change to the Version 8.0 deployment-manager management profile's *profile_root*/bin directory.

c. For each federated node that has not been converted, run the following command:

```
convertScriptCompatibility.sh –nodeName ${non_converted_nodename}
```

4. Perform a full resynchronization of the deployment manager's configuration with each federated node to produce a consistent configuration.

For more information about where to run this command, read the "Using command line tools" article in the information center.

Migrating z/OS stand-alone application servers

Use the available planning information, information on how to generate the migration jobs and instructions, high-level migration checklist, and detailed migration explanation when you migrate stand-alone application servers to WebSphere Application Server Version 8.0.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6. For resources to help you plan and perform your migration, visit Knowledge Collection: Migration planning for WebSphere Application Server.

About this task

Complete this task to migrate a WebSphere Application Server for z/OS Version 6.x or 7.x stand-alone application server to a Version 8.0 stand-alone application server.

Tip: For help, read Chapter 11, “Troubleshooting migration,” on page 143.

Procedure

1. Print the “Migrating a z/OS stand-alone application server: Checklist” on page 60, and use it as you conduct your migration.
2. Inventory your applications and capture key information as described in “Preparing to migrate a z/OS stand-alone application server” on page 39.
3. Generate a set of migration jobs and instructions.

   Read “Preparing to migrate a stand-alone application server using the z/OS Migration Management Tool” on page 41 and “Processing a migration definition using the z/OS Migration Management Tool” on page 31 for information on generating and uploading a migration definition using the z/OS Migration Management Tool.

   Read “Using the zmmt command to create migration definitions on z/OS” on page 32 for information on generating migration definitions using the zmmt command.
4. Follow the customized instructions in the BBOMBINS member of the CNTL dataset that you generated, and refer to the information in “Migrating a stand-alone application server on the z/OS operating system” on page 62.

What to do next

After you migrate the stand-alone application server, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer the migrated application server.

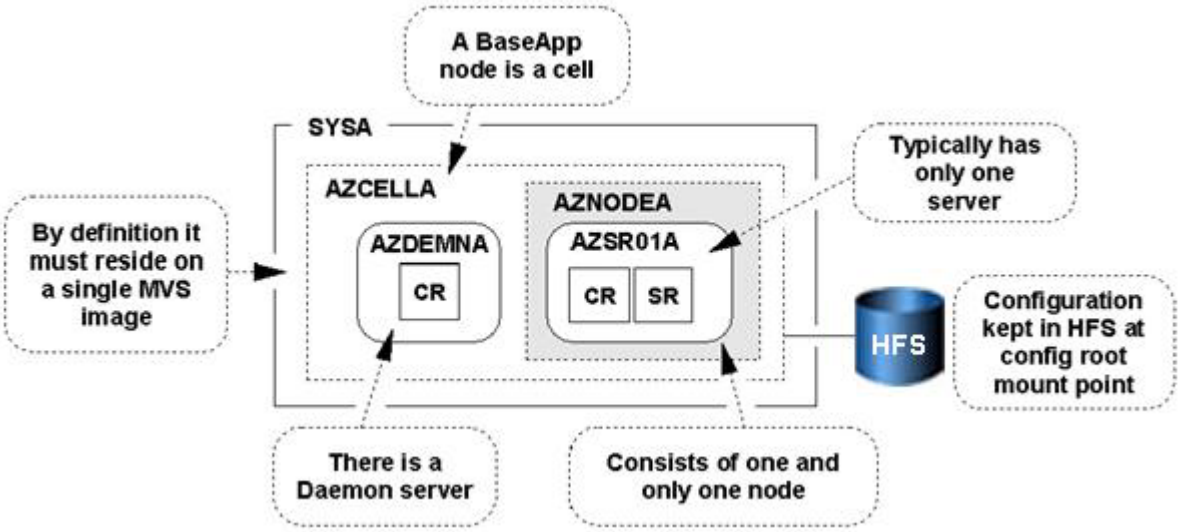
Preparing to migrate a z/OS stand-alone application server

Preparing to migrate a stand-alone application server node to WebSphere Application Server for z/OS Version 8.0 includes activities related to inventorying your existing configuration as well as generating CNTL and DATA datasets that you will use in the actual migration.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6.

Migrating a stand-alone application server node is relatively simple because it consists of a single server and a daemon.



You might have configured multiple application servers inside a stand-alone application server node. Generally, this configuration is not recommended; but if you have this configuration, the migration utilities will migrate all servers in the node just as it would all the servers in a Network Deployment application server node.

WebSphere Application Server for z/OS Version 8.0 supplies migration jobs in the target dataset that you allocate when you generate your migration jobs. Use these jobs to perform the actual migration.

- BBOWMG1B
- BBOWMG2B
- BBOMBHFS or BBOMBZFS
- BBOWMG3B
- BBOWBPRO
- BBOWBPRE
- BBOWBPOS
- BBOMBCP

For help, read Chapter 11, “Troubleshooting migration,” on page 143.

Procedure

1. Inventory your configurations.

Print this procedure, and enter the values for your configurations in the appropriate columns of the following tables.

- WebSphere Application Server for z/OS Version 6.x or Version 7.x configuration general information:

Table 3. Configuration general information. Enter your values:

Item	Value
Cell short name:	
Cell long name:	
Sysplex name:	
Location of original Version 6.x or Version 7.x configuration datasets:	
WebSphere Application Server Version 6.x or Version 7.x home directory (configuration file system):	

- WebSphere Application Server for z/OS Version 6.x or Version 7.x stand-alone application server node:

Table 4. Server node information. Enter your values:

Item	Value
Configuration mount point:	
Configuration file system dataset:	
Home directory:	Default: `/AppServer` Your value: _____
WebSphere Application Server SMP/E home:	Default: `/usr/lpp/zWebSphere/V5R1M2` Your value: _____
Controller Job Control Language (JCL) start procedure:	
Servant JCL start procedure:	
Daemon JCL start procedure:	

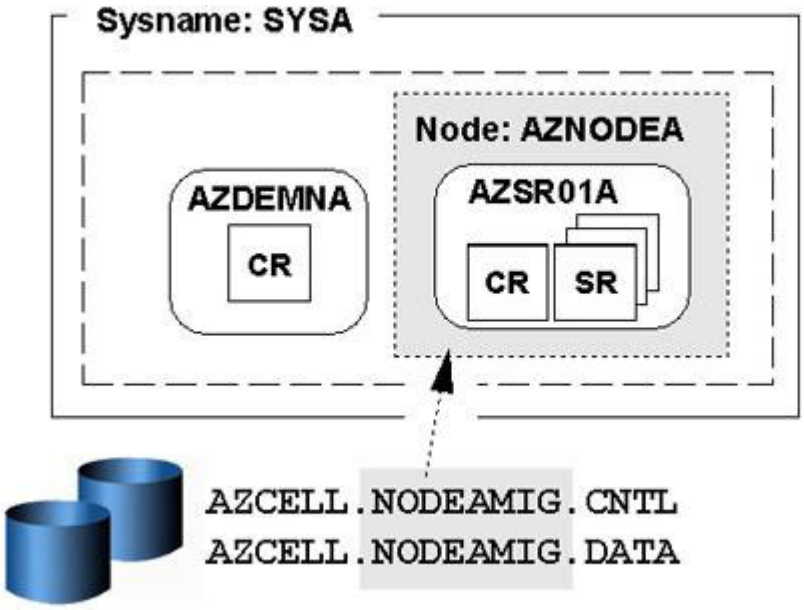
- WebSphere Application Server for z/OS Version 8.0 installation information:

Table 5. Installation information. Enter your values:

Item	Value
Product dataset HLQ:	
Product directory (SMP/E target for Version 8.0):	
WebSphere Application Server Version 8.0 in LPA/LNKLST?:	__ No __ Yes

2. Produce migration CNTL and DATA datasets for the node.

   You must provide a set of CNTL and DATA datasets for the migration utilities to read in order to perform the migration. Generate a set of customized jobs for the node in your configuration that you want to migrate.



Some important points about these steps:

- You submit these customized jobs
- When you go through the panels to generate these customized datasets, you must supply information related to your configuration on most of the panels. This information is used to generate your customized jobs.

  For a description of what you must do in the z/OS Migration Management Tool to generate your customized CNTL and DATA datasets, read "Preparing to migrate a stand-alone application server using the z/OS Migration Management Tool."

  For a description of what you must do to use the zmmt command to generate your customized CNTL and DATA datasets, read "Using the zmmt command to create migration definitions on z/OS" on page 32.

The purpose of creating these customized CNTL and DATA datasets is to give the migration utilities access to key information about your new Version 8.0 configuration as well as access to shell scripts customized with this new information. The migration utilities get the bulk of the configuration information from your existing Version 6.x or Version 7.x configuration file system.

Preparing to migrate a stand-alone application server using the z/OS Migration Management Tool

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload jobs for migrating a stand-alone application server.

Before you begin

Print and complete the "Specifying a stand-alone application server migration configuration in the z/OS Migration Management Tool: Worksheet" on page 48.

Procedure

1. Follow the instructions in “Creating a migration definition using the z/OS Migration Management Tool” on page 28.
   - On the **Migration Node Type Selection** panel, select **z/OS migrate a stand-alone application server**.
   - Fill in the fields on the panels using the values from your “Specifying a stand-alone application server migration configuration in the z/OS Migration Management Tool: Worksheet” on page 48.
2. Review the stand-alone application server migration definition to make sure that all the values are correct.
   a. In the **WebSphere Application Server for z/OS migration definition** table, select the migration definition that you want to review.
   b. Click **View**.
   c. Click the **Summary**, **Instructions**, or **Response file** tab for information about the migration definition.
3. Upload the migration jobs to the target z/OS system.

   Read “Processing a migration definition using the z/OS Migration Management Tool” on page 31 for instructions.

What to do next

1. To run the migration jobs on the target z/OS system, follow the instructions in the **Migration Instructions** view of the z/OS Migration Management Tool or in the BBOMBINS member of the CNTL dataset that you uploaded to the target z/OS system.

   Read “Migrating a stand-alone application server on the z/OS operating system” on page 62 for details.
2. After you migrate the stand-alone application server, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer it.

Configuration variables for migrating a stand-alone application server using the z/OS Migration Management Tool:

Before you migrate a WebSphere Application Server for z/OS Version 6.x or 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. The z/OS Migration Management Tool presents you with a set of configuration variables when creating a definition for migrating a stand-alone application server.

Migration Node Type Selection

Migration node type
Type of WebSphere Application Server node to migrate

Migration Definition Name

This section identifies the migration definition name and directory path to contain the batch jobs and instructions that will be used to migrate a WebSphere Application Server for z/OS node.

Migration definition name
Name of the z/OS migration definition

This name is used solely on the workstation to identify the migration jobs and instructions that are generated. The name chosen will have no effect on the WebSphere Application Server for z/OS configuration.

Response file path name (optional)
Full path name of a response file that contains default values to be preloaded in the tool

A response file is written each time a z/OS migration definition is created. It contains all of the variable data that was used to create the migration definition, and it can be used to preload the default values when defining a similar migration definition. The response file for a given migration definition is written to the *migration_definition_name*.responseFile file within the root directory for the migration definition.

Normally, you should specify a response file from a migration definition of the same type as that which you are about to define.

Note: A copy of the response file is included in the .DATA dataset that is included in the migration definition that you upload to a z/OS target system. This response file is not used on the z/OS system, but it is there for reference. The member name of the dataset is ZMMTBASE.

Target Datasets

High-level qualifier (HLQ)

High-level qualifier for the target z/OS datasets that will contain the generated jobs and instructions

When a z/OS migration definition is uploaded to the target z/OS system, the migration jobs and files are written to a pair of partitioned datasets. While is it possible to reuse these datasets, it is safest to create separate datasets for each z/OS system that is to be migrated.

- **HLQ.CNTL** - a partitioned dataset with fixed block 80-byte records to contain migration jobs
- **HLQ.DATA** - a partitioned dataset with variable length data to contain other data contained in the migration definition

Note: A multilevel high-level qualifier can be specified as the dataset high-level qualifier.

Dataset Names and Product Directory

JCL procedure library dataset name

Existing procedure library to which the WebSphere Application Server for z/OS cataloged procedures are to be copied

WebSphere Application Server product directory

Location of your WebSphere Application Server Version 8.0 installed product file system

Intermediate symbolic link

Select this option to allow to set up an intermediate symbolic link, and specify the path name of that link if you select it.

If you specify an intermediate symbolic link, symbolic links are created from the configuration file system to the intermediate symbolic link; otherwise, they are created directly to the product file system.

Selecting this option will allow you to specify the path name of an intermediate symbolic link. This link will be created by the customization jobs, pointing to the product file system directory.

Path name of intermediate symbolic link

Path name of intermediate symbolic link

Configuration File System

The configuration file system is where the configuration for the node being migrated is physically stored. You can choose to use an existing Version 8.0 file system if you already have an appropriate file system on the node being migrated. If you choose to use an existing Version 8.0 file system, you need to ensure that the mount point that you specify here is present before running the migration utilities (BBOWMG1B, BBOWMG2B, and so on) that are created using this tool. If you choose to create a new Version 8.0 file

system on the node being migrated, the actual creation of the new file system will not occur until you run the optional BBOMBHFS or BBOMBZFS job during the actual migration process. In any case, you must specify the correct value here for the mount point.

Refer to the customized instructions generated by this tool for specific information on setting the correct ownership and permissions on the configuration mount point. Read the generated instructions and “Migrating a stand-alone application server on the z/OS operating system” on page 62 for more information on specifying these variables.

Mount point
Read/write file system directory mount point where application data and environment files are written

If this mount point does not already exist, the migration process creates it when you run the optional BBOMBHFS or BBOMBZFS job.

Name File system dataset that you will create and mount at the above mount point

Volume, or '*' for SMS
Specify either the DASD volume serial number to contain the above dataset or "*" to let SMS select a volume.

Using "*" requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle dataset allocation automatically, list the volume explicitly.

Primary allocation in cylinders
Initial size allocation in cylinders for the configuration file system dataset

In an application server, the total space needed for this dataset increases with the size and number of installed applications.

Recommendation: The minimum suggested size is 420 cylinders.

Secondary allocation in cylinders
Size of each secondary extent in cylinders

Recommendation: The minimum suggested size is 100 cylinders.

File system type

Hierarchical File System (HFS)
Allocate and mount your configuration file system dataset using HFS

zSeries® File System (ZFS)
Allocate and mount your configuration file system dataset using ZFS

Server Customization (Part 1)

From configuration location

Mount point
Mount point of the configuration from which you are migrating

Home directory
Home directory of the configuration from which you are migrating

To configuration location

Mount point
Mount point of the configuration to which you are migrating

This was specified previously on the **Configuration File System** panel.

Home directory
Home directory of the configuration to which you are migrating

Daemon procedure name
Name of the JCL started procedure used to start the migrated daemon

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the daemon procedure or use the old one.

Controller procedure name
Name of the JCL started procedure used to start the migrated controllers

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the controller procedure or use the old one.

Servant procedure name
Name of the JCL started procedure used to start the migrated servants

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the servant procedure or use the old one.

Adjunct procedure name
Name of the JCL started procedure used to start the migrated adjunct

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the adjunct procedure or use the old one.

Replace started procedure command names
If you specified new names for your JCL procedures, then the corresponding START commands in the WebSphere Application Server configuration must be updated to match the new procedure names. Select this option to perform this configuration update.

If you choose to use the same procedure names, then do not select this option. If you are not using consistent procedure names for all the servers of a given process type (all servants for example) for the node that you are migrating, then it is recommended that you do not select this option. In this case, you will need to keep the same START commands and manually replace the procedures using the procedure that is generated during migration as a template.

Notes:

- Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x configuration. The migration process creates new Version 8.0 JCL procedures using the procedure names specified here.
- If you use the same names as you used in your Version 6.x configuration, the migration process overlays the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x procedures before running the migration jobs in case you need to roll back later.

Server Customization (Part 2)

Migrate to support script compatibility
Whether or not to migrate to support script compatibility

This specifies whether or not migration should create the following Version 6.x configuration definitions:

- Transport
- ProcessDef
- Version 6.x SSL

- Version 6.x ORB service threadpool

instead of the following Version 8.0 configuration definitions:

- Channels
- ProcessDefs
- Version 8.0 SSL
- Version 8.0 ORB service threadpool

Choose to migrate to support script compatibility if you want to minimize impacts to existing administration scripts. If you have existing wsadmin scripts or programs that use third-party configuration APIs to create or modify the Version 6.x configuration definitions, for example, you might want to choose this option.

Note: This is meant to provide a temporary transition until all of the nodes in the environment are at the Version 8.0 level. When they are all at the Version 8.0 level, you should perform the following actions:

1. Modify your administration scripts to use all of the Version 8.0 settings.
2. Use the convertScriptCompatability command to convert your configurations to match all of the Version 8.0 settings.

   Read “convertScriptCompatibility command” on page 35 for more information.

Application migration preference

How you would like to migrate your installed applications

Note: WebSphere Application Server system applications will migrate regardless of the value set here.

Migrate applications and use the default application installation directory

Install user enterprise applications in the default application installation directory as part of the migration

Migrate applications and use the specified application installation directory

Install user enterprise applications in a specified application installation directory as part of the migration

Application installation directory

Location where WebSphere Application Server installs your enterprise applications

This location is used when you specify that you want to migrate and install applications as your application migration preference. You have the option to choose a customized environment-specific location or use the default location.

Migrate and generate administrative scripts to install applications later

Prepare user enterprise applications for installation in the WebSphere Application Server Version 8.0 `installableApps` directory without actually installing them during migration

Scripts that can be used to install these applications are generated and saved in the migration backup directory. For WebSphere Application Server for z/OS, the location of this backup directory is relative to the temporary directory that you specify on this same panel. The location of the backup directory is also determined by the derived migration identifier and the type of node being migrated. If you specify `/tmp/migrate` as the temporary directory and the derived migration identifier is 55449, for example, then the location of the generated scripts is:

`/tmp/migrate/55449/`*`nodetype`*`_backup/`

where *nodetype* is dmgr, fed, or base depending on the type of node that you are migrating.

You can then run these files at any point and in any combination after migration has completed. You can also reorganize and combine these files for better application installation efficiency. Read the "Wsadmin tool" article in the information center for additional information.

Migrate applications and use the previous application installation directory
Install user enterprise applications as part of the migration, and keep the same application installation directories as the previous version

Restrictions: If you select this option, the location is shared by the existing WebSphere Application Server Version 6.x installation and the Version 8.0 installation. If you keep the migrated applications in the same locations as those of the previous version, the following restrictions apply:

- The WebSphere Application Server Version 8.0 mixed-node support limitations must be followed. This means that the following support cannot be used when evoking the wsadmin command:
  - Precompile JSP
  - Use Binary Configuration
  - Deploy EJB
- You risk losing the migrated applications unintentionally if you later delete applications from these locations when administering (uninstalling for example) your Version 6.x installation.
- Any application that is installed relative to a Version 6.x variable will be installed relative to the location assigned to that variable in Version 8.0. In other words, the absolute location is not preserved—the application is migrated to the relative location within the new Version 8.0 environment.

  If the binariesURL in the `deployment.xml` file for an application being migrated has a path that is relative to WebSphere Application Server—that is, it begins with `$(APP_INSTALL_ROOT)`, `$(WAS_INSTALL_ROOT)`, and so on—the new WebSphere Application Server variable value is used to resolve the path when the application is installed in the new location. This leads to the following results when you select this option:
  - Any application that is installed in a directory location relative to a WebSphere Application Server variable is installed under that variable value in Version 8.0.
  - Any application that is installed in a directory location that is not relative to a WebSphere Application Server variable is migrated and overwritten in that same directory. If an application is installed in the `/employee_records/retrieval_Apps` directory, for example, the application is migrated and overwritten in the `/employee_records/retrieval_Apps` directory.

Do not migrate applications
Do nothing with user enterprise applications

Migration trace options

If you choose to enable tracing, it will remain enabled throughout the entire migration process.

Enable script tracing
Enable or disable trace of the home creation, profile and migration tooling invocation, and final processing phases of migration

Enable profile creation tracing
Enable or disable trace trace during profile creation

Enable pre-upgrade tracing
Enable or disable trace during the WASPreUpgrade process

Enable post-upgrade tracing
Enable or disable trace during the WASPostUpgrade process

Migrate administrative console customized "My tasks" settings
"My tasks" is only supported in WebSphere Application Server Version 6.1 and later.

Migrate the settings for "My tasks" saved in the default workspace user root location (wstemp)

Migrate the settings for "My tasks" saved in a user defined workspace root location
User defined workspace root location

Temporary directory location
Directory where the backup of your previous configuration is written as well as the migration trace and temporary file output

During migration, a backup copy of the Version 6.x configuration is required. The default location of this backup is already specified, though you can override if needed. You might need to specify a location other than the default if the `/tmp` file system does not have adequate space to store the backup configuration. If you choose to override the default location of the backup copy, the best practice is to keep the same naming convention and just replace the `/tmp` portion with another path, `/myTemp/migrate` for example.

Migration definition identifier
Identifier that will be used to create a directory under the temporary directory that will contain the temporary migration datasets and backup configuration data

Job Statement Definition

All the migration jobs that will be tailored for you will need a job statement.

Enter a valid job statement for your installation. The migration creation process will update the job name for you in all the generated jobs, so you do not need to be concerned with that portion of the job statement. If continuation lines are needed, replace the comment lines with continuation lines.

Specifying a stand-alone application server migration configuration in the z/OS Migration Management Tool: Worksheet:

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. Print this worksheet, and use it when collecting and specifying information about the WebSphere Application Server for z/OS stand-alone application server migration configuration variables.

Before you begin

Procedure

1. Specify a value for the **Migration Node Type Selection** panel.

Table 6. Migration Node Type Selection. Enter your values:

Variable	Options	Your value
Migration node type	• z/OS migrate a stand-alone application server • z/OS migrate a deployment manager • z/OS migrate a federated node • z/OS migrate an administrative agent • z/OS migrate a job manager	z/OS migrate a stand-alone application server

2. Specify values for the **Migration Definition Name** panel.

Table 7. Migration Definition Name. Enter your values:

Variable	Default	Your value
Migration definition name	`ZMigSASrv`*`nn`*	
Response file path name (optional)	None	

3. Specify values for the **Target Datasets** panel.

Table 8. Target Datasets. Enter your values:

Variable	Default	Your value
High-level qualifier (HLQ)	None	

4. Specify values for the **Dataset Names and Product Directory** panel.

Table 9. Dataset Names and Product Directory. Enter your values:

Variable			Default	Your value
JCL procedure library dataset name			`SYS1.PROCLIB`	
WebSphere Application Server product directory			`/usr/lpp/zWebSphere/V8R0`	
Intermediate symbolic link				
	Create intermediate symbolic link		Selected	
		Path name of intermediate symbolic link	/wasv8config/*cell_long_name*/*node_long_name*/wasInstall	

5. Specify values for the **Configuration File System** panel.

Table 10. Configuration File System. Enter your values:

Variable	Default	Your value
Mount point	`/wasv8config`	

Table 10. Configuration File System (continued). Enter your values:

Variable		Default	Your value
Name		OMVS.WAS80. CONFIG.HFS	
Volume, or '*' for SMS		*	
Primary allocation in cylinders		420	
Secondary allocation in cylinders		100	
File system type			
	Hierarchical File System (HFS)	Selected	
	zSeries File System (ZFS)	Not selected	

6. Specify values for the **Server Customization (Part 1)** panel.

Table 11. Server Customization (Part 1). Enter your values:

Variable		Default	Your value
From configuration location			
	Mount point	/WebSphere/V6R0M0	
	Home directory	AppServer	
To configuration location			
	Mount point	Value from **Configuration File System** panel	
	Home directory	AppServer	
Daemon procedure name		BBO8DMN	
Controller procedure name		BBO8ACR	
Servant procedure name		BBO8ASR	
Adjunct procedure name		BBO8CRA	
Replace started procedure command names		Selected	

7. Specify values for the **Server Customization (Part 2)** panel.

Table 12. Server Customization (Part 2). Enter your values:

Variable	Default	Your value
Migrate to support script compatibility	Selected	
Application migration preference		

Table 12. Server Customization (Part 2) (continued). Enter your values:

Variable			Default	Your value
	Migrate applications and use the default application installation directory		Selected	
	Migrate applications and use the specified application installation directory		Not selected	
		Application installation directory	*/mount_point/home_directory*/profiles/default/installedApps	
	Migrate and generate administrative scripts to install applications later		Not selected	
	Migrate applications and use the previous application installation directory		Not selected	
	Do not migrate applications		Not selected	
Migration trace options				
	Enable script tracing		Not selected	
	Enable profile creation tracing		Not selected	
	Enable pre-upgrade tracing		Not selected	
	Enable post-upgrade tracing		Not selected	
Migrate administrative console customized "My tasks" settings				
	Migrate the settings for "My tasks" saved in the default workspace user root location (${USER_INSTALL_ROOT}/wstemp)		Selected	
	Migrate the settings for "My tasks" saved in a user defined workspace root location		Not selected	
		User defined workspace root location	None	
Temporary directory location			/tmp/migrate	
Migration definition identifier			Generated value	

8. Specify values for the **Job Statement Definition** panel.

Table 13. Job Statement Definition. Enter your values:

//jobname JOB	(ACCTNO,ROOM),'USERID',CLASS=A,REGION=OM
//*	
//*	
//*	

Preparing to migrate a stand-alone application server for z/OS using the zmmt command

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the zmmt command to create jobs for migrating a stand-alone application server.

Before you begin

Procedure

1. Print and complete the “Specifying a z/OS stand-alone application server migration configuration in the zmmt command response file: Worksheet” on page 58.
2. Follow the instructions in “Using the zmmt command to create migration definitions on z/OS” on page 32.

   Use the values from your “Specifying a stand-alone application server migration configuration in the z/OS Migration Management Tool: Worksheet” on page 48.

What to do next

1. Follow the instructions in the BBOMBINS member of the CNTL dataset that you generated using the zmmt command.

   Read “Migrating a stand-alone application server on the z/OS operating system” on page 62 for details.
2. After you migrate the stand-alone application server, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer it.

Configuration variables for migrating a z/OS stand-alone application server using the zmmt command:

Before you migrate WebSphere Application Server for z/OS, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run during the actual migration. You can use the zmmt command with a response file to create the appropriate migration jobs. The migration response file contains a set of configuration variables that you use to create jobs for migrating a stand-alone application server.

Response-file variables

create Required keyword to indicate the creation of a new migration definition

profileName
Name of the profile created during migration

This is always "default" on the z/OS platform.

profilePath
Fully qualified path where the generated migration definition output should be written

templatePath
Template path

One of the following values:

- zos-migDmgr
  Migrate a deployment manager
- zos-migFederated
  Migrate a federated node
- zos-migStandalone
  Migrate a stand-alone application server

zConfigHfsName
Name of the MVS dataset that will contain the configuration file system

This name is used as input to the migration job that creates the configuration file system.

zConfigHfsPrimaryCylinders
Number of primary cylinders that will be allocated to the configuration file system

In an application server, the total space needed for this dataset increases with the size and number of installed applications.

Recommendation: The minimum suggested size is 420 cylinders.

zConfigHfsSecondaryCylinders
Number of secondary cylinders that will be allocated to the configuration file system

Recommendation: The minimum suggested size is 100 cylinders.

zConfigHfsVolume
DASD volume serial number to contain the above dataset, or "*" to let SMS select a volume

Using "*" requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle dataset allocation automatically, list the volume explicitly.

zConfigMountPoint
File system directory mount point where application data and environment files are written

The configuration file system is where the configuration for the node being migrated is physically stored. You can choose to use an existing Version 8.0 file system if you already have an appropriate file system on the node being migrated. If you choose to use an existing Version 8.0 file system, you need to ensure that the mount point that you specify here is present before running the migration jobs that are created using this tool. If you choose to create a new Version 8.0 file system on the node being migrated, the actual creation of the new file system will not occur until you run the optional BBOMBHFS or BBOMBZFS job during the actual migration process. In any case, you must specify the correct value here for the mount point.

zFilesystemType
Type of file system

One of the following values:

- HFS

  Allocate and mount your configuration file system dataset using the Hierarchical File System
- ZFS

  Allocate and mount your configuration file system dataset using the zSeries File System

zJobStatement1 . . . *n*
Customizable JOB statements that will be used for the generated migration jobs

All the migration jobs that will be tailored for you will need a job statement. Enter a valid job statement for your installation. The migration creation process will update the job name for you in all the generated jobs, so you do not need to be concerned with that portion of the job statement. If continuation lines are needed, replace the comment lines with continuation lines.

zTargetHLQ
High-level qualifier for the target z/OS datasets that will contain the generated jobs and instructions

Note: A multilevel high-level qualifier can be specified as the dataset high-level qualifier.

zmbAppMigrationPreference
How you would like to migrate your installed applications

One of the following values:

- D

  Install user enterprise applications in the default application installation directory as part of the migration

- Y

  Install user enterprise applications in a directory specified in the following variables as part of the migration

  zmbAppInstallDirLine1
  zmbAppInstallDirLine2

  Location where WebSphere Application Server installs your enterprise applications

  This location is used when you specify that you want to migrate and install applications as your application migration preference. You have the option to choose a customized environment-specific location or use the default location.

  If the location path length exceeds 60 characters, then it must be specified on two lines as shown; but it cannot exceed a total length of 120 characters.

  Leave these two fields blank unless you specify Y for zmbAppMigrationPreference.

- S

  Prepare user enterprise applications for installation in the WebSphere Application Server Version 8.0 `installableApps` directory without actually installing them during migration

  Scripts that can be used to install these applications are generated and saved in the migration backup directory. For WebSphere Application Server for z/OS, the location of this backup directory is relative to the temporary directory that you specify for the zmbTempDirectory variable. The location of the backup directory is also determined by the derived migration identifier and the type of node being migrated. If you specify `/tmp/migrate` as the temporary directory and the derived migration identifier is 55449, for example, then the location of the generated scripts is:

  `/tmp/migrate/55449/`*`nodetype`*`_backup/`

  where *nodetype* is dmgr, fed, or base depending on the type of node that you are migrating.

  You can then run these files at any point and in any combination after migration has completed. You can also reorganize and combine these files for better application installation efficiency. Read the "Wsadmin tool" article in the information center for additional information.

- P

  Install user enterprise applications as part of the migration, and keep the same application installation directories as the previous version

  Restrictions: If you select this option, the location is shared by the existing WebSphere Application Server Version 6.x installation and the Version 8.0 installation. If you keep the migrated applications in the same locations as those of the previous version, the following restrictions apply:

  – The WebSphere Application Server Version 8.0 mixed-node support limitations must be followed. This means that the following support cannot be used when evoking the wsadmin command:
    - Precompile JSP
    - Use Binary Configuration
    - Deploy EJB
  – You risk losing the migrated applications unintentionally if you later delete applications from these locations when administering (uninstalling for example) your Version 6.x installation.

– Any application that is installed relative to a Version 6.x variable will be installed relative to the location assigned to that variable in Version 8.0. In other words, the absolute location is not preserved—the application is migrated to the relative location within the new Version 8.0 environment.

  If the binariesURL in the `deployment.xml` file for an application being migrated has a path that is relative to WebSphere Application Server—that is, it begins with `$(APP_INSTALL_ROOT)`, `$(WAS_INSTALL_ROOT)`, and so on—the new WebSphere Application Server variable value is used to resolve the path when the application is installed in the new location. This leads to the following results when you select this option:

  - Any application that is installed in a directory location relative to a WebSphere Application Server variable is installed under that variable value in Version 8.0.
  - Any application that is installed in a directory location that is not relative to a WebSphere Application Server variable is migrated and overwritten in that same directory. If an application is installed in the `/employee_records/retrieval_Apps` directory, for example, the application is migrated and overwritten in the `/employee_records/retrieval_Apps` directory.

• N

  Do nothing with user enterprise applications

Note: WebSphere Application Server system applications will migrate regardless of the value set here.

zmbControllerProcName

Name of the JCL started procedure used to start the migrated controllers

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the controller procedure or use the old one.

zmbServantProcName

Name of the JCL started procedure used to start the migrated servants

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the servant procedure or use the old one.

zmbDaemonProcName

Name of the JCL started procedure used to start the migrated daemon

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the daemon procedure or use the old one.

zmbAdjunctProcName

Name of the JCL started procedure used to start the migrated adjunct

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the adjunct procedure or use the old one.

zmbReplaceStartedProcedureNames

Whether to update the START commands in the configuration with the new names specified (true) or to preserve the same names (false).

If you specified new names for your JCL procedures, then the corresponding START commands in the WebSphere Application Server configuration must be updated to match the new procedure names. Specify true for this variable to perform this configuration update.

If you choose to use the same procedure names, then specify false for this variable. If you are not using consistent procedure names for all the servers of a given process type (all servants for example) for the node that you are migrating, then it is recommended that you specify false for this variable. In this case, you will need to keep the same START commands and manually replace the procedures using the procedure that is generated during migration as a template.

Notes:

- Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x configuration. The migration process creates new Version 8.0 JCL procedures using the procedure names specified here.
- If you use the same names as you used in your Version 6.x configuration, the migration process overlays the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x procedures before running the migration jobs in case you need to roll back later.
- Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x configuration. The migration process creates new Version 8.0 JCL procedures using the procedure names specified here.
- If you use the same names as you used in your Version 6.x configuration, the migration process overlays the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x procedures before running the migration jobs in case you need to roll back later.

zmbEnablePostUpgradeTrace
Enable (true) or disable (false) trace during the WASPostUpgrade process

zmbEnablePreUpgradeTrace
Enable (true) or disable (false) trace during the WASPreUpgrade process

zmbEnableProfileTrace
Enable (true) or disable (false) trace during profile creation

zmbEnableScriptingTrace
Enable (true) or disable (false) trace of the home creation, profile and migration tooling invocation, and final processing phases of migration

zmbEnableScriptCompatibility
Whether (true) or not (false) to migrate to support script compatibility

This specifies whether or not migration should create the following Version 6.x configuration definitions:

- Transport
- ProcessDef
- Version 6.x SSL
- Version 6.x ORB service threadpool

instead of the following Version 8.0 configuration definitions:

- Channels
- ProcessDefs
- Version 8.0 SSL
- Version 8.0 ORB service threadpool

Choose to migrate to support script compatibility if you want to minimize impacts to existing administration scripts. If you have existing wsadmin scripts or programs that use third-party configuration APIs to create or modify the Version 6.x configuration definitions, for example, you might want to specify true for this variable.

Note: This is meant to provide a temporary transition until all of the nodes in the environment are at the Version 8.0 level. When they are all at the Version 8.0 level, you should perform the following actions:

1. Modify your administration scripts to use all of the Version 8.0 settings.
2. Use the convertScriptCompatability command to convert your configurations to match all of the Version 8.0 settings.

   Read “convertScriptCompatibility command” on page 35 for more information.

zmbFromConfigRoot
Mount point of the configuration from which you are migrating

zmbFromWASHomeDir
Home directory of the configuration from which you are migrating

zmbProclibName
Existing procedure library to which the WebSphere Application Server for z/OS cataloged procedures are to be copied

zmbSMPEHome
Location of your WebSphere Application Server Version 8.0 SMPE installed product file system

zmbTempDirectory
Directory where the backup of your previous configuration is written as well as the migration trace and temporary file output

During migration, a backup copy of the Version 6.x configuration is required. The default location of this backup is already specified, though you can override if needed. You might need to specify a location other than the default if the `/tmp` file system does not have adequate space to store the backup configuration. If you choose to override the default location of the backup copy, the best practice is to keep the same naming convention and just replace the `/tmp` portion with another path, `/myTemp/migrate` for example.

zmbTimestamp
Identifier that will be used to create a directory under the temporary directory that will contain the temporary migration datasets and backup configuration data

zmbToConfigRoot
Mount point of the configuration to which you are migrating

This is the same value that is specified for the zConfigMountPoint variable.

zmbToWASHomeDir
Home directory of the configuration to which you are migrating

zmbWorkspaceRootPreference
Whether to migrate the administrative console customized "My tasks" settings saved in the default workspace user root location (D) or to migrate the settings saved in a user-defined workspace root location (U)

zmbUserWorkspaceRoot
User-defined workspace root location

intermediateSymlinkPreference
Whether to set up an intermediate symbolic link

If you specify an intermediate symbolic link, symbolic links are created from the configuration file system to the intermediate symbolic link; otherwise, they are created directly to the product file system.

IntermediateSymlink
Path name of intermediate symbolic link

This link will be created by the customization jobs, pointing to the product file system directory.

Sample of a response file for migrating a stand-alone application server

```
create
profileName=default
profilePath=/zmmt/workspace/.metadata/.plugins/com.ibm.ws390.mmt.config/profiles/ZMigSASrv01
templatePath=zos-migStandalone
zConfigHfsName=OMVS.WAS80.CONFIG.HFS
zConfigHfsPrimaryCylinders=420
zConfigHfsSecondaryCylinders=100
zConfigHfsVolume=*
zConfigMountPoint=/wasv8config
zFilesystemType=HFS
zJobStatement1=(ACCTNO,ROOM),'USERID',CLASS=A,REGION=0M
zJobStatement2=//*
zJobStatement3=//*
zJobStatement4=//*
zTargetHLQ=SAS
zmbAdjunctProcName=BBO7CRA
zmbAppInstallDirLine1=/wasv8config/AppServer/profiles/default/installedApps
zmbAppInstallDirLine2=
zmbAppMigrationPreference=D
zmbControllerProcName=BBO7ACR
zmbDaemonProcName=BBO7DMN
zmbEnablePostUpgradeTrace=true
zmbEnablePreUpgradeTrace=true
zmbEnableProfileTrace=true
zmbEnableScriptCompatibility=true
zmbEnableScriptingTrace=true
zmbFromConfigRoot=/WebSphere/V5R1M0
zmbFromWASHomeDir=AppServer
zmbInstallDefaultApp=true
zmbInstallSamples=true
zmbProclibName=SYS1.PROCLIB
zmbReplaceStartedProcedureNames=true
zmbSMPEHome=/usr/lpp/zWebSphere/Version 8.0
zmbServantProcName=BBO7ASR
zmbTempDirectory=/tmp/migrate
zmbTimestamp=30080203
zmbToConfigRoot=/wasv8config
zmbToWASHomeDir=AppServer
zmbWorkspaceRootPreference=D
intermediateSymlinkPreference=Y
IntermediateSymlink=/V80SMPE/symblink
```

Specifying a z/OS stand-alone application server migration configuration in the zmmt command response file: Worksheet:

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the zmmt command with a response file to create the appropriate migration jobs. Print this worksheet, and use it when collecting and specifying information about the WebSphere Application Server for z/OS stand-alone application server migration configuration variables in the zmmt command response file.

Before you begin

Procedure

1. Specify "create."
2. Specify values for the name and location.

Table 14. Name and location. Enter your values:

Variable	Response-file variable	Example value	Your value
Profile name	profileName	Default	
Profile directory	profilePath	/AppServer/profiles	
Template path	templatePath	zos-migStandalone	

3. Specify values for the target datasets.

Table 15. Target datasets. Enter your values:

Variable	Response-file variable	Example value	Your value
High-level qualifier (HLQ)	zTargetHLQ	SAS	

4. Specify values for the dataset names, product directory, and intermediate symbolic link.

Table 16. Dataset names, product directory, and intermediate symbolic link. Enter your values:

Variable			Response-file variable	Example value	Your value
JCL procedure library dataset name			zmbProclibName	SYS1.PROCLIB	
WebSphere Application Server product directory			zmbSMPEHome	/usr/lpp/zWebSphere/V7R0	
Intermediate symbolic link					
	Create intermediate symbolic link		intermediateSymlinkPreference	Y	
		Path name of intermediate symbolic link	IntermediateSymlink	/V70SMPE/symblink	

5. Specify values for the configuration file system.

Table 17. Configuration file system. Enter your values:

Variable		Response-file variable	Example value	Your value
Mount point		zConfigMountPoint	/wasv7config	
Name		zConfigHfsName	OMVS.WAS70. CONFIG.HFS	
Volume, or '*' for SMS		zConfigHfsVolume	*	
Primary allocation in cylinders		zConfigHfsPrimaryCylinders	420	
Secondary allocation in cylinders		zConfigHfsSecondaryCylinders	100	
File system type				
	Hierarchical File System (HFS)	zFilesystemType	HFS	
	zSeries File System (ZFS)			

6. Specify values for server customization.

Table 18. Server customization. Enter your values:

Variable		Response-file variable	Example value	Your value
From configuration location				
	Mount point	zmbFromConfigRoot	/WebSphere/V5R1M0	
	Home directory	zmbFromWASHomeDir	AppServer	
To configuration location				
	Mount point	zmbToConfigRoot	/wasv7config	
	Home directory	zmbToWASHomeDir	AppServer	
Daemon procedure name		zmbDaemonProcName	BBO7DMN	
Controller procedure name		zmbControllerProcName	BBO7ACR	
Servant procedure name		zmbServantProcName	BBO7ASR	
Adjunct procedure name		zmbAdjunctProcName	BBO7CRA	
Replace started procedure command names		zmbReplaceStartedProcedure Names	True	

Table 19. Server customization. Enter your values:

Variable			Response-file variable	Example value	Your value
Migrate to support script compatibility			zmbEnableScriptCompatibility	True	
Application migration preference					
	Migrate applications and use the default application installation directory (D)		zmbAppMigrationPreference	Y	
	Migrate and generate administrative scripts to install applications later (S)				
	Migrate applications and use the previous application installation directory (P)				
	Do not migrate applications (N)				
	Migrate applications and use the specified application installation directory (Y)				
		Application installation directory	zmbAppInstallDirLine1	/wasv7config/AppServer/ profiles/default/ installedApps	
			zmbAppInstallDirLine2	Apps	
Migration trace options					

Table 19. Server customization (continued). Enter your values:

Variable			Response-file variable	Example value	Your value
	Enable script tracing		zmbEnableScriptingTrace	False	
	Enable profile creation tracing		zmbEnableProfileTrace	False	
	Enable pre-upgrade tracing		zmbEnablePreUpgradeTrace	False	
	Enable post-upgrade tracing		zmbEnablePostUpgradeTrace	False	
Migrate administrative console customized "My tasks" settings					
	Migrate the settings for "My tasks" saved in the default workspace user root location (D)		zmbWorkspaceRoot Preference	D	
	Migrate the settings for "My tasks" saved in a user defined workspace root location (U)				
		User defined workspace root location	zmbUserWorkspaceRoot		
Temporary directory location			zmbTempDirectory	/tmp/migrate	
Migration definition identifier			zmbTimestamp	30080203	

7. Specify values for job statement definition.

Table 20. Job statement definition. Enter your values:

Variable	Response-file variable	Example value	Your value
//jobname JOB	zJobStatement1	(ACCTNO,ROOM),'USERID', CLASS=A,REGION=OM	
//*	zJobStatement2	//*	
//*	zJobStatement3	//*	
//*	zJobStatement4	//*	

Migrating a z/OS stand-alone application server: Checklist

Activities for migrating a stand-alone application server to WebSphere Application Server Version 8.0 after you generate the appropriate Job Control Language (JCL) jobs include activities related to general preparation, transaction-log clearance, and node migration. Use the following checklist as a quick reference to these activities. Follow the instructions that were generated by the z/OS Migration Management Tool or the zmmt command to complete the process of migrating your stand-alone application server to Version 8.0.

Before you begin

You can find more details about migrating a stand-alone application server in “Preparing to migrate a z/OS stand-alone application server” on page 39 and “Migrating a stand-alone application server on the z/OS operating system” on page 62.

About this task

Key similarities to migrating a deployment manager:

- You must still create CNTL and DATA datasets.
- You must still allocate and mount a new configuration file system to act as the target for the new Version 8.0 configuration, either manually or using the BBOMBHFS or BBOMBZFS job.
- You must still run the BBOWMG3B job to perform the migration.

Key differences from migrating a deployment manager:

- You must clear the transaction logs using the BBOWMG1B job; you do not have to complete this step for a deployment manager migration.
- You must disable Peer Restart and Recovery (PRR) mode using the BBOWMG2B job; you do not have to complete this step for a deployment manager migration.
- There is no deployment manager node to migrate.

Procedure

1. Prepare to migrate the application server.

Table 21. Application server checklist. Enter your values:

Check off	Item
_____	Make sure that you have a minimum level of Version 8.0 maintenance. Read “Overview of migration, coexistence, and interoperability” on page 1 for more information.
_____	Inventory your existing Version 5.1.x or Version 6.x configuration and capture key information. Read “Preparing to migrate a z/OS stand-alone application server” on page 39 for more information.
_____	Create CNTL and DATA datasets for each node to be migrated. Read “Preparing to migrate a stand-alone application server using the z/OS Migration Management Tool” on page 41 or “Preparing to migrate a stand-alone application server for z/OS using the zmmt command” on page 52 for more information.
_____	Create the Version 8.0 configuration mount point, allocate a configuration file system, and mount either manually or using the BBOMBHFS or BBOMBZFS job.
_____	Run the BBOMBCP job to copy the generated JCL procedures to your procedure library.

2. Clear the transaction logs for the application server node.

Table 22. Transaction logs checklist. Enter your values:

Check off	Item
_____	Stop the server in the stand-alone application server node.
_____	Run the BBOWMG1B job, and check for success.
_____	Restart the application server to clear the transaction logs.
_____	Run the BBOWMG2B job to disable PRR mode.

3. Migrate the application server.

Table 23. Application server checklist. Enter your values:

Check off	Item
_____	Ensure that all servers are stopped, including the daemon server.
_____	Run the migration jobs—including the BBOWMG3B job or the BBOWBPRO, BBOWBPRE, and BBOWBPOS jobs—by following the generated instructions; and check for success.
_____	Start the application server.

Migrating a stand-alone application server on the z/OS operating system

After you generate Job Control Language (JCL) jobs for migrating a stand-alone application server node to WebSphere Application Server for z/OS Version 8.0, you can perform the actual migration by running those jobs. When you generated your custom migration jobs, you also created customized instructions for preparing and running the migration jobs in the BBOMBINS members of the CNTL dataset that was used to generate your jobs. Follow these customized instructions to complete the process of migrating your stand-alone application server to Version 8.0.

Before you begin

- Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6.
- You will not be able to proceed if you did not generate the JCL migration jobs.
- The BBOWMG1B, BBOWMG2B, BBOWMG3B, BBOWBPRO, BBOWBPRE, and BBOWBPOS jobs referenced in these instructions must be submitted by a WebSphere Administrator User ID.

  All other jobs must be submitted by a user ID that has control over the file system.
- **Tip:** Before migrating a WebSphere Application Server Version 6.x or 7.x stand-alone application server, use the backupConfig command or your own preferred backup utility to back up your existing configuration if you want to be able to restore it to its previous state after migration. Read the "backupConfig command" article in the information center for more information. Make sure that you note the exact name and location of this backed-up configuration.

For help, read Chapter 11, “Troubleshooting migration,” on page 143.

Procedure

1. Create and mount a new Version 8.0 configuration file system.

   Before you perform the migration, Version 8.0 requires a file system to be present for your new configuration. You can run the BBOMBHFS or BBOMBZFS job to create and mount a new configuration file system, or you can mount one manually. Either way, you must have a file system for your Version 8.0 configuration created and mounted before you proceed. This configuration file system is the target of the migration; your Version 6.x or 7.x configuration file system is the source.

   The BBOMBHFS or BBOMBZFS job creates a mount point directory, allocates the configuration's file system, and mounts the file system at whatever value you specified for the mount point when you generated your migration jobs.

   Ensure that you allocated, created, and mounted your configuration file system datasets either manually or using the BBOMBHFS or BBOMBZFS job before you proceed. The mount point should be owned by the WebSphere Admin ID and have permissions of at least 755. The new configuration file system structures should be included in BPXPARM so that they will be mounted at the next IPL.
2. Copy your generated JCL procedures.

   The migration utility BBOMBCP copies the generated JCL procedures used to start the servers to the specified procedure library. Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x or 7.x configuration. This utility will update the new Version 8.0 configuration, substituting your new JCL names in place of the names that existed in your original Version 6.x or 7.x configuration.

   Caution: This utility copies the generated JCL to your procedure library. If you specified the same names as you used in your Version 6.x or 7.x configuration when you generated your migration jobs, this utility will overlay the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x or 7.x procedures before running this utility in case you need to roll back later.

   Submit the BBOMBCP job, and verify a return code of 0.

3. If you specified new procedure names, update your RACF STARTED profiles for the controller and daemon.

   The STARTED profile used by controller regions is based on the procedure name and JOBNAME. You must ensure that a STARTED profile will apply so that the proper identity will be assigned to the started task. If your Version 6.x or 7.x controller JCL procedure name is AZACR and you specified AZ6ACR for Version 7, for example, then you would need to create a STARTED profile for that new procedure name:

```
              new controller      same identity used in
                 JCL name         V6.x or V7.x configuration
                    |                      |
 RDEFINE STARTED AZ6ACR.* STDATA(USER(AZACRU) GROUP(AZCFG) TRACE(YES))
```

   Note:

   - Do not use a different user ID to start. There are other things tied to the user ID; and if you change the user ID, other changes would also be required.
   - If your original STARTED profile was generic, STARTED AZ*.* ... for example, you would not need to create a new STARTED profile.
   - Servant region STARTED profiles are based on JOBNAME, not procedure name. So there is no issue with the servant when you use a different procedure name.
   - Daemons and node agents are controllers, so using different procedure names for those implies a new STARTED profile.

4. Submit the BBOWMG1B and BBOWMG2B jobs.

   Note: If you are not using XA connectors, submitting the BBOWMG1B and BBOWMG2B jobs is optional. However, you should submit both jobs to ensure that your transaction logs are clear.

   The BBOWMG1B job enables all servers on the application server node being migrated to start in Peer Restart and Recovery (PRR) processing mode. PRR processing mode resolves any outstanding transactions, clears the transaction logs, and stops the server. The BBOWMG2B job disables PRR mode and returns all servers to normal operating state.

   Follow these steps to clear the XA transaction logs:

   a. Submit the BBOWMG1B job, and verify a return code of 0.
   b. Restart the application server, and wait for it to perform PRR processing and stop automatically.
   c. Submit the BBOWMG2B job, and verify a return code of 0.

5. Stop the Version 6.x or 7.x daemon and application server.

   The daemon must be at the highest version level of any of the servers that it manages on the same LPAR. It will be at the Version 8.0 level when the server is started.

   You must stop the Version 6.x or 7.x application server before proceeding.

6. Delete and redefine the log stream.

   Perform this step only if you previously configured the transaction XA partner log or compensation log on the Version 6.x or 7.x server to use a log stream.

   a. Make sure that the node is stopped.
   b. Delete and redefine the log stream.

      You can use the BBOLOGSD and BBOLOGSA jobs that were created during Version 6.x or 7.x customization if you configured the server initially to use the log stream.

      The following sample shows an example of such a job:

```
//RLSP1A  JOB 'xxxx,yyy,?','USERID',MSGCLASS=H,
//         CLASS=J,MSGLEVEL=(1,1),REGION=4M,NOTIFY=&SYSUID
//STEP1    EXEC PGM=IXCMIAPU
//STEPLIB  DD   DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
```

```
DATA TYPE(LOGR) REPORT(YES) /* Default to show output of job */
 DELETE LOGSTREAM NAME(P1ACEL6A.W51ASA2.D)
 DEFINE LOGSTREAM NAME(P1ACEL6A.W51ASA2.D)
        LOWOFFLOAD(20)
        HIGHOFFLOAD(79)
        STG_DUPLEX(YES)
        DUPLEXMODE(UNCOND)
        STG_DATACLAS(OPERLOG)
        STG_SIZE(5000)
        HLQ(Q10RRS)
        LS_SIZE(5000)
        LS_DATACLAS(OPERLOG)
        STRUCTNAME(WAS_LOGRLS)
/*
```

7. Perform one of the following actions:
   a. Submit the BBOWMG3B job.

      The BBOWMG3B job performs the physical migration of the Version 6.x or 7.x node to Version 8.0 based on the information that you supplied when you generated your migration jobs. Submit the BBOWMG3B job, verify that you are getting return codes of 0, and review the log files in the migration temporary directory on the file system. The migration temporary directory is *temporary_directory_location*/*nnnnn*, where *temporary_directory_location* is the directory specified for the temporary directory location (/tmp/migrate by default) and *nnnnn* is the numeric value generated for the migration identifier when you generated your migration jobs.
   b. Submit the following three jobs:
      1) Submit the BBOWBPRO job.

         BBOWBPRO creates the Websphere Application Server home and default profile.
      2) Submit the BBOWBPRE job.

         BBOWBPRE runs the migration pre-upgrade process.
      3) Submit the BBOWBPOS job.

         BBOWBPOS runs the migration post-upgrade and finish-up (change file permission) processes.
8. Start the new application server node.

   Use the existing commands that you would use to start a Version 6.x or 7.x application server, but replace the RACF STARTED procedure name with the value that you entered for the controller procedure name when you generated your migration jobs. This command starts the Version 8.0 application server. Wait until the server is finished initializing before proceeding.

   The following message is displayed on the console and in the job log of BBOS001:

   ```
   BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR z/OS CONTROL PROCESS BBOS001
   ```

   At this point, your migration to Version 8.0 is complete.

What to do next

After you verify a successful migration to Version 8.0 and are successfully running a migrated configuration, you should delete the following items:

- Everything in the source configuration's file system
- Everything in the target configuration's *temporary_directory_location*/*nnnnn* directory, where *temporary_directory_location* is the directory specified for the temporary directory location (/tmp/migrate by default) and *nnnnn* is the numeric value generated for the migration identifier when you created your migration jobs
- The bbomigrt2.sh file

Migrating WebSphere Application Server, Network Deployment configurations on the z/OS operating system

Use the available planning information, information on how to generate the migration jobs and instructions, high-level migration checklists, and detailed migration explanations when you migrate WebSphere Application Server, Network Deployment configurations to WebSphere Application Server Version 8.0.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6. For resources to help you plan and perform your migration, visit Knowledge Collection: Migration planning for WebSphere Application Server.

Procedure

1. Prepare to migrate your WebSphere Application Server, Network Deployment cell.

   Read “Preparing to migrate a WebSphere Application Server, Network Deployment cell for z/OS” for more information.
2. Migrate your deployment manager.

   Read “Migrating deployment managers for z/OS” on page 68 for more information.
3. Migrate your web sever configurations.

   Read Chapter 6, “Migrating web server configurations,” on page 131 for more information.
4. Migrate your federated nodes.

   Read “Migrating federated nodes for z/OS” on page 91 for more information.

Preparing to migrate a WebSphere Application Server, Network Deployment cell for z/OS

Before migrating a WebSphere Application Server, Network Deployment cell to WebSphere Application Server for z/OS Version 8.0 perform the following actions: inventory your existing configuration, plan your node-by-node strategy, and generate the CNTL and DATA datasets that you will use in the actual migration.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6.

WebSphere Application Server for z/OS Version 8.0 supplies migration jobs in the target dataset that you allocate when you generate your migration jobs. Use these jobs to perform the actual migration.

- BBOMDHFS or BBOMDZFS
- BBOWMG3D
- BBOWDPRO
- BBOWDPRE
- BBOWDPOS
- BBOMDCP

For help, read Chapter 11, “Troubleshooting migration,” on page 143.

Procedure

1. Inventory your configurations.

   Enter the values for your configurations on a printed copy of the following tables:

- WebSphere Application Server for z/OS Version 6.x or Version 7.x configuration general information:

Table 24. Configuration general information. Enter your values:

Item	Value
Cell short name:	
Cell long name:	
Sysplex name:	
Configuration file system strategy:	__ All nodes share one file system __ Unique file system per node
Job Control Language (JCL) start procedure strategy:	__ Use common JCL __ Unique JCL per server
Location of original Version 6.x or Version 7.x configuration datasets:	
WebSphere Application Server Version 6.x or Version 7.x home directory (configuration file system):	

You also need a WebSphere Application Server for z/OS administrator user ID and password.

- WebSphere Application Server for z/OS Version 6.x or Version 7.x deployment manager node:

Table 25. Configuration general information. Enter your values:

Item	Value
Configuration mount point:	
Configuration file system dataset:	
Home directory:	Default:`/DeploymentManager` Your value: _____
WebSphere Application Server SMP/E home:	Default: `/usr/lpp/zWebSphere/V6R1` Your value: _____
Controller JCL start procedure:	
Servant JCL start procedure:	
Daemon JCL start procedure:	

- WebSphere Application Server for z/OS Version 8.0 installation information:

Table 26. Installation information. Enter your values:

Item	Value
Product dataset HLQ:	
Product directory (SMP/E target for Version 8.0):	
WebSphere Application Server Version 8.0 in LPA/LNKLST?:	__ No __ Yes

2. Plan the node-by-node migration strategy.

   Map out your migration sequence on a printed copy of the following table:

Table 27. Migration sequence. Enter your values:

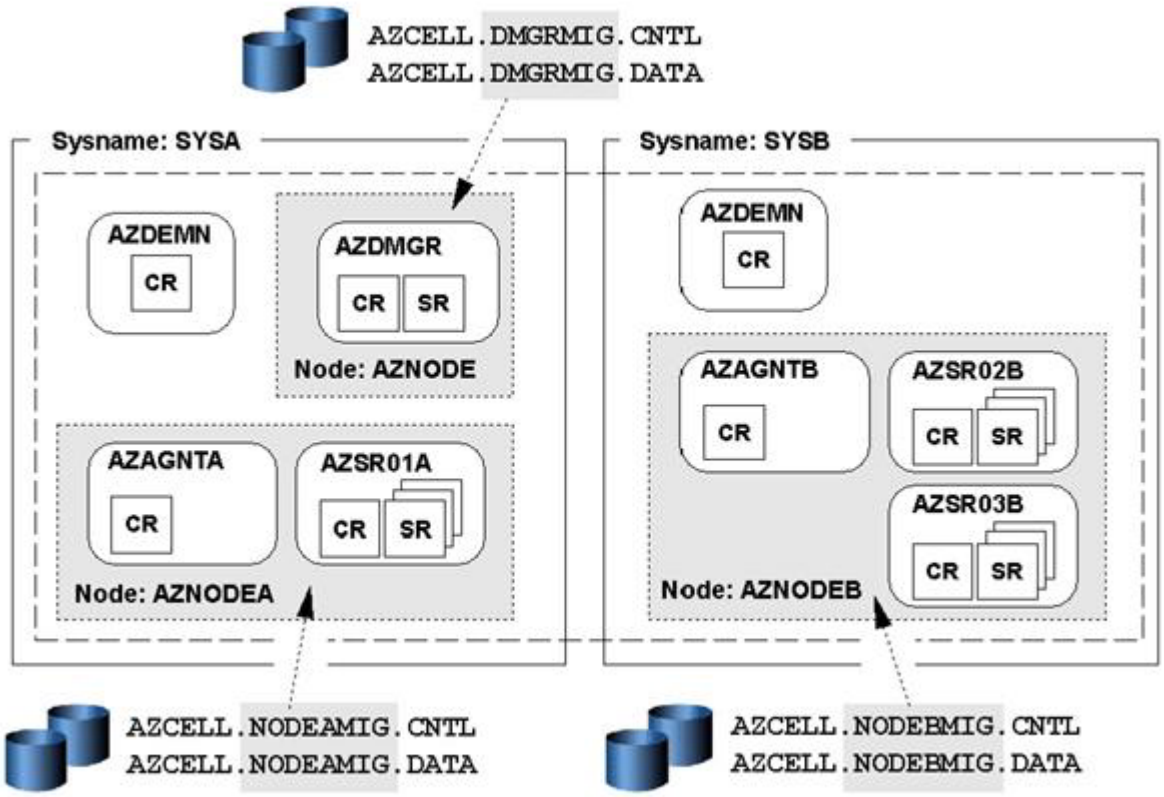
Item	Node Short Name	*System Name*
Deployment manager node		

Table 27. Migration sequence (continued). Enter your values:

Item	Node Short Name	*System Name*
Second node		
Third node		
Fourth node		
Fifth node		
Sixth node		

3. Produce migration CNTL and DATA datasets for each node that you want to migrate.

   You must provide a set of CNTL and DATA datasets for the migration utilities to read in order to perform the migration. Generate a set of customized jobs for each node in your configuration that you want to migrate.



Some important points about these steps:

- When you create these customized datasets, you must supply information related to your configuration on most of the panels. This information is used to generate your customized jobs.

  For a description of what you must do in the z/OS Migration Management Tool to generate your customized CNTL and DATA datasets, read “Preparing to migrate a deployment manager using the z/OS Migration Management Tool” on page 69 and “Preparing to migrate a federated node using the z/OS Migration Management Tool” on page 92.

  For a description of what you must do to use the zmmt command to generate your customized CNTL and DATA datasets, read “Using the zmmt command to create migration definitions on z/OS” on page 32.

- You submit these customized jobs

The purpose of creating these customized CNTL and DATA datasets is to give the migration utilities access to key information about your new Version 8.0 configuration as well as access to shell scripts

customized with this new information. The migration utilities get the bulk of the configuration information from your existing Version 6.x or Version 7.x configuration file system.

Migrating deployment managers for z/OS

Use the available planning information, information on how to generate the migration jobs and instructions, high-level migration checklist, and detailed migration explanation when you migrate deployment managers to WebSphere Application Server Version 8.0.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6.

About this task

Complete this task to migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x deployment manager to a Version 8.0 deployment manager.

Tip: For help, read Chapter 11, “Troubleshooting migration,” on page 143.

Procedure

1. Print the “Migrating a z/OS deployment manager: Checklist” on page 87, and use it as you conduct your migration.
2. Inventory your applications, and capture key information as described in “Preparing to migrate a WebSphere Application Server, Network Deployment cell for z/OS” on page 65.
3. Generate a set of migration jobs and instructions.

   Read “Preparing to migrate a deployment manager using the z/OS Migration Management Tool” on page 69 and “Processing a migration definition using the z/OS Migration Management Tool” on page 31 for information on generating and uploading your migration definition using the z/OS Migration Management Tool.

   Read “Using the zmmt command to create migration definitions on z/OS” on page 32 for information on generating migration definitions using the zmmt command.
4. Follow the customized instructions in the BBOMDINS member of the CNTL dataset that you generated and refer to the information in “Migrating a z/OS deployment manager” on page 88.

Results

After you migrate the deployment manager, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer the migrated deployment manager.

What to do next

If security is enabled in your Version 6.0.2 environment before migration, the LocalOSUserRegistry element in the `security.xml` file contains the `realm="customRealm"` value. This realm setting causes the following exception to occur in the controller region log for the previously migrated V6.0.2 deployment manager:

```
ExtendedMessage: BBOO0220E: HMGR0149E: An attempt to open a connection to
coregroup DefaultCoreGroup has been rejected
```

Complete the following steps to remove the customRealm value for the realm:

1. Access the *install_root*/*deployment_manager_profile*/`bin` directory.
2. Issue the following commands to remove the customRealm value for the realm.
   a. `wsadmin.sh -conntype SOAP -port` *port_name* `-user` *user_name* `-password` *password*

b. `set cell [$AdminConfig list Cell]`

c. `set security [$AdminConfig list Security $cell]`

d. `set localos [$AdminConfig list LocalOSUserRegistry $security]`

e. `$AdminConfig modify $localos [list {"realm" ""}]`

f. `$AdminConfig save`

After you complete these steps, realm value is removed. The error no longer occurs.

Preparing to migrate a deployment manager using the z/OS Migration Management Tool

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload jobs for migrating a deployment manager.

Before you begin

Print and complete the "Specifying a deployment manager migration configuration in the z/OS Migration Management Tool: Worksheet" on page 76.

Procedure

1. Follow the instructions in "Creating a migration definition using the z/OS Migration Management Tool" on page 28.
   - On the **Migration Node Type Selection** panel, select **z/OS migrate a deployment manager**.
   - Fill in the fields on the panels using the values from your "Specifying a deployment manager migration configuration in the z/OS Migration Management Tool: Worksheet" on page 76.
2. Review the deployment manager migration definition to make sure that all the values are correct.
   a. In the **WebSphere Application Server for z/OS migration definition** table, select the migration definition that you want to review.
   b. Click **View**.
   c. Click the **Summary**, **Instructions**, or **Response file** tab for information about the migration definition.
3. Upload the migration jobs to the target z/OS system.

   Read "Processing a migration definition using the z/OS Migration Management Tool" on page 31 for instructions.

What to do next

1. To run the migration jobs on the target z/OS system, follow the instructions in the **Migration Instructions** view of the z/OS Migration Management Tool or in the BBOMDINS member of the CNTL dataset that you uploaded to the target z/OS system.

   Read "Migrating a z/OS deployment manager" on page 88 for details.
2. After you migrate the deployment manager, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer it.

Configuration variables for migrating a deployment manager using the z/OS Migration Management Tool:

Before you migrate a WebSphere Application Server for z/OS, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. The z/OS Migration Management Tool presents you with a set of configuration variables when creating a definition for migrating a deployment manager.

Migration Node Type Selection

Migration node type
Type of WebSphere Application Server node to migrate

Migration Definition Name

This section identifies the migration definition name and directory path to contain the batch jobs and instructions that will be used to migrate a WebSphere Application Server for z/OS node.

Migration definition name
Name of the z/OS migration definition

This name is used solely on the workstation to identify the migration jobs and instructions that are generated. The name chosen will have no effect on the WebSphere Application Server for z/OS configuration.

Response file path name (optional)
Full path name of a response file that contains default values to be preloaded in the tool

A response file is written each time a z/OS migration definition is created. It contains all of the variable data that was used to create the migration definition, and it can be used to preload the default values when defining a similar migration definition. The response file for a given migration definition is written to the *migration_definition_name*.responseFile file within the root directory for the migration definition.

Normally, you should specify a response file from a migration definition of the same type as that which you are about to define.

Note: A copy of the response file is included in the .DATA dataset that is included in the migration definition that you upload to a z/OS target system. This response file is not used on the z/OS system, but it is there for reference. The member name of the dataset is ZMMTDMGR.

Target Datasets

High-level qualifier (HLQ)
High-level qualifier for the target z/OS datasets that will contain the generated jobs and instructions

When a z/OS migration definition is uploaded to the target z/OS system, the migration jobs and files are written to a pair of partitioned datasets. While is it possible to reuse these datasets, it is safest to create separate datasets for each z/OS system that is to be migrated.

- **HLQ.CNTL** - a partitioned dataset with fixed block 80-byte records to contain migration jobs
- **HLQ.DATA** - a partitioned dataset with variable length data to contain other data contained in the migration definition

Note: A multilevel high-level qualifier can be specified as the dataset high-level qualifier.

Dataset Names and Product Directory

JCL procedure library dataset name
Existing procedure library to which the WebSphere Application Server for z/OS cataloged procedures are to be copied

WebSphere Application Server product directory
Location of your WebSphere Application Server Version 8.0 installed product file system

Intermediate symbolic link
Select this option to allow to set up an intermediate symbolic link, and specify the path name of that link if you select it.

If you specify an intermediate symbolic link, symbolic links are created from the configuration file system to the intermediate symbolic link; otherwise, they are created directly to the product file system.

Selecting this option will allow you to specify the path name of an intermediate symbolic link. This link will be created by the customization jobs, pointing to the product file system directory.

Path name of intermediate symbolic link
Path name of intermediate symbolic link

Configuration File System

The configuration file system is where the configuration for the node being migrated is physically stored. You can choose to use an existing Version 8.0 file system if you already have an appropriate file system on the node being migrated. If you choose to use an existing Versions 8.0 file system, you need to ensure that the mount point that you specify here is present before running the migration utilities (BBOMDHFS or BBOMDZFS, BBOMDCP, and so on) that are created using this tool. If you choose to create a new Version 8.0 file system on the node being migrated, the actual creation of the new file system will not occur until you run the optional BBOMDHFS or BBOMDZFS job during the actual migration process. In any case, you must specify the correct value here for the mount point.

Refer to the customized instructions generated by this tool for specific information on setting the correct ownership and permissions on the configuration mount point. Read the generated instructions and “Migrating a z/OS deployment manager” on page 88 for more information on specifying these variables.

Mount point
Read/write file system directory mount point where application data and environment files are written

If this mount point does not already exist, the migration process creates it when you run the optional BBOMDHFS or BBOMDZFS job.

Name File system dataset that you will create and mount at the above mount point

Volume, or '*' for SMS
Specify either the DASD volume serial number to contain the above dataset or "*" to let SMS select a volume.

Using "*" requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle dataset allocation automatically, list the volume explicitly.

Primary allocation in cylinders
Initial size allocation in cylinders for the configuration file system dataset

In an application server, the total space needed for this dataset increases with the size and number of installed applications.

Recommendation: The minimum suggested size is 420 cylinders.

Secondary allocation in cylinders
Size of each secondary extent in cylinders

Recommendation: The minimum suggested size is 100 cylinders.

File system type

Hierarchical File System (HFS)
Allocate and mount your configuration file system using HFS

zSeries File System (ZFS)
Allocate and mount your configuration file system using ZFS

Server Customization (Part 1)

From configuration location

Mount point
Mount point of the configuration from which you are migrating

Home directory
Home directory of the configuration from which you are migrating

To configuration location

Mount point
Mount point of the configuration to which you are migrating

This was specified previously on the **Configuration File System** panel.

Home directory
Home directory of the configuration to which you are migrating

Daemon procedure name
Name of the JCL started procedure used to start the migrated daemon

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the daemon procedure or use the old one.

Controller procedure name
Name of the JCL started procedure used to start the migrated controllers

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the controller procedure or use the old one.

Servant procedure name
Name of the JCL started procedure used to start the migrated servants

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the servant procedure or use the old one.

Replace started procedure command names
If you specified new names for your JCL procedures, then the corresponding START commands in the WebSphere Application Server configuration must be updated to match the new procedure names. Select this option to perform this configuration update.

If you choose to use the same procedure names, then do not select this option. If you are not using consistent procedure names for all the servers of a given process type (all servants for example) for the node that you are migrating, then it is recommended that you do not select this option. In this case, you will need to keep the same START commands and manually replace the procedures using the procedure that is generated during migration as a template.

Notes:

- Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x configuration. The migration process creates new Version 8.0 JCL procedures using the procedure names specified here.
- If you use the same names that you used in your Version 6.x configuration, the migration process overlays the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x procedures before running the migration jobs in case you need to roll back later.

WebSphere administrator's user ID

The migration process requires the use of an administrative client. Specify a valid WebSphere administrator's user ID that the administrative client can use to authenticate.

If you want, you can defer entering a user ID at this point and manually edit the BBOWMG3D, BBOWDPRO, BBOWDPRE, or BBOWDPOS jobs later. To do this, update the `TO_AdminUserid` variable with the proper user ID before submitting the job.

WebSphere administrator's password

The migration process requires the use of an administrative client. Specify a valid WebSphere administrator's password that the administrative client can use to authenticate.

If you want, you can defer entering a password at this point and manually edit the BBOWMG3D, BBOWDPRO, BBOWDPRE, or BBOWDPOS jobs later. To do this, update the `TO_AdminPassword` variable with the proper password before submitting the job.

Server Customization (Part 2)

Migrate to support script compatibility

Whether or not to migrate to support script compatibility

This specifies whether or not migration should create the following Version 6.x configuration definitions:

- Transport
- ProcessDef
- Version 6.x SSL
- Version 6.x ORB service threadpool

instead of the following Version 8.0 configuration definitions:

- Channels
- ProcessDefs
- Version 8.0 SSL
- Version 8.0 ORB service threadpool

Choose to migrate to support script compatibility if you want to minimize impacts to existing administration scripts. If you have existing wsadmin scripts or programs that use third-party configuration APIs to create or modify the Version 6.x configuration definitions, for example, you might want to choose this option.

If you want to use a cell that contains Version 6.x nodes, you must specify true for this variable.

Note: This is meant to provide a temporary transition until all of the nodes in the environment are at the Version 8.0 level. When they are all at the Version 8.0 level, you should perform the following actions:

1. Modify your administration scripts to use all of the Version 8.0 settings.
2. Use the convertScriptCompatability command to convert your configurations to match all of the Version 8.0 settings.

   Read “convertScriptCompatibility command” on page 35 for more information.

Disable previous deployment manager

Whether or not to disable the previous deployment manager during migration

If the Version 6.x deployment manager is not disabled, you can continue to use it while the migration is completed.

Caution: Use this option with care. The reason that WebSphere Application Server Version 6.x deployment managers normally are stopped and disabled is to prevent multiple deployment managers from managing the same nodes. You must stop the Version 6.x

deployment manager before you start using the Version 8.0 deployment manager. Deselecting this option means that any configuration changes made in the old configuration during migration might not be migrated.

Application migration preference
How you would like to migrate your installed applications

Note: WebSphere Application Server system applications will migrate regardless of the value set here.

Migrate applications and use the default application installation directory
Install user enterprise applications in the default application installation directory as part of the migration

Migrate applications and use the specified application installation directory
Install user enterprise applications in a specified application installation directory as part of the migration

Application installation directory
Location where WebSphere Application Server installs your enterprise applications

This location is used when you specify that you want to migrate and install applications as your application migration preference. You have the option to choose a customized environment-specific location or use the default location.

Migrate and generate administrative scripts to install applications later
Prepare user enterprise applications for installation in the WebSphere Application Server Version 8.0 `installableApps` directory without actually installing them during migration

Scripts that can be used to install these applications are generated and saved in the migration backup directory. For WebSphere Application Server for z/OS, the location of this backup directory is relative to the temporary directory that you specify on this same panel. The location of the backup directory is also determined by the derived migration identifier and the type of node being migrated. If you specify `/tmp/migrate` as the temporary directory and the derived migration identifier is 55449, for example, then the location of the generated scripts is:

`/tmp/migrate/55449/`*`nodetype`*`_backup/`

where *nodetype* is dmgr, fed, or base depending on the type of node that you are migrating.

You can then run these files at any point and in any combination after migration has completed. You can also reorganize and combine these files for better application installation efficiency. Read the "Wsadmin tool" article in the information center for additional information.

Migrate applications and use the previous application installation directory
Install user enterprise applications as part of the migration, and keep the same application installation directories as the previous version

Restrictions: If you select this option, the location is shared by the existing WebSphere Application Server Version 6.x installation and the Version 8.0 installation. If you keep the migrated applications in the same locations as those of the previous version, the following restrictions apply:

- The WebSphere Application Server Version 8.0 mixed-node support limitations must be followed. This means that the following support cannot be used when evoking the wsadmin command:
  - Precompile JSP
  - Use Binary Configuration

  - Deploy EJB
- You risk losing the migrated applications unintentionally if you later delete applications from these locations when administering (uninstalling for example) your Version 6.x installation.
- Any application that is installed relative to a Version 6.x variable will be installed relative to the location assigned to that variable in Version 8.0. In other words, the absolute location is not preserved—the application is migrated to the relative location within the new Version 8.0 environment.

  If the binariesURL in the `deployment.xml` file for an application being migrated has a path that is relative to WebSphere Application Server—that is, it begins with `$(APP_INSTALL_ROOT)`, `$(WAS_INSTALL_ROOT)`, and so on—the new WebSphere Application Server variable value is used to resolve the path when the application is installed in the new location. This leads to the following results when you select this option:
  - Any application that is installed in a directory location relative to a WebSphere Application Server variable is installed under that variable value in Version 8.0.
  - Any application that is installed in a directory location that is not relative to a WebSphere Application Server variable is migrated and overwritten in that same directory. If an application is installed in the `/employee_records/retrieval_Apps` directory, for example, the application is migrated and overwritten in the `/employee_records/retrieval_Apps` directory.

Do not migrate applications
Do nothing with user enterprise applications

Migration trace options

If you choose to enable tracing, it will remain enabled throughout the entire migration process.

Enable script tracing
Enable or disable trace of the home creation, profile and migration tooling invocation, and final processing phases of migration

Enable profile creation tracing
Enable or disable trace trace during profile creation

Enable pre-upgrade tracing
Enable or disable trace during the WASPreUpgrade process

Enable post-upgrade tracing
Enable or disable trace during the WASPostUpgrade process

Migrate administrative console customized "My tasks" settings
"My tasks" is only supported in WebSphere Application Server Version 6.1 and later.

Migrate the settings for "My tasks" saved in the default workspace user root location (wstemp)

Migrate the settings for "My tasks" saved in a user defined workspace root location
User defined workspace root location

Temporary directory location
Directory where the backup of your previous configuration is written as well as the migration trace and temporary file output

During migration, a backup copy of the Version 6.x configuration is required. The default location of this backup is already specified, though you can override if needed. You might need to specify a location other than the default if the `/tmp` file system does not have adequate space to store the

backup configuration. If you choose to override the default location of the backup copy, the best practice is to keep the same naming convention and just replace the `/tmp` portion with another path, `/myTemp/migrate` for example.

Migration definition identifier
Identifier that will be used to create a directory under the temporary directory that will contain the temporary migration datasets and backup configuration data

Job Statement Definition

All the migration jobs that will be tailored for you will need a job statement.

Enter a valid job statement for your installation. The migration creation process will update the job name for you in all the generated jobs, so you do not need to be concerned with that portion of the job statement. If continuation lines are needed, replace the comment lines with continuation lines.

Specifying a deployment manager migration configuration in the z/OS Migration Management Tool: Worksheet:

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. Print this worksheet, and use it when collecting and specifying information about the WebSphere Application Server for z/OS deployment manager migration configuration variables.

Before you begin

Procedure

1. Specify a value for the **Migration Node Type Selection** panel.

Table 28. Migration Node Type Selection. Enter your values:

Variable	Options	Your value
Migration node type	• z/OS migrate a stand-alone application server • z/OS migrate a deployment manager • z/OS migrate a federated node • z/OS migrate an administrative agent • z/OS migrate a job manager	z/OS migrate a deployment manager

2. Specify values for the **Migration Definition Name** panel.

Table 29. Migration Definition Name. Enter your values:

Variable	Default	Your value
Migration definition name	`ZMigDmgr`*nn*	
Response file path name (optional)	None	

3. Specify values for the **Target Datasets** panel.

Table 30. Target Datasets. Enter your values:

Variable	Default	Your value
High-level qualifier (HLQ)	None	

4. Specify values for the **Dataset Names and Product Directory** panel.

Table 31. Dataset Names and Product Directory. Enter your values:

Variable			Default	Your value
JCL procedure library dataset name			SYS1.PROCLIB	
WebSphere Application Server product directory			/usr/lpp/zWebSphere/V8R0	
Intermediate symbolic link				
	Create intermediate symbolic link		Selected	
		Path name of intermediate symbolic link	/wasv8config/*cell_long_name*/*node_long_name*/wasInstall	

5. Specify values for the **Configuration File System** panel.

Table 32. Configuration File System. Enter your values:

Variable		Default	Your value
Mount point		/wasv8config	
Name		OMVS.WAS80.CONFIG.HFS	
Volume, or '*' for SMS		*	
Primary allocation in cylinders		420	
Secondary allocation in cylinders		100	
File system type			
	Hierarchical File System (HFS)	Selected	
	zSeries File System (ZFS)	Not selected	

6. Specify values for the **Server Customization (Part 1)** panel.

Table 33. Server Customization (Part 1). Enter your values:

Variable		Default	Your value
From configuration location			
	Mount point	/WebSphere/V6R0M0	
	Home directory	DeploymentManager	
To configuration location			
	Mount point	Value from **Configuration File System** panel	
	Home directory	DeploymentManager	
Daemon procedure name		BBO8DMN	
Controller procedure name		BBO8DCR	

Table 33. Server Customization (Part 1) (continued). Enter your values:

Variable	Default	Your value
Servant procedure name	BBO8DSR	
Replace started procedure command names	Selected	
WebSphere administrator's user ID	XXXXXXXX	
WebSphere administrator's password	XXXXXXXX	

7. Specify values for the **Server Customization (Part 2)** panel.

Table 34. Server Customization (Part 2). Enter your values:

Variable			Default	Your value
Migrate to support script compatibility			Selected	
Disable previous deployment manager			Selected	
Application migration preference				
	Migrate applications and use the default application installation directory		Selected	
	Migrate applications and use the specified application installation directory		Not selected	
		Application installation directory	*/mount_point/home_directory* /profiles/default/installedApps	
	Migrate and generate administrative scripts to install applications later		Not selected	
	Migrate applications and use the previous application installation directory		Not selected	
	Do not migrate applications		Not selected	
Migration trace options				
	Enable script tracing		Not selected	
	Enable profile creation tracing		Not selected	
	Enable pre-upgrade tracing		Not selected	
	Enable post-upgrade tracing		Not selected	
Migrate administrative console customized "My tasks" settings				
	Migrate the settings for "My tasks" saved in the default workspace user root location (${USER_INSTALL_ROOT}/wstemp)		Selected	
	Migrate the settings for "My tasks" saved in a user defined workspace root location		Not selected	
		User defined workspace root location	None	
Temporary directory location			/tmp/migrate	
Migration definition identifier			Generated value	

8. Specify values for the **Job Statement Definition** panel.

Table 35. Job Statement Definition. Enter your values:

//jobname JOB	(ACCTNO,ROOM),'USERID',CLASS=A,REGION=OM
//*	
//*	
//*	

Preparing to migrate a z/OS deployment manager using the zmmt command

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the zmmt command to create jobs for migrating a deployment manager.

Before you begin

Procedure

1. Print and complete the “Specifying a z/OS deployment manager migration configuration in the zmmt command response file: Worksheet” on page 86.
2. Follow the instructions in “Using the zmmt command to create migration definitions on z/OS” on page 32.

   Use the values from your “Specifying a z/OS deployment manager migration configuration in the zmmt command response file: Worksheet” on page 86.

What to do next

1. Follow the instructions in the BBOMDINS member of the CNTL dataset that you generated using the zmmt command.

   Read “Migrating a z/OS deployment manager” on page 88 for details.
2. After you migrate the deployment manager, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer it.

Configuration variables for migrating a z/OS deployment manager using the zmmt command:

Before you migrate WebSphere Application Server for z/OS, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run during the actual migration. You can use the zmmt command with a response file to create the appropriate migration jobs. The migration response file contains a set of configuration variables that you use to create jobs for migrating a deployment manager.

Response-file variables

create Required keyword to indicate the creation of a new migration definition

profileName
Name of the profile created during migration

This is always "default" on the z/OS platform.

profilePath
Fully qualified path where the generated migration definition output should be written

templatePath
Template path

One of the following values:

- zos-migDmgr

Migrate a deployment manager

- zos-migFederated

  Migrate a federated node

- zos-migStandalone

  Migrate a stand-alone application server

zConfigHfsName
Name of the MVS dataset that will contain the configuration file system

This name is used as input to the migration job that creates the configuration file system.

zConfigHfsPrimaryCylinders
Number of primary cylinders that will be allocated to the configuration file system

In an application server, the total space needed for this dataset increases with the size and number of installed applications.

Recommendation: The minimum suggested size is 420 cylinders.

zConfigHfsSecondaryCylinders
Number of secondary cylinders that will be allocated to the configuration file system

Recommendation: The minimum suggested size is 100 cylinders.

zConfigHfsVolume
DASD volume serial number to contain the above dataset, or "*" to let SMS select a volume

Using "*" requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle dataset allocation automatically, list the volume explicitly.

zConfigMountPoint
File system directory mount point where application data and environment files are written

The configuration file system is where the configuration for the node being migrated is physically stored. You can choose to use an existing Version 8.0 file system if you already have an appropriate file system on the node being migrated. If you choose to use an existing Versions 8.0 file system, you need to ensure that the mount point that you specify here is present before running the migration jobs that are created using this tool. If you choose to create a new Version 8.0 file system on the node being migrated, the actual creation of the new file system will not occur until you run the optional BBOMDHFS or BBOMDZFS job during the actual migration process. In any case, you must specify the correct value here for the mount point.

zFilesystemType
Type of file system

One of the following values:

- HFS

  Allocate and mount your configuration file system dataset using the Hierarchical File System

- ZFS

  Allocate and mount your configuration file system dataset using the zSeries File System

zJobStatement1 . . . *n*
Customizable JOB statements that will be used for the generated migration jobs

All the migration jobs that will be tailored for you will need a job statement. Enter a valid job statement for your installation. The migration creation process will update the job name for you in all the generated jobs, so you do not need to be concerned with that portion of the job statement. If continuation lines are needed, replace the comment lines with continuation lines.

zTargetHLQ
High-level qualifier for the target z/OS datasets that will contain the generated jobs and instructions

Note: A multilevel high-level qualifier can be specified as the dataset high-level qualifier.

zmbAdminUserid
User ID of an administrator that is used to manage the node that is being migrated

This is required to perform required administrative actions during the migration process.

zmbAdminPassword
Password for the user ID of the administrator that is used to manage the node that is being migrated

This is required to perform required administrative actions during the migration process.

zmbAppMigrationPreference
How you would like to migrate your installed applications

One of the following values:

- D

  Install user enterprise applications in the default application installation directory as part of the migration
- Y

  Install user enterprise applications in a directory specified in the following variables as part of the migration

  zmbAppInstallDirLine1
  zmbAppInstallDirLine2
  Location where WebSphere Application Server installs your enterprise applications

  This location is used when you specify that you want to migrate and install applications as your application migration preference. You have the option to choose a customized environment-specific location or use the default location.

  If the location path length exceeds 60 characters, then it must be specified on two lines as shown; but it cannot exceed a total length of 120 characters.

  Leave these two fields blank unless you specify Y for zmbAppMigrationPreference.
- S

  Prepare user enterprise applications for installation in the WebSphere Application Server Version 8.0 `installableApps` directory without actually installing them during migration

  Scripts that can be used to install these applications are generated and saved in the migration backup directory. For WebSphere Application Server for z/OS, the location of this backup directory is relative to the temporary directory that you specify for the zmbTempDirectory variable. The location of the backup directory is also determined by the derived migration identifier and the type of node being migrated. If you specify `/tmp/migrate` as the temporary directory and the derived migration identifier is 55449, for example, then the location of the generated scripts is:

  `/tmp/migrate/55449/`*`nodetype`*`_backup/`

  where *nodetype* is dmgr, fed, or base depending on the type of node that you are migrating.

  You can then run these files at any point and in any combination after migration has completed. You can also reorganize and combine these files for better application installation efficiency. Read the "Wsadmin tool" article in the information center for additional information.
- P

Install user enterprise applications as part of the migration, and keep the same application installation directories as the previous version

Restrictions: If you select this option, the location is shared by the existing WebSphere Application Server Version 6.x installation and the Version 8.0 installation. If you keep the migrated applications in the same locations as those of the previous version, the following restrictions apply:

- The WebSphere Application Server Version 8.0 mixed-node support limitations must be followed. This means that the following support cannot be used when evoking the wsadmin command:
  - Precompile JSP
  - Use Binary Configuration
  - Deploy EJB
- You risk losing the migrated applications unintentionally if you later delete applications from these locations when administering (uninstalling for example) your Version 6.x installation.
- Any application that is installed relative to a Version 6.x variable will be installed relative to the location assigned to that variable in Version 8.0. In other words, the absolute location is not preserved—the application is migrated to the relative location within the new Version 8.0 environment.

  If the binariesURL in the `deployment.xml` file for an application being migrated has a path that is relative to WebSphere Application Server—that is, it begins with `$(APP_INSTALL_ROOT)`, `$(WAS_INSTALL_ROOT)`, and so on—the new WebSphere Application Server variable value is used to resolve the path when the application is installed in the new location. This leads to the following results when you select this option:
  - Any application that is installed in a directory location relative to a WebSphere Application Server variable is installed under that variable value in Version 8.0.
  - Any application that is installed in a directory location that is not relative to a WebSphere Application Server variable is migrated and overwritten in that same directory. If an application is installed in the `/employee_records/retrieval_Apps` directory, for example, the application is migrated and overwritten in the `/employee_records/retrieval_Apps` directory.

- N

  Do nothing with user enterprise applications

Note: WebSphere Application Server system applications will migrate regardless of the value set here.

zmbControllerProcName

Name of the JCL started procedure used to start the migrated controllers

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the controller procedure or use the old one.

zmbServantProcName

Name of the JCL started procedure used to start the migrated servants

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the servant procedure or use the old one.

zmbDaemonProcName
Name of the JCL started procedure used to start the migrated daemon

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the daemon procedure or use the old one.

zmbReplaceStartedProcedureNames
Whether to update the START commands in the configuration with the new names specified (true) or to preserve the same names (false).

If you specified new names for your JCL procedures, then the corresponding START commands in the WebSphere Application Server configuration must be updated to match the new procedure names. Specify true for this variable to perform this configuration update.

If you choose to use the same procedure names, then specify false for this variable. If you are not using consistent procedure names for all the servers of a given process type (all servants for example) for the node that you are migrating, then it is recommended that you specify false for this variable. In this case, you will need to keep the same START commands and manually replace the procedures using the procedure that is generated during migration as a template.

Notes:

- Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x configuration. The migration process creates new Version 8.0 JCL procedures using the procedure names specified here.
- If you use the same names as you used in your Version 6.x configuration, the migration process overlays the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x procedures before running the migration jobs in case you need to roll back later.

zmbDisableDmgr
Whether (true) or not (false) to disable the previous deployment manager during migration

If the Version 6.x deployment manager is not disabled, you can continue to use it while the migration is completed.

Caution: Use this option with care. The reason that WebSphere Application Server Version 6.x deployment managers normally are stopped and disabled is to prevent multiple deployment managers from managing the same nodes. You must stop the Version 6.x deployment manager before you start using the Version 8.0 deployment manager. Deselecting this option means that any configuration changes made in the old configuration during migration might not be migrated.

zmbEnablePostUpgradeTrace
Enable (true) or disable (false) trace during the WASPostUpgrade process

zmbEnablePreUpgradeTrace
Enable (true) or disable (false) trace during the WASPreUpgrade process

zmbEnableProfileTrace
Enable (true) or disable (false) trace during profile creation

zmbEnableScriptingTrace
Enable (true) or disable (false) trace of the home creation, profile and migration tooling invocation, and final processing phases of migration

zmbEnableScriptCompatibility
Whether (true) or not (false) to migrate to support script compatibility

This specifies whether or not migration should create the following Version 6.x configuration definitions:

- Transport
- ProcessDef
- Version 6.x SSL
- Version 6.x ORB service threadpool

instead of the following Version 8.0 configuration definitions:

- Channels
- ProcessDefs
- Version 8.0 SSL
- Version 8.0 ORB service threadpool

Choose to migrate to support script compatibility if you want to minimize impacts to existing administration scripts. If you have existing wsadmin scripts or programs that use third-party configuration APIs to create or modify the Version 6.x configuration definitions, for example, you might want to specify true for this variable.

If you want to use a cell that contains Version 6.x nodes, you must specify true for this variable.

Note: This is meant to provide a temporary transition until all of the nodes in the environment are at the Version 8.0 level. When they are all at the Version 8.0 level, you should perform the following actions:

1. Modify your administration scripts to use all of the Version 8.0 settings.
2. Use the convertScriptCompatability command to convert your configurations to match all of the Version 8.0 settings.

   Read “convertScriptCompatibility command” on page 35 for more information.

zmbFromConfigRoot
Mount point of the configuration from which you are migrating

zmbFromWASHomeDir
Home directory of the configuration from which you are migrating

zmbProclibName
Existing procedure library to which the WebSphere Application Server for z/OS cataloged procedures are to be copied

zmbSMPEHome
Location of your WebSphere Application Server Version 8.0 SMPE installed product file system

zmbTempDirectory
Directory where the backup of your previous configuration is written as well as the migration trace and temporary file output

During migration, a backup copy of the Version 6.x configuration is required. The default location of this backup is already specified, though you can override if needed. You might need to specify a location other than the default if the `/tmp` file system does not have adequate space to store the backup configuration. If you choose to override the default location of the backup copy, the best practice is to keep the same naming convention and just replace the `/tmp` portion with another path, `/myTemp/migrate` for example.

zmbTimestamp
Identifier that will be used to create a directory under the temporary directory that will contain the temporary migration datasets and backup configuration data

zmbToConfigRoot
Mount point of the configuration to which you are migrating

This is the same value that is specified for the zConfigMountPoint variable.

zmbToWASHomeDir
Home directory of the configuration to which you are migrating

zmbWorkspaceRootPreference
Whether to migrate the administrative console customized "My tasks" settings saved in the default workspace user root location (D) or to migrate the settings saved in a user-defined workspace root location (U)

zmbUserWorkspaceRoot
User-defined workspace root location

intermediateSymlinkPreference
Whether to set up an intermediate symbolic link

If you specify an intermediate symbolic link, symbolic links are created from the configuration file system to the intermediate symbolic link; otherwise, they are created directly to the product file system.

IntermediateSymlink
Path name of intermediate symbolic link

This link will be created by the customization jobs, pointing to the product file system directory.

Sample of a response file for migrating a deployment manager

```
create
profileName=default
profilePath=/zmmt/workspace/.metadata/.plugins/com.ibm.ws390.mmt.config/profiles/ZMigDmgr01
templatePath=zos-migDmgr
zConfigHfsName=OMVS.WAS80.CONFIG.HFS
zConfigHfsPrimaryCylinders=420
zConfigHfsSecondaryCylinders=100
zConfigHfsVolume=*
zConfigMountPoint=/wasv8config
zFilesystemType=HFS
zJobStatement1=(ACCTNO,ROOM),'USERID',CLASS=A,REGION=0M
zJobStatement2=//*
zJobStatement3=//*
zJobStatement4=//*
zTargetHLQ=DMR
zmbAdminPassword=XXXXXXXX
zmbAdminUserid=XXXXXXXX
zmbAppInstallDirLine1=/wasv8config/ Manager/profiles/default/installedApps
zmbAppInstallDirLine2=
zmbAppMigrationPreference=D
zmbControllerProcName=BBO7DCR
zmbDaemonProcName=BBO7DMN
zmbDisableDmgr=true
zmbEnablePostUpgradeTrace=true
zmbEnablePreUpgradeTrace=true
zmbEnableProfileTrace=true
zmbEnableScriptCompatibility=true
zmbEnableScriptingTrace=true
zmbFromConfigRoot=/WebSphere/V5R1M0
zmbFromWASHomeDir= Manager
zmbProclibName=SYS1.PROCLIB
zmbReplaceStartedProcedureNames=true
zmbSMPEHome=/usr/lpp/zWebSphere/Version 8.0
zmbServantProcName=BBO7DSR
zmbTempDirectory=/tmp/migrate
zmbTimestamp=30082852
zmbToConfigRoot=/wasv8config
```

```
zmbToWASHomeDir= Manager
zmbWorkspaceRootPreference=D
intermediateSymlinkPreference=Y
IntermediateSymlink=/V80SMPE/symblink
```

Specifying a z/OS deployment manager migration configuration in the zmmt command response file: Worksheet:

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the zmmt command with a response file to create the appropriate migration jobs. Print this worksheet, and use it when collecting and specifying information about the WebSphere Application Server for z/OS deployment manager migration configuration variables in the zmmt command response file.

Before you begin

Procedure

1. Specify "create."
2. Specify values for the name and location.

Table 36. Name and location. Enter your values.

Variable	Response-file variable	Example value	Your value
Profile name	profileName	Default	
Profile directory	profilePath	`/AppServer/profiles`	
Template path	templatePath	zos-migDmgr	

3. Specify values for the target datasets.

Table 37. Target datasets. Enter your values.

Variable	Response-file variable	Example value	Your value
High-level qualifier (HLQ)	zTargetHLQ	DMR	

4. Specify values for the dataset names, product directory, and intermediate symbolic link.

Table 38. Dataset names, product directory, and intermediate symbolic link. Enter your values.

Variable			Response-file variable	Example value	Your value
JCL procedure library dataset name			zmbProclibName	SYS1.PROCLIB	
WebSphere Application Server product directory			zmbSMPEHome	`/usr/lpp/zWebSphere/V7R0`	
Intermediate symbolic link					
	Create intermediate symbolic link		intermediateSymlinkPreference	Y	
		Path name of intermediate symbolic link	IntermediateSymlink	/V70SMPE/symblink	

5. Specify values for the configuration file system.

Table 39. Configuration file system. Enter your values.

Variable		Response-file variable	Example value	Your value
Mount point		zConfigMountPoint	`/wasv7config`	
Name		zConfigHfsName	OMVS.WAS70. CONFIG.HFS	
Volume, or '*' for SMS		zConfigHfsVolume	*	
Primary allocation in cylinders		zConfigHfsPrimaryCylinders	420	
Secondary allocation in cylinders		zConfigHfsSecondaryCylinders	100	
File system type				
	Hierarchical File System (HFS)	zFilesystemType	HFS	
	zSeries File System (ZFS)			

6. Specify values for server customization.

Table 40. Server customization. Enter your values.

Variable		Response-file variable	Example value	Your value
From configuration location				
	Mount point	zmbFromConfigRoot	`/WebSphere/V5R1M0`	
	Home directory	zmbFromWASHomeDir	`DeploymentManager`	
To configuration location				

Table 40. Server customization (continued). Enter your values.

Variable		Response-file variable	Example value	Your value
	Mount point	zmbToConfigRoot	/WebSphere/V7R0	
	Home directory	zmbToWASHomeDir	DeploymentManager	
Daemon procedure name		zmbDaemonProcName	BBO7DMN	
Controller procedure name		zmbControllerProcName	BBO7DCR	
Servant procedure name		zmbServantProcName	BBO7DSR	
Replace started procedure command names		zmbReplaceStartedProcedure Names	True	
WebSphere administrator's user ID		zmbAdminUserid	XXXXXXXX	
WebSphere administrator's password		zmbAdminPassword	XXXXXXXX	

Table 41. Server customization. Enter your values.

Variable			Response-file variable	Example value	Your value
Migrate to support script compatibility			zmbEnableScriptCompatibility	True	
Disable previous deployment manager			zmbDisableDmgr	True	
Application migration preference					
	Migrate applications and use the default application installation directory (D)		zmbAppMigrationPreference	Y	
	Migrate and generate administrative scripts to install applications later (S)				
	Migrate applications and use the previous application installation directory (P)				
	Do not migrate applications (N)				
	Migrate applications and use the specified application installation directory (Y)				
		Application installation directory	zmbAppInstallDirLine1	/wasv7config/AppServer/ profiles/default/ installedApps	
			zmbAppInstallDirLine2	Apps	
Migration trace options					
	Enable script tracing		zmbEnableScriptingTrace	False	
	Enable profile creation tracing		zmbEnableProfileTrace	False	
	Enable pre-upgrade tracing		zmbEnablePreUpgradeTrace	False	
	Enable post-upgrade tracing		zmbEnablePostUpgradeTrace	False	
Migrate administrative console customized "My tasks" settings					
	Migrate the settings for "My tasks" saved in the default workspace user root location (D)		zmbWorkspaceRoot Preference	D	
	Migrate the settings for "My tasks" saved in a user defined workspace root location (U)				
		User defined workspace root location	zmbUserWorkspaceRoot		
Temporary directory location			zmbTempDirectory	/tmp/migrate	
Migration definition identifier			zmbTimestamp	21130546	

7. Specify values for job statement definition.

Table 42. Job statement definition. Enter your values.

Variable	Response-file variable	Example value	Your value
//jobname JOB	zJobStatement1	(ACCTNO,ROOM),'USERID', CLASS=A,REGION=OM	
//*	zJobStatement2	//*	
//*	zJobStatement3	//*	
//*	zJobStatement4	//*	

Migrating a z/OS deployment manager: Checklist

Activities for migrating a deployment manager to WebSphere Application Server Version 8.0 after you generate the appropriate Job Control Language (JCL) jobs include activities related to general preparation and node migration. Use the following checklist as a quick reference to these activities. Follow the instructions that were generated by the z/OS Migration Management Tool or the zmmt command to complete the process of migrating your deployment manager to Version 8.0.

Before you begin

You can find more details about migrating a deployment manager in “Preparing to migrate a WebSphere Application Server, Network Deployment cell for z/OS” on page 65 and “Migrating a z/OS deployment manager” on page 88.

Procedure

1. Prepare to migrate the deployment manager.

Table 43. Deployment manager checklist. Enter your values:

Check off	Item
_____	Inventory your existing Version 5.1.x or Version 6.x configuration and capture key information. Read “Preparing to migrate a WebSphere Application Server, Network Deployment cell for z/OS” on page 65 for more information.
_____	Map a strategy for the sequence of nodes to be migrated.
_____	Create CNTL and DATA datasets for each node to be migrated. Read “Preparing to migrate a deployment manager using the z/OS Migration Management Tool” on page 69 or “Preparing to migrate a z/OS deployment manager using the zmmt command” on page 79 for more information.
_____	Create the Version 7.0 configuration mount point, allocate a configuration file system, and mount either manually or by running the BBOMDHFS or BBOMDZFS job.
_____	Customize and run the BBOMDCP job to copy the generated JCL procedures to your procedure library.

2. Migrate the deployment manager.

Table 44. Deployment manager checklist. Enter your values:

Check off	Item
_____	Run the migration jobs—including the BBOWMG3D job or the BBOWDPRO, BBOWDPRE, and BBOWDPOS jobs—by following the generated instructions; and check for success.
_____	Start the Version 7.0 copy of the deployment manager.

Migrating a z/OS deployment manager

After you generate JCL jobs for migrating a deployment manager to WebSphere Application Server for z/OS Version 8.0, you can perform the actual migration by running those jobs. When you generated your custom migration jobs, you also created customized instructions for preparing and running the migration jobs in the BBOMDINS members of the CNTL dataset that was used to generate your jobs. Follow these customized instructions to complete the process of migrating your deployment manager to Version 8.0.

Before you begin

- Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6.
- You will not be able to proceed if you did not generate the Job Control Language (JCL) migration jobs.
- Always migrate the deployment manager node first.
- Review the applicable coexistence requirements for your system.
- The BBOWMG3D job referenced in these instructions must be submitted by a WebSphere Administrator User ID.

  All other jobs must be submitted by a user ID that has control over the file system.
- Note on maintaining high availability during migration from Version 6.x or 7.x to Version 8.0:

Migrating a deployment manager from Version 6.x or 7.x to Version 8.0 redeploys all application binaries when the deployment manager is restarted. This action results in the deployment manager recycling all the applications across the sysplex. This can cause an outage in a sysplex that is set up for high availability if the synchronization settings are not disabled.

To maintain high availability during the migration, turn off all synchronization options for all of the node agents in the sysplex before restarting the deployment manager.

1. Open the administrative console.
2. Go to **System administration > Node agents**.
3. For each node agent in the sysplex, go to ***node_agent_server_name*** **> File synchronization service** and disable all synchronization processing.

- **Tip:** Before migrating a WebSphere Application Server Version 6.x or 7.x deployment manager, use the backupConfig command or your own preferred backup utility to back up your existing configuration if you want to be able to restore it to its previous state after migration. Read the "backupConfig command" article in the information center for more information. Make sure that you note the exact name and location of this backed-up configuration.

For help, read Chapter 11, “Troubleshooting migration,” on page 143.

Procedure

1. Create and mount a new Version 8.0 configuration file system.

   Before you perform the migration, Version 8.0 requires a configuration file system to be present for your new configuration. You can run BBOMDHFS or BBOMDZFS to create and mount a new configuration file system, or you can mount one manually. Either way, you must have a configuration file system for your Version 8.0 configuration created and mounted before you proceed. This configuration file system is the target of the migration; your Version 6.x or Version 7.x configuration file system is the source.

   BBOMDHFS or BBOMDZFS creates a mount point directory, allocates the configuration's file system, and mounts the file system at whatever value you specified for the mount point when you generated you migrations jobs.

   Ensure that you have allocated, created, and mounted your configuration file system data sets either manually or using BBOMDHFS or BBOMDZFS before you proceed. The mount point should be owned by the WebSphere Admin ID, and have permissions of at least 755. The new configuration file system structures should be included in BPXPARM so that they will be mounted at the next IPL.

2. Copy your generated JCL procedures.

   The migration utility BBOMDCP copies the generated JCL procedures to start the servers to the specified procedure library. Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x or Version 7.x configuration. This utility will update the new Version 8.0 configuration, substituting your new JCL names in place of the names that existed in your original Version 6.x or Version 7.x configuration.

   Caution: This utility copies the generated JCL to your procedure library. If you specified the same names as you used in your Version 6.x or Version 7.x configuration when you generated your migration jobs, this utility will overlay the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x or Version 7.x procedures before running this utility in case you need to roll back later.

   Submit BBOMDCP, and verify a return code of 0.

3. If you specified new procedure names, update your RACF STARTED profiles for the controller and daemon.

   The STARTED profile used by controller regions is based on the procedure name and JOBNAME. You must ensure that a STARTED profile will apply so that the proper identity will be assigned to the

started task. For example, if your Version 6.x or Version 7.x deployment manager controller JCL procedure name is AZDCR, and you specified AZ1DCR for Version 8.0, then you would need to create a STARTED profile for that new procedure name:

```
              new controller      same identity used in
                 JCL name         V6.x or V7.x configuration
                    |                      |
 RDEFINE STARTED AZ1DCR.* STDATA(USER(AZDCRU) GROUP(AZCFG) TRACE(YES))
```

Note:

- Do not use a different user ID to start. There are other things tied to the user ID, and other changes would also be required if you change the user ID.
- If your original STARTED profile was generic, STARTED AZ*.* ... for example, you would not need to create a new STARTED profile.
- Servant region STARTED profiles are based on JOBNAME, not procedure name. So there is no issue with the servant when you use a different procedure name.
- Daemons and node agents are controllers, so using different procedure names for those implies a new STARTED profile.

4. Perform one of the following actions:
   a. Submit the BBOWMG3D job.

      A deployment manager migration does not require bringing the node into and out of Peer Restart and Recovery (PRR) mode as stand-alone application server and federated node migrations do. There are two less jobs to submit for a deployment manager migration, therefore, and you are ready to perform the physical migration.

      BBOWMG3D is the job that performs the physical migration of the Version 6.x or Version 7.x deployment manager to Version 8.0 based on the information that you supplied when you generated your migration jobs. Submit BBOWMG3D. Verify that you are getting return codes of 0, and review the log files in the migration temp directory on the configuration file system. The migration temporary directory is *temporary_directory_location*/*nnnnn*, where *temporary_directory_location* is the directory specified for the temporary directory location (/tmp/migrate by default) and *nnnnn* is the numeric value generated for the migration identifier when you generated your migration jobs.

   b. Submit the following three jobs:
      1) Submit the BBOWDPRO job.

         BBOWDPRO creates the Websphere Application Server home and default profile.

      2) Submit the BBOWDPRE job.

         BBOWDPRE runs the migration pre-upgrade process.

      3) Submit the BBOWDPOS job.

         BBOWDPOS runs the migration post-upgrade and finish-up (change file permission) processes.

5. Shut down the old application servers and daemon.

   Ensure that all nodes on the deployment manager's LPAR in the same cell are shut down.

6. Update daemon JCL procedure if necessary.

   WebSphere Application Server for z/OS Version 8.0 requires that the daemon process be at the highest level of code of any of the servers that it manages on the same LPAR. It will be at the Version 8.0 level when the deployment manager is started.

   After you migrate all nodes to Version 8.0 and before you remove the previous version's libraries from the system, you must update the daemon JCL procedure. Failure to do so will result in a failure of the daemon to start.

   Note:

   - If you are migrating from Version 6.x or 7.x to Version 8.0, the daemon needs to have a STEPLIB that includes the Version 7.x SBBOLD2 and SBBOLPA datasets.

- If a Version 8.0 cell has a Version 8.0 server that is communicating with a Version 6.0.1.x or 7.x server in a Version 6.0.1.x or 7.x cell that is on the same system as the Version 8.0 cell, you also need to add the version 6.0.1.x SBBOLD2 and SBBOLPA datasets to the STEPLIB of the Version 8.0 daemon.

7. Start the new deployment manager.

   Use the existing commands that you would use to start a Version 6.x or Version 7.x application server, but replace the RACF STARTED procedure name with the value that you entered in the deployment manager panel for the controller procedure name when you generated your migration jobs. This command starts the Version 8.0 deployment manager. Wait until the server is finished initializing before proceeding.

   The following message is displayed on the console and in the job log of BBODMGR:

   ```
   BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR z/OS CONTROL PROCESS BBODMGR
   ```

What to do next

After you verify a successful migration to Version 8.0 and are successfully running a migrated configuration, you should delete the following items:

- Everything in the source configuration's file system
- Everything in the target configuration's *temporary_directory_location*/*nnnnn* directory, where *temporary_directory_location* is the directory specified for the temporary directory location (/tmp/migrate by default) and *nnnnn* is the numeric value generated for the migration identifier when you created your migration jobs
- The bbomigrt2.sh file

Migrating federated nodes for z/OS

When you migrate federated nodes to WebSphere Application Server Version 8.0, use the available planning information, high-level migration checklists, and detailed migration explanations to learn how to generate the migration jobs and plan your migration.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6.

For resources to help you plan and perform your migration, visit Knowledge Collection: Migration planning for WebSphere Application Server.

Important: Before you migrate a federated node, you must migrate its deployment manager first. If you have not migrated the deployment manager, read “Migrating WebSphere Application Server, Network Deployment configurations on the z/OS operating system” on page 65 and then return to this article after you migrate the deployment manager.

About this task

Complete this task to migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x federated node to Version 8.0.

Tips:

- For help, read Chapter 11, “Troubleshooting migration,” on page 143.
- For a WebSphere Application Server, Network Deployment cell, the applications are migrated during deployment-manager migration rather than during migration of its federated nodes.
- You can reuse the same target datasets, file system, and Job Control Language (JCL) procedures for your federated nodes that you used for the deployment manager.

Procedure

1. Print the “Migrating a z/OS federated node: Checklist” on page 107, and use it as you conduct your migration.
2. Inventory your applications for the node being migrated, and capture key information as described in “Preparing to migrate a WebSphere Application Server, Network Deployment cell for z/OS” on page 65.
3. Generate a set of migration jobs and instructions.

   Read “Preparing to migrate a federated node using the z/OS Migration Management Tool” and “Processing a migration definition using the z/OS Migration Management Tool” on page 31 for information on generating and uploading your migration definition using the z/OS Migration Management Tool.

   Read “Using the zmmt command to create migration definitions on z/OS” on page 32 for information on generating migration definitions using the zmmt command.
4. Follow the customized instructions in the BBOMMINS member of the CNTL dataset that you generated, and refer to the information in “Migrating a z/OS federated node” on page 108.

What to do next

After you migrate the federated node, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer the migrated federated node.

Preparing to migrate a federated node using the z/OS Migration Management Tool

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload jobs for migrating a federated node.

Before you begin

Print and complete the “Specifying a federated node migration configuration in the z/OS Migration Management Tool: Worksheet” on page 97.

Procedure

1. Follow the instructions in “Creating a migration definition using the z/OS Migration Management Tool” on page 28.
   - On the **Migration Node Type Selection** panel, select **z/OS migrate a federated node**.
   - Fill in the fields on the panels using the values from your “Specifying a federated node migration configuration in the z/OS Migration Management Tool: Worksheet” on page 97.
2. Review the federated node migration definition to make sure that all the values are correct.
   a. In the **WebSphere Application Server for z/OS migration definition** table, select the migration definition that you want to review.
   b. Click **View**.
   c. Click the **Summary**, **Instructions**, or **Response file** tab for information about the migration definition.
3. Upload the migration jobs to the target z/OS system.

   Read “Processing a migration definition using the z/OS Migration Management Tool” on page 31 for instructions.

What to do next

1. To run the migration jobs on the target z/OS system, follow the instructions in the **Migration Instructions** view of the z/OS Migration Management Tool or in the BBOMMINS member of the CNTL dataset that you uploaded to the target z/OS system.

   Read “Migrating a z/OS federated node” on page 108 for details.

2. After you migrate the federated node, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer it.

Configuration variables for migrating a federated node using the z/OS Migration Management Tool:

Before you migrate a WebSphere Application Server for z/OS Version 6.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. The z/OS Migration Management Tool presents you with a set of configuration variables when creating a definition for migrating a federated node.

Migration Node Type Selection

Migration node type
Type of WebSphere Application Server node to migrate

Migration Definition Name

This section identifies the migration definition name and directory path to contain the batch jobs and instructions that will be used to migrate a WebSphere Application Server for z/OS node.

Migration definition name
Name of the z/OS migration definition

This name is used solely on the workstation to identify the migration jobs and instructions that are generated. The name chosen will have no effect on the WebSphere Application Server for z/OS configuration.

Response file path name (optional)
Full path name of a response file that contains default values to be preloaded in the tool

A response file is written each time a z/OS migration definition is created. It contains all of the variable data that was used to create the migration definition, and it can be used to preload the default values when defining a similar migration definition. The response file for a given migration definition is written to the *migration_definition_name*.responseFile file within the root directory for the migration definition.

Normally, you should specify a response file from a migration definition of the same type as that which you are about to define.

Note: A copy of the response file is included in the .DATA dataset that is included in the migration definition that you upload to a z/OS target system. This response file is not used on the z/OS system, but it is there for reference. The member name of the dataset is ZMMTFED.

Target Datasets

High-level qualifier (HLQ)
High-level qualifier for the target z/OS datasets that will contain the generated jobs and instructions

When a z/OS migration definition is uploaded to the target z/OS system, the migration jobs and files are written to a pair of partitioned datasets. While is it possible to reuse these datasets, it is safest to create separate datasets for each z/OS system that is to be migrated.

- **HLQ.CNTL** - a partitioned dataset with fixed block 80-byte records to contain migration jobs
- **HLQ.DATA** - a partitioned dataset with variable length data to contain other data contained in the migration definition

Note: A multilevel high-level qualifier can be specified as the dataset high-level qualifier.

Dataset Names and Product Directory

JCL procedure library dataset name
Existing procedure library to which the WebSphere Application Server for z/OS cataloged procedures are to be copied

WebSphere Application Server product directory
Location of your WebSphere Application Server Version 8.0 installed product file system

Intermediate symbolic link
Select this option to allow to set up an intermediate symbolic link, and specify the path name of that link if you select it.

If you specify an intermediate symbolic link, symbolic links are created from the configuration file system to the intermediate symbolic link; otherwise, they are created directly to the product file system.

Selecting this option will allow you to specify the path name of an intermediate symbolic link. This link will be created by the customization jobs, pointing to the product file system directory.

Path name of intermediate symbolic link
Path name of intermediate symbolic link

Configuration File System

The configuration file system is where the configuration for the node being migrated is physically stored. You can choose to use an existing Version 8.0 file system if you already have an appropriate file system on the node being migrated. If you choose to use an existing Version 8.0 file system, you need to ensure that the mount point that you specify here is present before running the migration utilities (BBOWMG1F, BBOWMG2F, and so on) that are created using this tool. If you choose to create a new Version 8.0 file system on the node being migrated, the actual creation of the new file system will not occur until you run the optional BBOMMHFS or BBOMMZFS job during the actual migration process. In any case, you must specify the correct value here for the mount point.

Refer to the customized instructions generated by this tool for specific information on setting the correct ownership and permissions on the configuration mount point. Read the generated instructions and “Migrating a z/OS federated node” on page 108 for more information on specifying these variables.

Mount point
Read/write file system directory mount point where application data and environment files are written

If this mount point does not already exist, the migration process creates it when you run the optional BBOMMHFS or BBOMMZFS job.

Name File system dataset that you will create and mount at the above mount point

Volume, or '*' for SMS
Specify either the DASD volume serial number to contain the above dataset or "*" to let SMS select a volume.

Using "*" requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle dataset allocation automatically, list the volume explicitly.

Primary allocation in cylinders
Initial size allocation in cylinders for the configuration file system dataset

In an application server, the total space needed for this dataset increases with the size and number of installed applications.

Recommendation: The minimum suggested size is 420 cylinders.

Secondary allocation in cylinders
Size of each secondary extent in cylinders

Recommendation: The minimum suggested size is 100 cylinders.

File system type

Hierarchical File System (HFS)
Allocate and mount your configuration file system using HFS

zSeries File System (ZFS)
Allocate and mount your configuration file system using ZFS

Server Customization (Part 1)

From configuration location

Mount point
Mount point of the configuration from which you are migrating

Home directory
Home directory of the configuration from which you are migrating

To configuration location

Mount point
Mount point of the configuration to which you are migrating

This was specified previously on the **Configuration File System** panel.

Home directory
Home directory of the configuration to which you are migrating

Daemon procedure name
Name of the JCL started procedure used to start the migrated daemon

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the daemon procedure or use the old one.

Controller procedure name
Name of the JCL started procedure used to start the migrated controllers

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the controller procedure or use the old one.

Servant procedure name
Name of the JCL started procedure used to start the migrated servants

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the servant procedure or use the old one.

Adjunct procedure name
Name of the JCL started procedure used to start the migrated adjunct

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the adjunct procedure or use the old one.

Replace started procedure command names
If you specified new names for your JCL procedures, then the corresponding START commands in the WebSphere Application Server configuration must be updated to match the new procedure names. Select this option to perform this configuration update.

If you choose to use the same procedure names, then do not select this option. If you are not using consistent procedure names for all the servers of a given process type (all servants for example) for the node that you are migrating, then it is recommended that you do not select this option. In this case, you will need to keep the same START commands and manually replace the procedures using the procedure that is generated during migration as a template.

Notes:

1. Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x configuration. The migration process creates new Version 8.0 JCL procedures using the procedure names specified.
2. If you use the same names as you used in your Version 6.x configuration, the migration process overlays the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x procedures before running the migration jobs in case you need to roll back later.

WebSphere administrator's user ID

The migration process requires the use of an administrative client. Specify a valid WebSphere administrator's user ID that the administrative client can use to authenticate.

If you want, you can defer entering a user ID at this point and manually edit the BBOWMG3F, BBOWMPRO, BBOWMPRE, or BBOWMPOS jobs later. To do this, update the TO_AdminUserid variable with the proper user ID prior to submitting the job.

WebSphere administrator's password

The migration process requires the use of an administrative client. Specify a valid WebSphere administrator's password that the administrative client can use to authenticate.

If you want, you can defer entering a password at this point and manually edit the BBOWMG3F, BBOWMPRO, BBOWMPRE, or BBOWMPOS jobs later. To do this, update the TO_AdminPassword variable with the proper password before submitting the job.

Server Customization (Part 2)

Migrate to support script compatibility

Whether or not to migrate to support script compatibility

This specifies whether or not migration should create the following Version 6.x configuration definitions:

- Transport
- ProcessDef
- Version 6.x SSL
- Version 6.x ORB service threadpool

instead of the following Version 8.0 configuration definitions:

- Channels
- ProcessDefs
- Version 8.0 SSL
- Version 8.0 ORB service threadpool

Choose to migrate to support script compatibility if you want to minimize impacts to existing administration scripts. If you have existing wsadmin scripts or programs that use third-party configuration APIs to create or modify the Version 6.x configuration definitions, for example, you might want to choose this option.

If you want to use a cell that contains Version 6.x nodes, you must specify true for this variable.

Note: This is meant to provide a temporary transition until all of the nodes in the environment are at the Version 8.0 level. When they are all at the Version 8.0 level, you should perform the following actions:

1. Modify your administration scripts to use all of the Version 8.0 settings.
2. Use the convertScriptCompatability command to convert your configurations to match all of the Version 8.0 settings.

   Read “convertScriptCompatibility command” on page 35 for more information.

Migration trace options

If you choose to enable tracing, it will remain enabled throughout the entire migration process.

Enable script tracing
Enable or disable trace of the home creation, profile and migration tooling invocation, and final processing phases of migration

Enable profile creation tracing
Enable or disable trace trace during profile creation

Enable pre-upgrade tracing
Enable or disable trace during the WASPreUpgrade process

Enable post-upgrade tracing
Enable or disable trace during the WASPostUpgrade process

Migrate administrative console customized "My tasks" settings
"My tasks" is only supported in WebSphere Application Server Version 6.1 and later.

Migrate the settings for "My tasks" saved in the default workspace user root location (wstemp)

Migrate the settings for "My tasks" saved in a user defined workspace root location
User defined workspace root location

Temporary directory location
Directory where the backup of your previous configuration is written as well as the migration trace and temporary file output

During migration, a backup copy of the Version 6.x configuration is required. The default location of this backup is already specified, though you can override if needed. You might need to specify a location other than the default if the `/tmp` file system does not have adequate space to store the backup configuration. If you choose to override the default location of the backup copy, the best practice is to keep the same naming convention and just replace the `/tmp` portion with another path, `/myTemp/migrate` for example.

Migration definition identifier
Identifier that will be used to create a directory under the temporary directory that will contain the temporary migration datasets and backup configuration data

Job Statement Definition

All the migration jobs that will be tailored for you will need a job statement.

Enter a valid job statement for your installation. The migration creation process will update the job name for you in all the generated jobs, so you do not need to be concerned with that portion of the job statement. If continuation lines are needed, replace the comment lines with continuation lines.

Specifying a federated node migration configuration in the z/OS Migration Management Tool: Worksheet:

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the z/OS Migration Management Tool to create a migration definition and upload the appropriate migration jobs. Print this worksheet, and use it when collecting and specifying information about the WebSphere Application Server for z/OS federated node migration configuration variables.

Before you begin

Procedure

1. Specify a value for the **Migration Node Type Selection** panel.

Table 45. Migration Node Type Selection. Enter your values:

Variable	Options	Your value
Migration node type	• z/OS migrate a stand-alone application server • z/OS migrate a deployment manager • z/OS migrate a federated node • z/OS migrate an administrative agent • z/OS migrate a job manager	z/OS migrate a federated node

2. Specify values for the **Migration Definition Name** panel.

Table 46. Migration Definition Name. Enter your values:

Variable	Default value	Your value
Migration definition name	ZMigFed*nn*	
Response file path name (optional)	None	

3. Specify values for the **Target Datasets** panel.

Table 47. Target Datasets. Enter your values:

Variable	Default value	Your value
High-level qualifier (HLQ)	None	

4. Specify values for the **Dataset Names and Product Directory** panel.

Table 48. Dataset Names and Product Directory. Enter your values:

Variable	Default	Your value
JCL procedure library dataset name	SYS1.PROCLIB	
WebSphere Application Server product directory	/usr/lpp/zWebSphere/V8R0	
Intermediate symbolic link		

Table 48. Dataset Names and Product Directory (continued). Enter your values:

Variable			Default	Your value
	Create intermediate symbolic link		Selected	
		Path name of intermediate symbolic link	/wasv8config/ *cell_long_name*/ *node_long_name*/ wasInstall	

5. Specify values for the **Configuration File System** panel.

Table 49. Configuration File System. Enter your values:

Variable		Default value	Your value
Mount point		`/wasv8config`	
Name		`OMVS.WAS80. CONFIG.HFS`	
Volume, or '*' for SMS		`*`	
Primary allocation in cylinders		`420`	
Secondary allocation in cylinders		`100`	
File system type			
	Hierarchical File System (HFS)	Selected	
	zSeries File System (ZFS)	Not selected	

6. Specify values for the **Server Customization (Part 1)** panel.

Table 50. Server Customization (Part 1). Enter your values:

Variable		Default value	Your value
From configuration location			
	Mount point	`/WebSphere/V6R0M0`	
	Home directory	`AppServer`	
To configuration location			
	Mount point	Value from **Configuration File System** panel	
	Home directory	`AppServer`	
Daemon procedure name		`BBO8DMN`	
Controller procedure name		`BBO8ACR`	
Servant procedure name		`BBO8ASR`	
Adjunct procedure name		`BBO8CRA`	
Replace started procedure command names		Selected	
WebSphere administrator's user ID		`XXXXXXXX`	
WebSphere administrator's password		`XXXXXXXX`	

7. Specify values for the **Server Customization (Part 2)** panel.

Table 51. Server Customization (Part 2). Enter your values:

Variable			Default	Your value
Migrate to support script compatibility			Selected	
Migration trace options				
	Enable script tracing		Not selected	
	Enable profile creation tracing		Not selected	
	Enable pre-upgrade tracing		Not selected	
	Enable post-upgrade tracing		Not selected	
Migrate administrative console customized "My tasks" settings				
	Migrate the settings for "My tasks" saved in the default workspace user root location (${USER_INSTALL_ROOT}/wstemp)		Selected	
	Migrate the settings for "My tasks" saved in a user defined workspace root location		Not selected	
		User defined workspace root location	None	
Temporary directory location			`/tmp/migrate`	
Migration definition identifier			Generated value	

8. Specify values for the **Job Statement Definition** panel.

Table 52. Job Statement Definition. Enter your values:

//jobname JOB	(ACCTNO,ROOM),'USERID',CLASS=A,REGION=OM
//*	
//*	
//*	

Preparing to migrate a z/OS federated node using the zmmt command

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.x node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the zmmt command to create jobs for migrating a federated node.

Before you begin

Procedure

1. Print and complete the “Specifying a z/OS federated node migration configuration in the zmmt command response file: Worksheet” on page 105.
2. Follow the instructions in “Using the zmmt command to create migration definitions on z/OS” on page 32.

   Fill in the fields on the panels using the values from your “Specifying a z/OS federated node migration configuration in the zmmt command response file: Worksheet” on page 105.

What to do next

1. Follow the instructions in the BBOMMINS member of the CNTL dataset that you generated using the zmmt command.

   Read “Migrating a z/OS federated node” on page 108 for details.

2. After you migrate the federated node, use the WebSphere Application Server for z/OS Version 8.0 administrative console or scripting to administer it.

Configuration variables for migrating a z/OS federated node using the zmmt command:

Before you migrate WebSphere Application Server for z/OS, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run during the actual migration. You can use the zmmt command with a response file to create the appropriate migration jobs. The migration response file contains a set of configuration variables that you use to create jobs for migrating a federated node.

Response-file variables

create Required keyword to indicate the creation of a new migration definition

profileName
Name of the profile created during migration

This is always "default" on the z/OS platform.

profilePath
Fully qualified path where the generated migration definition output should be written

templatePath
Template path

One of the following values:

- zos-migDmgr
  Migrate a deployment manager
- zos-migFederated
  Migrate a federated node
- zos-migStandalone
  Migrate a stand-alone application server

zConfigHfsName
Name of the MVS dataset that will contain the configuration file system

This name is used as input to the migration job that creates the configuration file system.

zConfigHfsPrimaryCylinders
Number of primary cylinders that will be allocated to the configuration file system

In an application server, the total space needed for this dataset increases with the size and number of installed applications.

Recommendation: The minimum suggested size is 420 cylinders.

zConfigHfsSecondaryCylinders
Number of secondary cylinders that will be allocated to the configuration file system

Recommendation: The minimum suggested size is 100 cylinders.

zConfigHfsVolume
DASD volume serial number to contain the above dataset, or "*" to let SMS select a volume

Using "*" requires that SMS automatic class selection (ACS) routines be in place to select the volume. If you do not have SMS set up to handle dataset allocation automatically, list the volume explicitly.

zConfigMountPoint
File system directory mount point where application data and environment files are written

zFilesystemType
Type of file system

One of the following values:
- HFS

  Allocate and mount your configuration file system dataset using the Hierarchical File System
- ZFS

  Allocate and mount your configuration file system dataset using the zSeries File System

zJobStatement1 . . . *n*
Customizable JOB statements that will be used for the generated migration jobs

All the migration jobs that will be tailored for you will need a job statement. Enter a valid job statement for your installation. The migration creation process will update the job name for you in all the generated jobs, so you do not need to be concerned with that portion of the job statement. If continuation lines are needed, replace the comment lines with continuation lines.

zTargetHLQ
High-level qualifier for the target z/OS datasets that will contain the generated jobs and instructions

Note: A multilevel high-level qualifier can be specified as the dataset high-level qualifier.

zmbAdminUserid
User ID of an administrator that is used to manage the node that is being migrated

This is required to perform required administrative actions during the migration process.

zmbAdminPassword
Password for the user ID of the administrator that is used to manage the node that is being migrated

This is required to perform required administrative actions during the migration process.

zmbControllerProcName
Name of the JCL started procedure used to start the migrated controllers

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the controller procedure or use the old one.

zmbServantProcName
Name of the JCL started procedure used to start the migrated servants

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the servant procedure or use the old one.

zmbDaemonProcName
Name of the JCL started procedure used to start the migrated daemon

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the daemon procedure or use the old one.

zmbAdjunctProcName
Name of the JCL started procedure used to start the migrated adjunct

When migrating to Version 8.0, you must upgrade your JCL started procedures. A new started procedure will be generated for you during migration. You can specify a new name for the adjunct procedure or use the old one.

zmbReplaceStartedProcedureNames

Whether to update the START commands in the configuration with the new names specified (true) or to preserve the same names (false).

If you specified new names for your JCL procedures, then the corresponding START commands in the WebSphere Application Server configuration must be updated to match the new procedure names. Specify true for this variable to perform this configuration update.

If you choose to use the same procedure names, then specify false for this variable. If you are not using consistent procedure names for all the servers of a given process type (all servants for example) for the node that you are migrating, then it is recommended that you specify false for this variable. In this case, you will need to keep the same START commands and manually replace the procedures using the procedure that is generated during migration as a template.

Notes:

- Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x configuration. The migration process creates new Version 8.0 JCL procedures using the procedure names specified here.
- If you use the same names as you used in your Version 6.x configuration, the migration process overlays the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x procedures before running the migration jobs in case you need to roll back later.

zmbEnablePostUpgradeTrace

Enable (true) or disable (false) trace during the WASPostUpgrade process

zmbEnablePreUpgradeTrace

Enable (true) or disable (false) trace during the WASPreUpgrade process

zmbEnableProfileTrace

Enable (true) or disable (false) trace during profile creation

zmbEnableScriptingTrace

Enable (true) or disable (false) trace of the home creation, profile and migration tooling invocation, and final processing phases of migration

zmbEnableScriptCompatibility

Whether (true) or not (false) to migrate to support script compatibility

This specifies whether or not migration should create the following Version 6.x configuration definitions:

- Transport
- ProcessDef
- Version 6.x SSL
- Version 6.x ORB service threadpool

instead of the following Version 8.0 configuration definitions:

- Channels
- ProcessDefs
- Version 8.0 SSL
- Version 8.0 ORB service threadpool

Choose to migrate to support script compatibility if you want to minimize impacts to existing administration scripts. If you have existing wsadmin scripts or programs that use third-party configuration APIs to create or modify the Version 6.x configuration definitions, for example, you might want to specify true for this variable.

If you want to use a cell that contains Version 6.x nodes, you must specify true for this variable.

Note: This is meant to provide a temporary transition until all of the nodes in the environment are at the Version 8.0 level. When they are all at the Version 8.0 level, you should perform the following actions:

1. Modify your administration scripts to use all of the Version 8.0 settings.
2. Use the convertScriptCompatability command to convert your configurations to match all of the Version 8.0 settings.

   Read “convertScriptCompatibility command” on page 35 for more information.

zmbFromConfigRoot
Mount point of the configuration from which you are migrating

zmbFromWASHomeDir
Home directory of the configuration from which you are migrating

zmbProclibName
Existing procedure library to which the WebSphere Application Server for z/OS cataloged procedures are to be copied

zmbSMPEHome
Location of your WebSphere Application Server Version 8.0 SMPE installed product file system

zmbTempDirectory
Directory where the backup of your previous configuration is written as well as the migration trace and temporary file output

During migration, a backup copy of the Version 6.x configuration is required. The default location of this backup is already specified, though you can override if needed. You might need to specify a location other than the default if the `/tmp` file system does not have adequate space to store the backup configuration. If you choose to override the default location of the backup copy, the best practice is to keep the same naming convention and just replace the `/tmp` portion with another path, `/myTemp/migrate` for example.

zmbTimestamp
Identifier that will be used to create a directory under the temporary directory that will contain the temporary migration datasets and backup configuration data

zmbToConfigRoot
Mount point of the configuration to which you are migrating

This is the same value that is specified for the zConfigMountPoint variable.

zmbToWASHomeDir
Home directory of the configuration to which you are migrating

zmbWorkspaceRootPreference
Whether to migrate the administrative console customized "My tasks" settings saved in the default workspace user root location (D) or to migrate the settings saved in a user-defined workspace root location (U)

zmbUserWorkspaceRoot
User-defined workspace root location

intermediateSymlinkPreference
Whether to set up an intermediate symbolic link

If you specify an intermediate symbolic link, symbolic links are created from the configuration file system to the intermediate symbolic link; otherwise, they are created directly to the product file system.

IntermediateSymlink
Path name of intermediate symbolic link

This link will be created by the customization jobs, pointing to the product file system directory.

Sample of a response file for migrating a federated node

```
create
profileName=default
profilePath=/zmmt/workspace/.metadata/.plugins/com.ibm.ws390.mmt.config/profiles/ZMigFed01
templatePath=zos-migFederated
zConfigHfsName=OMVS.WAS80.CONFIG.HFS
zConfigHfsPrimaryCylinders=420
zConfigHfsSecondaryCylinders=100
zConfigHfsVolume=*
zConfigMountPoint=/wasv8config
zFilesystemType=HFS
zJobStatement1=(ACCTNO,ROOM),'USERID',CLASS=A,REGION=0M
zJobStatement2=//*
zJobStatement3=//*
zJobStatement4=//*
zTargetHLQ=FED
zmbAdjunctProcName=BBO7CRA
zmbAdminPassword=XXXXXXXX
zmbAdminUserid=XXXXXXXX
zmbControllerProcName=BBO7ACR
zmbDaemonProcName=BBO7DMN
zmbEnablePostUpgradeTrace=false
zmbEnablePreUpgradeTrace=true
zmbEnableProfileTrace=false
zmbEnableScriptCompatibility=true
zmbEnableScriptingTrace=true
zmbFromConfigRoot=/WebSphere/V5R1M0
zmbFromWASHomeDir=AppServer
zmbProclibName=SYS1.PROCLIB
zmbReplaceStartedProcedureNames=true
zmbSMPEHome=/usr/lpp/zWebSphere/Version 8.0
zmbServantProcName=BBO7ASR
zmbTempDirectory=/tmp/migrate
zmbTimestamp=30083908
zmbToConfigRoot=/wasv8config
zmbToWASHomeDir=AppServer
zmbWorkspaceRootPreference=D
intermediateSymlinkPreference=Y
IntermediateSymlink=/V80SMPE/symblink
```

Specifying a z/OS federated node migration configuration in the zmmt command response file: Worksheet:

Before you migrate a WebSphere Application Server for z/OS Version 6.x or Version 7.0 node to Version 8.0, you must create Job Control Language (JCL) jobs (CNTL and DATA datasets) that you will run on z/OS during the actual migration. You can use the zmmt command with a response file to create the appropriate migration jobs. Print this worksheet, and use it when collecting and specifying information about the WebSphere Application Server for z/OS federated node migration configuration variables in the zmmt command response file.

Before you begin

Procedure

1. Specify "create."
2. Specify values for the name and location.

Table 53. Name and location. Enter your values:

Variable	Response-file variable	Example value	Your value
Profile name	profileName	Default	

Table 53. Name and location (continued). Enter your values:

Variable	Response-file variable	Example value	Your value
Profile directory	profilePath	/AppServer/profiles	
Template path	templatePath	zos-migFederated	

3. Specify values for the target datasets.

Table 54. Target datasets. Enter your values:

Variable	Response-file variable	Example value	Your value
High-level qualifier (HLQ)	zTargetHLQ	FED	

4. Specify values for the dataset names, product directory, and intermediate symbolic link.

Table 55. Dataset names, product directory, and intermediate symbolic link. Enter your values:

Variable			Response-file variable	Example value	Your value
JCL procedure library dataset name			zmbProclibName	SYS1.PROCLIB	
WebSphere Application Server product directory			zmbSMPEHome	/usr/lpp/zWebSphere/V7R0	
Intermediate symbolic link					
	Create intermediate symbolic link		intermediateSymlinkPreference	Y	
		Path name of intermediate symbolic link	IntermediateSymlink	/V70SMPE/symblink	

5. Specify values for the configuration file system.

Table 56. Configuration file system. Enter your values:

Variable		Response-file variable	Example value	Your value
Mount point		zConfigMountPoint	/wasv7config	
Name		zConfigHfsName	OMVS.WAS70. CONFIG.HFS	
Volume, or '*' for SMS		zConfigHfsVolume	*	
Primary allocation in cylinders		zConfigHfsPrimaryCylinders	420	
Secondary allocation in cylinders		zConfigHfsSecondaryCylinders	100	
File system type				
	Hierarchical File System (HFS)	zFilesystemType	HFS	
	zSeries File System (ZFS)			

6. Specify values for server customization.

Table 57. Server customization. Enter your values:

Variable		Response-file variable	Example value	Your value
From configuration location				
	Mount point	zmbFromConfigRoot	/WebSphere/V5R1M0	
	Home directory	zmbFromWASHomeDir	AppServer	
To configuration location				
	Mount point	zmbToConfigRoot	/wasv7config	
	Home directory	zmbToWASHomeDir	AppServer	
Daemon procedure name		zmbDaemonProcName	BBO7DMN	
Controller procedure name		zmbControllerProcName	BBO7ACR	
Servant procedure name		zmbServantProcName	BBO7ASR	
Adjunct procedure name		zmbAdjunctProcName	BBO7CRA	
Replace started procedure command names		zmbReplaceStartedProcedure Names	True	
WebSphere administrator's user ID		zmbAdminUserid	XXXXXXXX	
WebSphere administrator's password		zmbAdminPassword	XXXXXXXX	

Table 58. Server customization. Enter your values:

Variable			Response-file variable	Example value	Your value
Migrate to support script compatibility			zmbEnableScriptCompatibility	True	
Migration trace options					
	Enable script tracing		zmbEnableScriptingTrace	False	
	Enable profile creation tracing		zmbEnableProfileTrace	False	
	Enable pre-upgrade tracing		zmbEnablePreUpgradeTrace	False	
	Enable post-upgrade tracing		zmbEnablePostUpgradeTrace	False	
Migrate administrative console customized "My tasks" settings					
	Migrate the settings for "My tasks" saved in the default workspace user root location (D)		zmbWorkspaceRoot Preference	D	
	Migrate the settings for "My tasks" saved in a user defined workspace root location (U)				
		User defined workspace root location	zmbUserWorkspaceRoot		
Temporary directory location			zmbTempDirectory	/tmp/migrate	
Migration definition identifier			zmbTimestamp	21130546	

7. Specify values for job statement definition.

Table 59. Job statement definition. Enter your values:

Variable	Response-file variable	Example value	Your value
//jobname JOB	zJobStatement1	(ACCTNO,ROOM),'USERID', CLASS=A,REGION=OM	
//*	zJobStatement2	//*	
//*	zJobStatement3	//*	
//*	zJobStatement4	//*	

Migrating a z/OS federated node: Checklist

Activities for migrating a federated node to WebSphere Application Server Version 8.0 after you generate the appropriate Job Control Language (JCL) jobs include activities related to general preparation, transaction-log clearance, and node migration. Use the following checklist as a quick reference to these activities. Follow the instructions that were generated by the z/OS Migration Management Tool or the zmmt command to complete the process of migrating your federated node to Version 8.0.

Before you begin

You can find more details about migrating a federated node in “Preparing to migrate a WebSphere Application Server, Network Deployment cell for z/OS” on page 65 and “Migrating a z/OS federated node” on page 108.

Procedure

1. Prepare to migrate the federated node.

Table 60. Migrated federated node checklist. Enter your values:

Check off	Item
_____	Make sure that you have a minimum level of Version 7.0 maintenance. Read “Overview of migration, coexistence, and interoperability” on page 1 for more information.
_____	Create CNTL and DATA datasets for each node to be migrated. Read “Preparing to migrate a federated node using the z/OS Migration Management Tool” on page 92 or “Preparing to migrate a z/OS federated node using the zmmt command” on page 100 for more information.
_____	Create the Version 7.0 configuration mount point; allocate a configuration file system, and mount either manually or using the BBOMMHFS or BBOMMZFS job.

2. Clear the transaction logs for the federated node.

Table 61. Transaction logs checklist. Enter your values:

Check off	Item
_____	Stop the federated application server.
_____	Run the BBOWMG1F job, and check for success.
_____	Restart the federated application server to clear the transaction logs.
_____	Run the BBOWMG2F job to disable Peer Restart and Recovery (PRR) mode.

3. Prepare the procedures.

Table 62. Procedures checklist. Enter your values:

Check off	Item
_____	Run the BBOMMCP job to copy the generated JCL procedures to your procedure library.
_____	If you specified new procedure names, update your RACF STARTED profiles for the controller and daemon.

4. Migrate the federated node.

Table 63. Federated node checklist. Enter your values:

Check off	Item
_____	Ensure that all servers are stopped, including the daemon server.
_____	Ensure that the deployment manager is running at a Version 7.0 level. If the Version 7.0 deployment manager is not running, start it.
_____	Run the migration jobs—including the BBOWMG3F job or the BBOWMPRO, BBOWMPRE, and BBOWMPOS jobs—by following the generated instructions; and check for success.
_____	Start the federated application server.

Migrating a z/OS federated node

After the deployment manager is migrated and restarted, you can perform the actual migration of its federated application server nodes by running the Job Control Language (JCL) jobs that you generated. When you generated your custom migration jobs, you also created customized instructions for preparing and running the migration jobs in the BBOMMINS members of the CNTL dataset that was used to generate your jobs. Follow these customized instructions to complete the process of migrating your federated nodes to Version 8.0.

Before you begin

- Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6.
- You will not be able to proceed if you did not generate the JCL migration jobs.
- The BBOWMG1F, BBOWMG2F, and BBOWMG3F jobs referenced in these instructions must be submitted by a WebSphere Administrator user ID.

  All other jobs must be submitted by a user ID that has control over the file system.
- To migrate a federated node on another MVS image, make sure that you run the jobs on the same system on which the node itself resides.
- **Tip:** When migrating a WebSphere Application Server Version 6.x or Version 7.x federated node, you must perform the following actions if you want to be able to roll it back to its previous state after migration:
  1. Back up your existing configuration using the backupConfig command or your own preferred backup utility.
     – Run the backupConfig command or your own preferred utility to back up the Version 8.0 deployment manager configuration.

       Important: Make sure that you note the exact name and location of this backed-up configuration.

       Read the "backupConfig command" article in the information center for more information.

– Run the backupConfig command or your own preferred utility to back up the Version 6.x or Version 7.x federated node configuration.

Important: Make sure that you note the exact name and location of this backed-up configuration.

Read the "backupConfig command" article in the information center for more information.

2. Migrate the federated node.

If necessary, you can now roll back the federated node that you just migrated. Read “Rolling back a federated node” on page 119 for more information.

For help, read Chapter 11, “Troubleshooting migration,” on page 143.

Procedure

1. Ensure that the application servers and the node agent are stopped for the federated node being migrated.

   You must stop the Version 6.x or Version 7.x federated node before proceeding.

2. Ensure that the newly migrated deployment manager is up and running.

   For the application server node to be properly migrated, the deployment manager must be running. For this migration to work, the deployment manager must be up and listening on its SOAP port.

Table 64. Deployment manager is running. Check when complete

Check Off	Item
	Access the administrative console of the WebSphere Application Server Version 8.0 deployment manager. This validates that the deployment manager is running.

Table 65. Application server is running. Check when complete

Check Off	Item
	Ensure that the WebSphere Application Server Version 8.0 copy of the code is running. Under "About your WebSphere Application Server," the build number should begin with `W80`.

3. Create and mount a new Version 8.0 configuration file system.

   Before you perform the migration, Version 8.0 requires that a configuration file system be present for your new configuration. You can run BBOMMHFS or BBOMMZFS to create and mount a new configuration file system, or you can mount one manually. Either way, you must have a file system for your Version 8.0 configuration created and mounted before you proceed. This configuration file system is the target of the migration; your Version 6.x or Version 7.x configuration file system is the source.

   BBOMMHFS or BBOMMZFS creates a mount-point directory, allocates the configuration's file system, and mounts the file system at whatever value you specified for the mount point when you generated your migration jobs.

   Before you proceed, ensure that you have allocated, created, and mounted your configuration file system datasets either manually or using BBOMMHFS or BBOMMZFS. The mount point should be owned by the WebSphere Administrator user ID and have permissions of at least 755. The new configuration file system structures should be included in BPXPARM so that they will be mounted at the next IPL.

4. Submit BBOWMG1F and BBOWMG2F.

   Note: If you are not using XA connectors, submitting BBOWMG1F and BBOWMG2F is optional. However, you should submit both jobs to ensure that your transaction logs are clear.

BBOWMG1F enables all servers on the federated application server node being migrated to start in Peer Restart and Recovery (PRR) mode. PRR processing mode resolves any outstanding transactions, clears the transaction logs, and stops the server. BBOWMG2F disables PRR mode and returns all servers to normal operating state.

Follow these steps to clear the XA transaction logs:

a. Stop the application server.

b. Submit the job BBOWMG1F, and verify a return code of 0.

c. Restart the federated application server, and wait for it to perform PRR processing and stop automatically.

   Tip: After the transactions are resolved and the server stops, a nonzero return code is normal and expected. The following is an example of an acceptable result code from the server process:

   ```
   BBOO0035W TERMINATING THE CURRENT PROCESS, REASON=C9C218D5
   ```

d. Submit the job BBOWMG2F, and verify a return code of 0.

5. Copy your generated JCL procedures.

   The migration utility BBOMMCP copies the generated JCL procedures to start the servers to the specified procedure library. Your Version 8.0 configuration must use different JCL procedures from those used by your Version 6.x or Version 7.x configuration. This utility will update the new Version 8.0 configuration, substituting your new JCL names in place of the names that existed in your original Version 6.x or Version 7.x configuration.

   Caution: This utility copies the generated JCL to your procedure library. If you specified the same names as you used in your Version 6.x or Version 7.x configuration when you generated your migration jobs, this utility will overlay the existing procedures. If you are using the same names, make sure that you back up the existing Version 6.x or Version 7.x procedures before running this utility in case you need to roll back later.

   Submit BBOMMCP, and verify a return code of 0.

6. If you specified new procedure names, update your RACF STARTED profiles for the controller and daemon.

   The STARTED profile used by controller regions is based on the procedure name and JOBNAME. You must ensure that a STARTED profile will apply so that the proper identity will be assigned to the started task. If your Version 6.x or Version 7.x controller JCL procedure name is AZACR and you specified AZ1ACR for Version 8, for example, then you would need to create a STARTED profile for that new procedure name:

   ```
                new controller      same identity used in
                   JCL name         V6.x or V7.x configuration
                      |                        |
   RDEFINE STARTED AZ1ACR.* STDATA(USER(AZACRU) GROUP(AZCFG) TRACE(YES))
   ```

   Note:

   - Do not use a different user ID to start. There are other things tied to the user ID, and other changes would also be required if you change the user ID.
   - If your original STARTED profile was generic, STARTED AZ*.* ... for example, you would not need to create a new STARTED profile.
   - Servant region STARTED profiles are based on JOBNAME, not procedure name. So there is no issue with the servant when you use a different procedure name.
   - Daemons and node agents are controllers; so, using different procedure names for these implies a new STARTED profile.

7. Delete and redefine the log stream if necessary.

   Perform this step only if you previously configured the transaction XA partner log or compensation log on the Version 6.x or Version 7.x server to use a log stream.

a. Make sure that the node is stopped.

b. Delete and redefine the log stream.

You can use the BBOLOGSD and BBOLOGSA jobs that were created during Version 6.x or Version 7.x customization if you configured the server initially to use the log stream.

The following sample shows an example of such a job:

```
//RLSP1A  JOB 'xxxx,yyy,?','USERID',MSGCLASS=H,
//         CLASS=J,MSGLEVEL=(1,1),REGION=4M,NOTIFY=&SYSUID
//STEP1    EXEC PGM=IXCMIAPU
//STEPLIB  DD   DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *

DATA TYPE(LOGR) REPORT(YES) /* Default to show output of job */
 DELETE LOGSTREAM NAME(P1ACEL6A.W51ASA2.D)
 DEFINE LOGSTREAM NAME(P1ACEL6A.W51ASA2.D)
        LOWOFFLOAD(20)
        HIGHOFFLOAD(79)
        STG_DUPLEX(YES)
        DUPLEXMODE(UNCOND)
        STG_DATACLAS(OPERLOG)
        STG_SIZE(5000)
        HLQ(Q10RRS)
        LS_SIZE(5000)
        LS_DATACLAS(OPERLOG)
        STRUCTNAME(WAS_LOGRLS)
/*
```

If you are migrating nodes in a sysplex, follow this procedure for each federated node that you migrate.

8. Perform one of the following actions:

a. Submit the BBOWMG3F job.

Attention: When running in an z/OS environment and configured with multiple TCP/IP stacks, you may need to add the environment variable _BPXK_SETIBMOPT_TRANSPORT to direct the migration job to the proper TCP/IP stack that should be used. If you use the incorrect stack, the result leads to a `java.net.UnknownHostException` that causes the subsequent wsadmin login to fail.

An `export _BPXK_SETIBMOPT_TRANSPORT=<stack.name>` statement is required in the JCL to identify the appropriate stack. Your JCL might look something like this:

```
//*************************************************************
//*
//* UPGRADE: Perform the migration to the new Profile
//*
//*************************************************************
//*
//*
//UPGRADE EXEC PGM=IKJEFT01,REGION=0M,COND=(4,LE)
//SYSTSPRT DD SYSOUT=*
//STDENV DD * // _CEE_RUNOPTS=TRAP(ON,NOSPIE) //*
//SYSTSIN  DD *
  BPXBATCH SH +
  export _BPXK_SETIBMOPT_TRANSPORT=TCP; +
  /tmp/migrate/bbomigrt2.sh WASPostUpgrade +
      /tmp/migrate/28133744/_      +
  1>> /tmp/migrate/28133744/BBOWMG3F.out +
  2>> /tmp/migrate/28133744/BBOWMG3F.err;
/*
```

BBOWMG3F is the job that performs the physical migration of the Version 6.x or Version 7.x node to Version 8.0 based on the information that you supplied when you generated your migration jobs. Submit BBOWMG3F, verify that you are getting return codes of 0, and review the log files in the temporary migration directory on the configuration file system. The migration temporary

directory is *temporary_directory_location*/*nnnnn*, where *temporary_directory_location* is the directory specified for the temporary directory location (/tmp/migrate by default) and *nnnnn* is the numeric value generated for the migration identifier when you generated your migration jobs.

b. Submit the following three jobs:

1) Submit the BBOWMPRO job.

BBOWMPRO creates the Websphere Application Server home and default profile.

2) Submit the BBOWMPRE job.

BBOWMPRE runs the migration pre-upgrade process.

3) Submit the BBOWMPOS job.

BBOWMPOS runs the migration post-upgrade and finish-up (change file permission) processes.

9. Ensure that the old daemon is stopped.

Ensure that all federated nodes in the same cell on this LPAR are stopped.

10. Update the daemon JCL procedure if necessary.

WebSphere Application Server for z/OS Version 8.0 requires that the daemon process be at the highest level of code of any of the servers that it manages on the same LPAR. It will be at the Version 8.0 level when this federated node is started.

After you migrate all nodes to Version 8.0 and before you remove the previous version's libraries from the system, you must update the daemon JCL procedure. Failure to do so will result in a failure of the daemon to start.

Note:

- If you are migrating from Version 5.1.x to Version 8.0, the daemon needs to have a STEPLIB that includes the Version 5.1.x SBBOLD2 and SBBOLPA datasets.
- If you are migrating from Version 6.0.1.x to Version 8.0, the daemon needs to have a STEPLIB that includes the Version 6.0.1.x SBBOLD2 and SBBOLPA datasets.
- If a Version 8.0 cell has a Version 8.0 server that is communicating with a Version 6.0.1.x server in a Version 6.0.1.x cell that is on the same system as the Version 8.0 cell, you also need to add the version 6.0.1.x SBBOLD2 and SBBOLPA datasets to the STEPLIB of the Version 8.0 daemon.

11. Start the new federated application server node.

a. Start the node agent.

The following message is displayed on the console and in the job log of BBON001:

```
BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR z/OS CONTROL PROCESS BBON001
```

b. Start the federated application server.

Use the existing command that you would use to start a Version 6.x or Version 7.x application server, but replace the RACF STARTED procedure name with the value that you entered in the federated node panel for the controller procedure name when you generated your migration jobs. This command starts the Version 8.0 federated application server. Wait until the server is finished initializing before proceeding.

The following message is displayed on the console and in the job log of BBOS001:

```
BBOO0019I INITIALIZATION COMPLETE FOR WEBSPHERE FOR z/OS CONTROL PROCESS BBOS001
```

At this point, your migration to Version 8.0 is complete.

What to do next

After you verify a successful migration to Version 8.0 and are successfully running a migrated configuration, you should delete the following items:

- Everything in the source configuration's file system

- Everything in the target configuration's *temporary_directory_location*/*nnnnn* directory, where *temporary_directory_location* is the directory specified for the temporary directory location (/tmp/migrate by default) and *nnnnn* is the numeric value specified for the migration identifier when you created your migration jobs
- The bbomigrt2.sh file

Migrating IBM Cloudscape or Apache Derby databases

The migration tools migrate any IBM Cloudscape database instances to Apache Derby instances in the new configuration, and they copy any Apache Derby instances that are stored in the previous release's WebSphere Application Server configuration tree to the new release's configuration tree. After you use the migration tools, you should verify the results of the database migration and manually migrate any Cloudscape database instances or copy any Derby database instances that are not automatically migrated or copied by the tools.

Before you begin

Read "Overview of migration, coexistence, and interoperability" on page 1 and "Premigration considerations" on page 6. For resources to help you plan and perform your migration, visit Knowledge Collection: Migration planning for WebSphere Application Server.

Tips:

- Before you run the migration tools, ensure that any application servers hosting applications that are using a Cloudscape or a Derby database are closed.

  Otherwise, the database migration will fail.
- Before you run the migration tools, ensure that the *debug migration trace* is active.

  By default, this trace function is enabled. To reactivate the debug migration trace if it is disabled, set one of the following trace options:

  - all traces*=all
  - com.ibm.ws.migration.WASUpgrade=all

About this task

WebSphere Application Server Version 8.0 requires Apache Derby Version 10.3 or later. Apache Derby Version 10.3 is a pure Java database server that combines the Derby runtime with the opportunity to use the full services of IBM Software Support. For comprehensive information about Apache Derby Version 10.3, read the Apache Derby website.

For help, read Chapter 11, "Troubleshooting migration," on page 143.

Important: Derby-to-Derby migration performs a file-system copy of the data at a given point in time. This snapshot will not remain in sync with the database in the previous installation. If you roll back to the previous release, any updates to the database that you made after migration will not be reflected in the previous installation.

Procedure

1. Migrate the configuration to Version 8.0.
2. Verify the automatic migration of Cloudscape database instances or copying of Derby database instances.

   When you migrate from WebSphere Application Server Version 6.x to Version 8.0, the migration tools automatically upgrade Cloudscape or Derby database instances that are accessed through the embedded framework by some internal components such as the UDDI registry. The tools also attempt

to upgrade Cloudscape or Derby instances that your applications access through the embedded framework. You must verify these migration results after running the migration tools.

- To distinguish between a partially and a completely successful Cloudscape-to-Derby migration, verify the automatic-migration results by performing the following tasks:
  a. Check the general migration post-upgrade log for database error messages.

     These exceptions indicate database migration failures. The migration tool references all database exceptions with the prefix `DSRA`.

  b. Check the individual database migration logs.

     These logs have the same timestamp as that of the general migration post-upgrade log. The individual logs display more detail about errors that are listed in the general post-upgrade log as well as expose errors that are not documented by the general log.

     The path name of each database log is *app_server_root*`/profiles/`*`profileName`*`/logs/`*`myFulldbPathName`*`_migrationLog`*`timestamp`*`.log`.

  c. Look at the debug log that corresponds with the database migration log.

     The WebSphere Application Server migration utility triggers a *debug migration trace* by default; this trace function generates the database debug logs.

     The full path name of each debug log is *app_server_root*`/profiles/`*`profileName`*`/logs/`*`myFulldbPathName`*`_migrationDebug`*`timestamp`*`.log`.

  Performing these tasks gives you vital diagnostic data to troubleshoot the partially migrated databases as well as those that fail automatic migration completely. Ultimately, you must migrate databases that were not completely migrated automatically through a manual process. The log messages contain the exact old and new database path names that you must use to run the manual migration. Note these new path names precisely.

  Read the "Verifying the Cloudscape automatic migration" article in the information center for more information.

- Verify that any Derby database instances that are stored in the previous release's WebSphere Application Server configuration tree were copied to the new release's configuration tree

  Check the general migration post-upgrade log for database error messages. These exceptions indicate database migration failures. The migration tool references all database exceptions with the prefix `DSRA`..

3. Manually migrate Cloudscape database instances or copy Derby database instances where necessary.
   - The Version 8.0 migration tools do not attempt to migrate database instances that transact with applications through the Cloudscape Network Server or the Apache Derby Network Server framework. This exclusion eliminates the risk of corrupting third-party applications that access the same database instances as those accessed by WebSphere Application Server.

     To minimize the risk of migration errors for databases that were only partially upgraded during the automatic migration process, delete the new database. Troubleshoot the original database according to the log diagnostic data, then perform manual migration of the original database.

     Read the "Upgrading Cloudscape manually" article in the information center for more information.

   - The Version 8.0 migration tools do not copy any Derby database instances outside the WebSphere Application Server configuration tree.

     If migration does not copy a Derby database instance automatically, copy the database instance manually.

4. Manually migrate your UDDI registry if it uses a database on the Cloudscape Network Server or the Apache Derby Network Server framework.

   Read the "Migrating the UDDI registry" article in the information center for more information.

What to do next

Service integration bus-enabled web services use a Service Data Objects (SDO) repository for storing and serving WSDL definitions. If you migrate a configuration that uses a Cloudscape database as the SDO repository, the SDO application will still be configured to use Cloudscape in the new configuration. Migration converts the Cloudscape database to Derby, but you must still update any SDO application's backend ID to use the new database. After you migrate all of the nodes on a server with an SDO repository application that uses Cloudscape, perform the following actions to reset the database type used by the SDO application on the new configuration to Derby:

1. Read about the basic usage for the installSdoRepository.jacl script inside the script file.
2. Run the installSdoRepository.jacl script by changing to the *app_server_root*/bin/ directory and running the following command:

   ```
   wsadmin.extension -f app_server_root/bin/installSdoRepository.jacl -editBackendId DERBY_V10
   ```

Read the "Installing and configuring the SDO repository" article in the information center for more information on upgrading the SDO repository application to Version 8.0 .

Migrating from the WebSphere Connect JDBC driver

WebSphere Application Server Version 8.0 does not include the WebSphere Connect JDBC driver for SQL Server. Use the WebSphereConnectJDBCDriverConversion command to convert data sources from the WebSphere Connect JDBC driver to the DataDirect Connect JDBC driver or the Microsoft SQL Server JDBC driver. The WebSphereConnectJDBCDriverConversion command processes `resources.xml` files, and there are many options that can be specified to indicate which `resources.xml` files to process.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6.

About this task

gotcha: If you are running in a mixed node environment, you must issue this command for each node, specifying a specific cell name for the cellName parameter, and a specific node name for the nodeName parameter. If you do not issue these commands, you will get the following exception for your back-level nodes:

```
IGR0480E: The node node–name specified for the -nodeName parameter
is not at the current release. Migrate this node to the current release and rerun the tool.
```

Syntax

```
WebSphereConnectJDBCDriverConversion.sh
   [-profileName profile_name]
   [-driverType MS | DD]
   [-classPath class_path]
   [-nativePath native_path]
   [-pathSeparator separator]
   [[-cellName ALL | cell_name [-clusterName ALL | cluster_name] |
     [-applicationName ALL | application_name] |
     [-nodeName ALL | node_name] [-serverName ALL | server_name]]]
   [-backupConfig true | false]
   [-username userID]
   [-password password]
   [[-traceString trace_spec [-traceFile file_name]]
```

Parameters

-profileName *profile_name*
This optional parameter is used to specify a specific profile configuration in the Version 8.0 environment.

If this parameter is not specified, the default profile is used. If the default is not set or cannot be found, an error is returned.

If the command is run from within the *profile_name*/bin directory, it is implicit that the *profile_name* profile should be migrated and this parameter is not needed.

-driverType MS | DD
This required parameter is used to indicate the type of conversion that you want to perform.

Specifying MS converts to the Microsoft SQL Server JDBC Driver, and specifying DD converts to the DataDirect Connect JDBC driver.

-classPath *class_path*
This required parameter is used to specify the class path for the new JDBC driver.

-nativePath *native_path*
This optional parameter is used to specify the native path for the new JDBC driver.

-pathSeparator *separator*
This optional parameter is used to specify a path separator other than the default.

The default is operating-system dependant and is listed in the command-line help.

-cellName ALL | *cell_name*
This optional parameter is used to specify the specific name of the cell to process or to specify all cells.

The default is ALL. If ALL is specified, the values specified for the nodeName, serverName, applicationName, and clusterName parameters are ignored, and the settings for these parameters revert to their default values of ALL.

-clusterName ALL | *cluster_name*
This optional parameter is used to specify the specific name of the cluster to process or to specify all clusters.

The default is ALL

-application nameName ALL | *application_name*
This optional parameter is used to specify applications to process. (Some resource.xml files might exist in applications if enhanced EAR file support is used.)

The default is ALL

-nodeName ALL | *node_name*
This optional parameter is used to specify the specific name of the node to process or to specify all node names in the configuration. If you specify a specific nodeName, you must also specify a specific cell name on the cellName parameter. If a specific cell name is not specified, the default value of ALL is used for this parameter.

If ALL is specified for this parameter, the value specified for the serverName parameter is ignored, and the setting for the serverName parameter reverts to its default value of ALL.

-serverName ALL | *server_name*
This optional parameter is used to specify the specific name of a server or to specify all server names in a node.

The default is ALL.

-backupConfig true | false
This parameter is used to specify whether (true) or not (false) to back up the existing configuration before the command makes changes to the configuration.

The default is true.

-username *userID*
This optional parameter is used to specify the user name to be used by this command.

-password *password*
This optional parameter is used to specify the password to be used by this command.

-traceString *trace_spec*
This optional parameter is used with -traceFile to gather trace information for use by IBM service personnel.

The value of traceString is "*=all=enabled" and must be specified with quotation marks to be processed correctly.

-traceFile *file_name*
This optional parameter is used with -traceString to gather trace information for use by IBM service personnel.

Rolling back environments

After migrating to a WebSphere Application Server Version 8.0 environment, you can roll back to a Version 6.x or 7.x environment. This returns the configuration to the state that it was in before migration. After rolling back the environment, you can restart the migration process.

Before you begin

About this task

Generally, migration does not modify anything in the configuration of the prior release; however, there are cases where minimal changes are made that are reversible.

Procedure

- To roll back a Version 8.0 deployment manager and its federated nodes to Version 6.x or 7.x, follow the instructions in “Rolling back a WebSphere Application Server, Network Deployment cell.”
- To roll back a Version 8.0 federated node to Version 6.x or 7.x, follow the instructions in “Rolling back a federated node” on page 119.
- To roll back a Version 8.0 stand-alone application server to Version 6.x or 7.x, follow the instructions in “Rolling back stand-alone application servers” on page 121.

Results

The configuration should now be returned to the state that it was in before migration.

What to do next

You can now restart the migration process if you want to do so.

Rolling back a WebSphere Application Server, Network Deployment cell

You can use the restoreConfig and wsadmin commands to roll back a migrated WebSphere Application Server Version 8.0 WebSphere Application Server, Network Deployment cell to Version 6.x or 7.x. This returns the configuration to the state that it was in before migration. After rolling back the WebSphere Application Server, Network Deployment cell, you can restart the migration process.

Before you begin

Best practice: When migrating a Version 6.x or 7.x WebSphere Application Server, Network Deployment cell, the best practice is to perform the following actions if you want to be able to roll it back to its previous state after migration:

1. Back up your existing configuration.
   - Run the backupConfig command or your own preferred utility to back up the Version 6.x or 7.x deployment manager configuration.

     Important: Make sure that you note the exact name and location of this backed-up configuration.

     Read the "backupConfig command" article in the information center for more information.
   - Run the backupConfig command or your own preferred utility to back up the Version 6.x or 7.x federated node configurations.

     Important: Make sure that you note the exact name and location of each of these backed-up configurations.

     Read the "backupConfig command" article in the information center for more information.
2. Migrate the WebSphere Application Server, Network Deployment cell.

Procedure

1. Stop all of the servers and node agents that are currently running in the Version 8.0 environment.
2. If you chose to disable the previous deployment manager when you migrated to the Version 8.0 deployment manager, perform one of the following actions.

   Note: Disablement is the default.

   a. If you backed up your previous deployment manager configuration using the backupConfig command or your own preferred backup utility, run the restoreConfig command or your own preferred utility to restore the Version 6.x or 7.x configuration for the deployment manager.

      Important: Make sure that you restore the same backed-up configuration that you created just before you migrated the deployment manager.

      Read the "restoreConfig command" article in the information center for more information.

   b. If you did not back up your previous deployment manager configuration, use the wsadmin command to run the `migrationDisablementReversal.jacl` script from the Version 6.x or 7.x *app_server_root*`/bin` directory of the deployment manager.

      Use the following parameters:

      ```
      ./wsadmin.sh -f migrationDisablementReversal.jacl -conntype NONE
      ```

      Tip: If you have trouble running the `migrationDisablementReversal.jacl` script, try to manually perform the steps in the script.

      1) Go to the following directory:

         profile_root`/config/cells/`*cell_name*`/nodes/`*node_name*

         where *node_name* is the name of the deployment manager node that you want to roll back.

      2) If you see a `serverindex.xml_disabled` file in this directory, perform the following actions:
         a) Delete or rename the `serverindex.xml` file.
         b) Rename the `serverindex.xml_disabled` file to `serverindex.xml`.

3. Perform one of the following actions for each of the WebSphere Application Server, Network Deployment cell's federated nodes that you need to roll back.
   a. If you backed up your previous federated node configuration using the backupConfig command or your own preferred backup utility, run the restoreConfig command or your own preferred utility to restore the Version 6.x or 7.x configuration for the federated node.

      Important: Make sure that you restore the same backed-up configuration that you created just before you migrated the federated node.

      Read the "restoreConfig command" article in the information center for more information.
   b. If you did not back up your previous federated node configuration, use the wsadmin command to run the `migrationDisablementReversal.jacl` script from the Version 6.x or 7.x *app_server_root*`/bin` directory of the federated node.

      Use the following parameters:

      ```
      ./wsadmin.sh -f migrationDisablementReversal.jacl -conntype NONE
      ```

      Tip: If you have trouble running the `migrationDisablementReversal.jacl` script, try to manually perform the steps in the script.
      1) Go to the following directory:

         profile_root`/config/cells/`*cell_name*`/nodes/`*node_name*

         where *node_name* is the name of the federated node that you want to roll back.
      2) If you see a `serverindex.xml_disabled` file in this directory, perform the following actions:
         a) Delete or rename the `serverindex.xml` file.
         b) Rename the `serverindex.xml_disabled` file to `serverindex.xml`.
4. Synchronize the federated nodes if they were ever running when the Version 8.0 deployment manager was running.

   Read the "Synchronizing nodes with the wsadmin tool" article in the information center for more information.
5. If you chose to keep the installed applications in the same location as the prior release during migration to Version 8.0 and any of the Version 8.0 applications are not compatible with the prior release, install applications that are compatible.
6. Start the rolled-back deployment manager and its federated nodes in the Version 6.x or 7.x environment.

Results

The configuration should now be returned to the state that it was in before migration.

What to do next

You can now restart the migration process if you want to do so.

Rolling back a federated node

You can use the restoreConfig and wsadmin commands to roll back a migrated WebSphere Application Server Version 8.0 federated node to the state that it was in before migration. For each federated node that you want to roll back, you must roll back the federated node itself and the corresponding changes made to the primary repository located on the deployment manager.

Before you begin

Best practice: When migrating a Version 6.x or 7.x federated node, the best practice is to perform the following actions if you want to be able to roll it back to its previous state after migration:

1. Back up your existing configuration.
   a. Run the backupConfig command or your own preferred utility to back up the Version 8.0 deployment manager configuration.

      Important: Make sure that you note the exact name and location of this backed-up configuration.

      Read the "backupConfig command" article in the information center for more information.
   b. Run the backupConfig command or your own preferred utility to back up the Version 6.x or 7.x federated node configuration.

      Important: Make sure that you note the exact name and location of this backed-up configuration.

      Read the "backupConfig command" article in the information center for more information.
2. Migrate the federated node.
3. If necessary, you can now roll back the federated node that you just migrated.

Important: If you do not have a backup copy of your Version 8.0 deployment manager configuration as it was before you migrated the Version 6.x or 7.x federated node that you want to roll back, you cannot use the procedure described in this article and you must roll back your whole cell.

About this task

You must perform all of the backup and rollback actions for each migrated federated node before you proceed to migrate another federated node.

Procedure

1. Run the backupConfig command or your own preferred utility to back up the Version 8.0 deployment manager configuration at the current state.

   Important: Make sure that you note the exact name and location of this backed-up configuration.

   Read the "backupConfig command" article in the information center for more information.
2. Stop all servers and the node agent on the Version 8.0 federated node that you want to roll back.
3. Restore your previous configuration.
   a. Run the restoreConfig command or your own preferred utility to restore the previous Version 8.0 deployment manager configuration.

      Important:

      - Make sure that you restore the same backed-up configuration that you created just before you migrated the federated node.
      - If you have made changes to your environment (application or configuration changes for example), these changes are rolled back at the same time and cause the other nodes to force synchronization with the deployment manager.

      Read the "restoreConfig command" article in the information center for more information.
   b. Perform one of the following actions to restore the Version 6.x or 7.x configuration for the federated node.
      - Run the restoreConfig command or your own preferred utility to restore the Version 6.x or 7.x configuration.

        Important: Make sure that you restore the same backed-up configuration that you created just before you migrated this federated node.

Read the "restoreConfig command" article in the information center for more information.

- Use the wsadmin command to run the `migrationDisablementReversal.jacl` script from the Version 6.x or 7.x `WAS_HOME/bin` directory of the federated node that you need to roll back from Version 8.0.

  Use the following parameters:

  ```
  ./wsadmin.sh -f migrationDisablementReversal.jacl -conntype NONE
  ```

  Tip: If you have trouble running the `migrationDisablementReversal.jacl` script, try to manually perform the steps in the script.

  1) Go to the following directory:

     profile_root/config/cells/*cell_name*/nodes/*node_name*

     where *node_name* is the name of the federated node that you want to roll back.

  2) If you see a `serverindex.xml_disabled` file in this directory, perform the following actions:

     a) Delete or rename the `serverindex.xml` file.

     b) Rename the `serverindex.xml_disabled` file to `serverindex.xml`.

4. Start the Version 8.0 deployment manager.
5. Perform any application maintenance that is required.
6. Synchronize the Version 6.x or 7.x federated node with the deployment manager.

   Read the "Synchronizing nodes with the wsadmin tool" article in the information center for more information.
7. If you chose to keep the installed applications in the same location as the prior release during migration to Version 8.0 and any of the Version 8.0 applications are not compatible with the prior release, install applications that are compatible.
8. Start the rolled-back Version 6.x or 7.x federated node and servers.
9. Validate that the configuration is satisfactory.

   This is the last chance to undo the rollback action by restoring the deployment-manager configuration that you backed up in the first step.

Results

The configuration should now be returned to the state that it was in before migration.

What to do next

You can now restart the migration process if you want to do so.

Rolling back stand-alone application servers

You can use the restoreConfig and wsadmin commands to roll back a migrated WebSphere Application Server Version 8.0 stand-alone application server to the state that it was in before migration.

Before you begin

Best practice: When migrating a Version 6.x or 7.x stand-alone application server, the best practice is to perform the following actions if you want to be able to roll it back to its previous state after migration:

1. Run the backupConfig command or your own preferred utility to back up the Version 6.x or 7.x stand-alone application server configuration.

Important: Make sure that you note the exact name and location of this backed-up configuration.

Read the "backupConfig command" article in the information center for more information.

2. Migrate the stand-alone application server.
3. If necessary, you can now roll back the stand-alone application server that you just migrated.

Procedure

1. Stop all of the servers that are currently running in the Version 8.0 environment.
2. Perform one of the following actions to restore the Version 6.x or 7.x configuration for the stand-alone application server.
   - Run the restoreConfig command or your own preferred utility to restore the Version 6.x or 7.x configuration.

     Important: Make sure that you restore the same backed-up configuration that you created just before you migrated this stand-alone application server.

     Read the "restoreConfig command" article in the information center for more information.
   - Use the wsadmin command to run the `migrationDisablementReversal.jacl` script from the Version 6.x or 7.x `WAS_HOME/bin` directory of the stand-alone application server that you need to roll back from Version 8.0.

     Use the following parameters:

     ```
     ./wsadmin.sh -f migrationDisablementReversal.jacl -conntype NONE
     ```
3. If you chose to keep the installed applications in the same location as the prior release during migration to Version 8.0 and any of the Version 8.0 applications are not compatible with the prior release, install applications that are compatible.
4. Start the rolled-back stand-alone application server in the Version 8.0 environment.

Results

The configuration should now be returned to the state that it was in before migration.

What to do next

You can now restart the migration process if you want to do so.

Chapter 4. Scenario 1: Migrating an administrative agent profile and its registered set of managed base application servers on z/OS

Administrative agent profiles manage multiple base application servers in environments such as development, unit test, or that portion of a server farm that resides on a single machine. Before you can migrate managed base application servers from version 7.0 to version 8.0, you must first migrate the administrative agent.

Before you begin

Review the migration planning information. See Knowledge Collection: Migration planning for WebSphere Application Server.

Note: You can migrate administrative agent profiles and the registered set of managed base application servers from Version 7.0 to Version 8.0.

About this task

This task describes how to use the z/OS Configuration Migration Management Tool to migrate an administrative agent and its associated set of managed base application servers from WebSphere Application Server Version 7.0 to Version 8.0. A base application server becomes managed when it is registered with a single administrative agent. An administrative agent may manage one or more base application servers and must be at the same release level and on the same machine as the base application servers it is managing. z/OS restricts that only the version 7.0 or version 8.0 administrative agent be running at any one time. The following instructions will tell you when to start and stop the old and new administrative agents. This restriction requires that all managed base application servers be migrated at the same time.

Note: When migrating the managed base application server in a flexible management environment, ensure that the node names are the same on Version 8.0 and previous releases.

Procedure

1. Run the backupConfig command on the administrative agent and all the managed base application servers In case of failure during the migration, save the current administrative agent and node configuration to a file that you can use later for recovery purposes.
   a. Change to the *<administrative agent profile root>/bin* directory.
   b. Run the backupConfig command with the appropriate parameters and save the current profile configuration to a file. For example:

      ```
      /opt/WebSphereV61/profiles/v61dmgr01/bin/backupConfig.sh
      /mybackupdir/v61dmgr01backupBeforeV7migration.zip
      -username myuser -password mypass -nostop
      ```

   c. For each node in the configuration, change to the *<profile root >/bin* directory.
   d. Run the backupConfig command with the appropriate parameters, and save the current profile configuration to a file. For example:

      ```
      /opt/WebSphereV70/profiles/v70mas01/bin/backupConfig.sh
      /mybackupdir/v70mas01rbackupBeforeV8migration.zip
      -username myuser -password mypass -nostop
      ```

   e. Alternately, you can use the PAX command to back up all HFS files. For more information, see Using the z/OS UNIX pax command
2. Install WebSphere Application Server Version 8.0 Install WebSphere Application Server Version 8.0 using the SMP/E tool.

© IBM Corporation 2009

a. Install the correct version (Network Deployment, Base, Express) of WebSphere Application Server onto each target host. For more information, see How do I install an application serving environment?

3. Specify an administrative agent migration configuration in the z/OS Migration Management Tool Use the z/OS Migration Management Tool to create a migration definition and upload the jobs for migrating the administrative agent.

   Note: The Version 8.0 cell name must match the cell name in the Version 7.0 configuration. If you create a profile with a new cell name, then the migration will fail

   a. Complete the worksheet found in the following information center topic: Specifying a deployment manager migration configuration in the z/OS Migration Management Tool: Worksheet
   b. On the Migration Node Type Selection panel, select **z/OS migrate a administrative agent**.
   c. Complete the fields on the subsequent panels using the values from your worksheet.
   d. Review the administrative agent migration definition to make sure that all the values are correct.
      - In the WebSphere Application Server for z/OS migration definition table, select the migration definition that you want to review.
      - Click **View**.
      - For information about the migration definition, click the **Summary**, **Instructions**, or **Response file** tab.
   e. Upload the migration jobs to the target z/OS system.
4. Ensure that all the in progress jobs are completed on the managed profiles Before running the WASPreUpgrade command on a managed application server or deployment manager, all in progress jobs must be complete.
5. Stop polling the job manager To avoid problems, stop polling the job manager before running the WASPreUpgrade on profiles that are getting jobs from the job manager. Before you start polling for jobs, complete WASPreUpgrade and WASPostUpgrade for the managed profile. For more information, see ManagedNodeAgent command group for the AdminTask object using wsadmin scripting.
6. Run the migration jobs for the administrative agent Follow the instructions in the 'Migration Instructions' view of the z/OS Migration Management Tool or in the BBOMAINS member of the CNTL dataset that you uploaded to the target z/OS system. The CRPROF (profile create), PREUPGRD (preUpgrade) and UPGRADE steps are all executed during this process.
7. Verify all of the migration output that is generated Verify all output created in the working directory of the HFS/ZFS (specified in ZMMT) and all MVS job output created by the batch jobs. Verify that you are getting return codes of 0, and review the log files in the migration temp directory on the configuration file system. The migration temporary directory is temporary_directory_location/nnnnn, where temporary_directory_location is the directory specified for the temporary directory location (/tmp/migrate by default) and nnnnn is the numeric value generated for the migration identifier when you generated your migration jobs.
8. Add STEPLIB statements that include the previous version's load modules to the migrated Daemon's startup proc Use this step if you are running a mixed cell environment only. STEPLIB statements apply to WebSphere Application Server Version 6.1 and earlier. If you are running with multiple nodes on the same LPAR, or if you are running with a node on the same LPAR as the migrated DMGR (which will create a mixed cell environment), you will need to add STEPLIB statements that include the previous version's load modules to the migrated Daemon's startup proc. This is due to the fact that the Daemon must run at the highest WAS level on that LPAR. This is necessary if for any reason you need to stop and restart the down level node agent or application server. For example, after migration of the DMGR, the BBO7DMN procedure was modified to include:

```
//*
//STEPLIB  DD DISP=SHR,DSN=xxxxxxxx.SBBOLD2
//         DD DISP=SHR,DSN=xxxxxxxx.SBBOLOAD
//         DD DISP=SHR,DSN=xxxxxxxx.SBBOLPA
```

Where xxxxxxxx = pointers to the previous WebSphere Application Server level. This enables the Daemon to manage the migrated DMGR and previous level node agents.

9. Specify a managed app migration configuration in the z/OS Migration Management Tool Use the z/OS Migration Management Tool to create a migration definition and upload the jobs for migrating the manged base application servers.

   Note: The Version 8.0 cell name must match the cell name in the Version 6.x Version 7.0 configuration. If you create a profile with a new cell name, then the migration will fail.

   a. Complete the worksheet found in the following information center topic: Specifying a federated node migration configuration in the z/OS Migration Management Tool: Worksheet
   b. On the Migration Node Type Selection panel, select **z/OS migrate a managed base application server**.
   c. Complete the fields on the subsequent panels using the values from your worksheet.
   d. Review the federated node migration definition to make sure that all the values are correct.
      - In the WebSphere Application Server for z/OS migration definition table, select the migration definition that you want to review.
      - Click **View**.
      - For information about the migration definition, click the **Summary**, **Instructions**, or **Response file** tab.
   e. Upload the migration jobs to the target z/OS system.

10. Run the migration jobs for the managed base application server

    Note:

    - Ensure that the Version 7.0 administrative agent is running. This allows the application server to be de-registered with the administrative agent.
    - If the managed application server is also registered to a job manager, you must unregister from the job manager before running a managed application server migration. After the migration is complete, register to the administrative agent and job manager again.

    Follow the instructions in the 'Migration Instructions' view of the z/OS Migration Management Tool or in the BBOMDINS member of the CNTL dataset that you uploaded to the target z/OS system. The CRPROF (profile create), PREUPGRD (preUpgrade) and UPGRADE steps are all executed during this process.

11. Verify all of the migration output that is generated Verify all output created in the working directory of the HFS (specified in ZMMT) and all MVS job output created by the batch jobs. Verify that you are getting return codes of 0, and review the log files in the migration temp directory on the configuration file system. The migration temporary directory is `temporary_directory_location/nnnnn`, where `temporary_directory_location` is the directory specified for the temporary directory location (`/tmp/migrate` by default) and nnnnn is the numeric value generated for the migration identifier when you generated your migration jobs.
12. Add STEPLIB statements that include the previous version's load modules to the migrated Daemon's startup proc Use this step if you are running a mixed cell environment only. STEPLIB statements apply to WebSphere Application Server Version 6.1 and earlier. If you are running with multiple nodes on the same LPAR, or if you are running with a node on the same LPAR as the migrated DMGR (which will create a mixed cell environment), you will need to add STEPLIB statements that include the previous version's load modules to the migrated Daemon's startup proc. This is due to the fact that the Daemon must run at the highest WAS level on that LPAR. This is necessary if for any reason you need to stop and restart the down level node agent or application server. For example, after migration of the DMGR, the BBO7DMN procedure was modified to include:

```
//*
//STEPLIB  DD DISP=SHR,DSN=xxxxxxxx.SBBOLD2
//         DD DISP=SHR,DSN=xxxxxxxx.SBBOLOAD
//         DD DISP=SHR,DSN=xxxxxxxx.SBBOLPA
```

Where xxxxxxxx = pointers to the previous WebSphere Application Server level. This enables the Daemon to manage the migrated DMGR and previous level node agents.

13. Change to the new V8 migrated administrative agent

    Note: Be sure all the managed application server registered to the old administrative agent are migrated. These should have all been de-registered during their migration process.

    a. Stop the old Version 7.0 administrative agent.
    b. Start the new Version 8.0 administrative agent.
14. Start the managed application server Check the Node Agent logs for errors.
    a. Start each managed base application server.
    b. Register each managed application server with the new administrative agent.
15. Troubleshooting See the information center documentation on troubleshooting migration.

Results

You have migrated an administrative agent profile and its associated managed base application servers from WebSphere Application Server Version 7.0 to Version 8.0 using the migration tools.

Chapter 5. Scenario 2: Migrating a job manager profile and its registered set of servers using the z/OS Configuration Migration Management Tool

You can migrate a job manager profile and its registered set of servers from Version 7.0 to Version 8.0.

Before you begin

Review the migration planning information. See Knowledge Collection: Migration planning for WebSphere Application Server.

Note: You can migrate administrative agent profiles and the registered set of managed base application servers from Version 7.0 to Version 8.0.

About this task

This task outlines the steps needed to migrate the job manager and the set of registered servers.

Job manager profiles can have one or more of the following server types registered:

- Deployment manager servers
- Managed base application servers (which are also registered to an administrative agent)

Note:

1. Managed base application servers and deployment manager servers cannot accept jobs from a job manager that is of a previous version. To avoid problems, migrate your job manager profiles to Version 8.0 before you migrate managed base application servers and deployment manager servers to Version 8.0.
2. When migrating the managed base application server or managed deployment manager in a flexible management environment, the node names must be the same in Version 8.0 and the previous releases.

Note: You can migrate job manager profiles and the registered set of servers from Version 7.0 to Version 8.0.

Procedure

1. Run the backupConfig command on the administrative agent and all the managed base application servers In case of failure during the migration, save the current administrative agent and node configuration to a file that you can use later for recovery purposes.
   a. Change to the *<job manager profile root>/bin* directory.
   b. Run the backupConfig command with the appropriate parameters and save the current profile configuration to a file. For example:

      ```
      /opt/WebSphereV61/profiles/jobMgr01/bin/backupConfig.sh
      /mybackupdir/jobMgrBeforeV8migration.zip -username myuser
      -password mypass -nostop
      ```

   c. Alternately, you can use the PAX command to back up all HFS files. For more information, see Using the z/OS UNIX pax command
2. Install WebSphere Application Server Version 8.0 Install WebSphere Application Server Version 8.0 using the SMP/E tool.
   a. Install the correct version (Network Deployment, Base, Express) of WebSphere Application Server onto each target host. For more information, see How do I install an application serving environment?

© IBM Corporation 2009

3. Specify an job manager migration configuration in the z/OS Migration Management Tool Use the z/OS Migration Management Tool to create a migration definition and upload the jobs for migrating the job manager.

   Note: The Version 8.0 cell name must match the cell name in the Version 7.0 configuration. If you create a profile with a new cell name, then the migration will fail.

   a. Complete the worksheet found in the following information center topic: Specifying a deployment manager migration configuration in the z/OS Migration Management Tool: Worksheet
   b. On the Migration Node Type Selection panel, select **z/OS migrate a job manager**.
   c. Complete the fields on the subsequent panels using the values from your worksheet.
   d. Review the job manager migration definition to make sure that all the values are correct.
      - In the WebSphere Application Server for z/OS migration definition table, select the migration definition that you want to review.
      - Click **View**.
      - For information about the migration definition, click the **Summary**, **Instructions**, or **Response file** tab.
   e. Upload the migration jobs to the target z/OS system.
4. Stop the old job manager Any jobs that exist within the old job manager database will be migrated as part of the migration.
5. Run the migration jobs for the administrative agent Follow the instructions in the 'Migration Instructions' view of the z/OS Migration Management Tool or in the BBOMAINS member of the CNTL dataset that you uploaded to the target z/OS system. The CRPROF (profile create), PREUPGRD (preUpgrade) and UPGRADE steps are all executed during this process.
6. Verify all of the migration output that is generated Verify all output created in the working directory of the HFS/ZFS (specified in ZMMT) and all MVS job output created by the batch jobs. Verify that you are getting return codes of 0, and review the log files in the migration temp directory on the configuration file system. The migration temporary directory is temporary_directory_location/nnnnn, where temporary_directory_location is the directory specified for the temporary directory location (/tmp/migrate by default) and nnnnn is the numeric value generated for the migration identifier when you generated your migration jobs.
7. Add STEPLIB statements that include the previous version's load modules to the migrated Daemon's startup proc Use this step if you are running a mixed cell environment only. STEPLIB statements apply to WebSphere Application Server Version 6.1 and earlier. If you are running with multiple nodes on the same LPAR, or if you are running with a node on the same LPAR as the migrated DMGR (which will create a mixed cell environment), you will need to add STEPLIB statements that include the previous version's load modules to the migrated Daemon's startup proc. This is due to the fact that the Daemon must run at the highest WAS level on that LPAR. This is necessary if for any reason you need to stop and restart the down level node agent or application server. For example, after migration of the DMGR, the BBO7DMN procedure was modified to include:

   ```
   //*
   //STEPLIB  DD DISP=SHR,DSN=xxxxxxxx.SBBOLD2
   //         DD DISP=SHR,DSN=xxxxxxxx.SBBOLOAD
   //         DD DISP=SHR,DSN=xxxxxxxx.SBBOLPA
   ```

   Where xxxxxxxx = pointers to the previous WebSphere Application Server level. This enables the Daemon to manage the migrated DMGR and previous level node agents.
8. Start the Version 8.0 job manager Ensure that both Version 7.0 and 8.0 of the job manager are running.
   a. Change to the new Version 8.0 job manager profile bin directory
   b. Run the startServer jobmgr command
   c. Check the SystemOut.log file for warnings or errors

9. Migrate the registered servers The Version 8.0 job manager can manage Version 7.0 registered servers. For the Version 7.0 topology to function with the new Version 8.0 job manager, you are not required to migrate the registered servers. For each registered server that you plan to migrate to Version 8.0, perform the following steps:
   - If the registered server you wish to migrate is of type deployment manager, see Scenario 1: Migrating a cell using the command line tools
   - If the registered server you wish to migrate is of type managed base application server, see Flexible Management Scenario 1: Migrating an administrative agent profile and its registered set of managed base application servers
10. Troubleshooting See the information center documentation on troubleshooting migration.

Results

You have migrated a job manager profile and its associated managed base application servers from WebSphere Application Server Version 7.0 to Version 8.0 using the migration tools.

Chapter 6. Migrating web server configurations

You can migrate a web server so that it supports the latest version of WebSphere Application Server.

Before you begin

About this task

Migrating from WebSphere Application Server Version 6.x or 7.x:

- Only web servers defined on managed nodes are migrated to WebSphere Application Server Version 8.0.
- If you are migrating a web server and plug-ins from WebSphere Application Server Version 6.x or 7.x to Version 8.0 and the web server is defined on an unmanaged node, the web server creation and application mapping must be done manually.

  To create the web server definition manually, perform one of the following actions:

  – Use the administration console wizard.

    To generate mapping to all applications that are installed at web server creation, use the mapping `ALL` option in the wizard.

  – Use the wsadmin command.

    ```
    $AdminTask createWebServer -interactive
    ```

    and reply `ALL` to the mapping applications prompt.

  – Use the `configureWebserverDefintion.jacl` script.

    This script maps all installed applications to the web server. The script updates all of the information related to the web server plug-in such as the locations of the plug-in installation root, log file, configuration file, and key stores on the Web-server system. However, the script does not update other properties related to the web

© IBM Corporation 2003, 2008

server if the web server definition already exists.

Chapter 7. Migrating administrative scripts

You can migrate administrative scripts using scripting and the wsadmin tool.

Before you begin

About this task

WebSphere Application Server Version 8.0 supports migrating administrative scripts from Version 6.x or 7.x.

Procedure

If you are migrating administrative scripts from Version 6.x or 7.x, see:
"Migrating administrative scripts from Version 6.x or 7.x to Version 8.0" on page 135

Migrating administrative scripts from a previously Version 5.1.x application server

There are some changes you should be aware of when migrating from a version of WebSphere Application Server that was previously Version 5.1.x.

Before you begin

About this task

There are a few changes to be aware of that are required for your existing scripts when moving to WebSphere Application Server Version 8.0. In general, the administration model has changed very little. However, there are some changes required when moving from a previously Version 5.1.x to Version 8.0.

Procedure

- There are a few changes to be aware of that are required for your existing Version 5.1.x scripts when moving to WebSphere Application Server Version 8.0. These types of changes can be evolved directly without the assistance of script compatibility support. The data can be accessed from multiple locations, including the old and new locations. As long as the new location is not updated, the data is accessed from the old location. Once the new location is updated, it becomes the current data and is used for further accesses and updates. Warning messages are logged when the old location is still being used.

  Read "Example: Migrating - Changing transaction log directory using wsadmin scripting" on page 135 for more information.
- Be aware of removed features that might have an impact on administration scripts.

  These might include the following:
  - Support for the Secure Authentication Service (SAS) IIOP security protocol
  - Support for the Common Connector Framework (CCF)
  - Support for the IBM Cloudscape Version 5.1.x database
  - Support for the following Java Database Connectivity (JDBC) drivers:
    - WebSphere Connect JDBC driver
    - Microsoft SQL Server 2000 Driver for JDBC
    - WebSphere SequeLink JDBC driver for Microsoft SQL Server

© Copyright IBM Corp. 2011

Read “Migrating from the WebSphere Connect JDBC driver” on page 115 for information on using the WebSphereConnectJDBCDriverConversion command to convert data sources from the WebSphere Connect JDBC driver to the DataDirect Connect JDBC driver or the Microsoft SQL Server JDBC driver.

– Use of JSSE instead of System SSL for all servers except the Daemon
– DB2 for zOS Local JDBC Provider (RRS).

For more information, see the “Premigration considerations” on page 6 article.

For more information, see the "Deprecated and removed features" article in the information center.

Example: Migrating - Allowing configuration overwrite when saving configurations

These examples demonstrate how to enable configuration overwrite in network deployment for WebSphere Application Server Version 6.x.

Use the following examples:

- wsadmin Version 6.x

1. Enable configuration repository to allow configuration overwrite:

   depfeat: Using Jacl:

```
set s1AdminService [$AdminConfig getid /Server:dmgr/AdminService:/]

set configRepository [$AdminConfig showAttribute $s1AdminService configRepository]
set props [$AdminConfig showAttribute $configRepository properties]
set foundAllowConfigOverwrites ""
if {$props != "{}"} {
 foreach prop $props {
  if {[$AdminConfig showAttribute $prop name] == "allowConfigOverwrites"} {
   set foundAllowConfigOverwrites $prop
   break
  }
 }
}

if {$foundAllowConfigOverwrites == ""} {
 $AdminConfig create Property $configRepository {{name allowConfigOverwrites} {value true}}
} else {
 $AdminConfig modify $foundAllowConfigOverwrites {{value true}}
}

$AdminConfig save
```

   Using Jython:

```
s1AdminService = AdminConfig.getid('/Server:dmgr/AdminService:/')
configRepository = AdminConfig.showAttribute(s1AdminService, 'configRepository')
props = AdminConfig.showAttribute(configRepository, 'properties')
foundAllowConfigOverwrites = ''
if props != '[]':
 properties = props[1:len(props)-1].split(' ')
 for prop in properties:
  name = AdminConfig.showAttribute(prop, 'name')
  if name == 'allowConfigOverwrites':
   foundAllowConfigOverwrites = prop
   break

if len(foundAllowConfigOverwrites) != 0:
 AdminConfig.modify(foundAllowConfigOverwrites, [['value', 'true']])
else:
 AdminConfig.create('Property', configRepository, [['name', 'allowConfigOverwrites'], ['value', 'true']])

AdminConfig.save()
```

2. Restart the deployment manager. From the `bin` directory of the deployment manager profile, run the following:

```
./stopManager.sh
./startManager.sh
```

3. Allow configuration overwrite, for example:

   depfeat: Using Jacl:

   ```
   $AdminConfig setSaveMode overwriteOnConflict
   ```

   Using Jython:

   ```
   AdminConfig.setSaveMode('overwriteOnConflict')
   ```

Example: Migrating - Changing transaction log directory using wsadmin scripting

Prepare for evolutionary changes without script compatibility support.

The location of the `transaction logs directory` attribute has changed from the `ApplicationServer::TransactionService` to the `ServerEntry::recoveryLogs`. As long as the new location is not used, the value from the old location will continue to be used. Scripts that modify the old location can still be used; that value will take effect until a value in the new location is set. The change to scripts to use the new location is as follows:

Old location:

- Using Jacl:

```
set transService [$AdminConfig list TransactionService $server1]
$AdminConfig showAttribute $transService transactionLogDirectory
```

New Location:

- Using Jython:

```
AdminConfig.list("ServerEntry")

# Select one entry from the list, e.g the entry for server1:
serverEntryId = AdminConfig.getid("/ServerEntry:server1")
serverEntry = AdminConfig.list("ServerEntry", serverEntryId)

recoveryLog = AdminConfig.showAttribute(serverEntry, "recoveryLog")
AdminConfig.showAttribute(recoveryLog, "transactionLogDirectory")
```

- Using Jacl:

```
$AdminConfig list ServerEntry $node
set serverEntry <select one of the ServerEntry from output of above command>
set recoveryLog [$AdminConfig showAttribute $serverEntry recoveryLog]
$AdminConfig showAttribute $recoveryLog transactionLogDirectory
```

Migrating administrative scripts from Version 6.x or 7.x to Version 8.0

There are some changes you should be aware of when migrating your administrative scripts from WebSphere Application Server Version 6.x or 7.x to Version 8.0.

Before you begin

Procedure

- Be aware of removed features that might have an impact on administration scripts.

  These might include the following:

  – Support for the Secure Authentication Service (SAS) IIOP security protocol
  – Support for the Common Connector Framework (CCF)
  – Support for the IBM Cloudscape Version 5.1.x database
  – Support for the following Java Database Connectivity (JDBC) drivers:

- WebSphere Connect JDBC driver
- Microsoft SQL Server 2000 Driver for JDBC
- WebSphere SequeLink JDBC driver for Microsoft SQL Server

Read “Migrating from the WebSphere Connect JDBC driver” on page 115 for information on using the WebSphereConnectJDBCDriverConversion command to convert data sources from the WebSphere Connect JDBC driver to the DataDirect Connect JDBC driver or the Microsoft SQL Server 2005 JDBC driver.

– Use of JSSE instead of System SSL for all servers except the Daemon
– DB2 for zOS Local JDBC Provider (RRS).

For more information, see the “Premigration considerations” on page 6 article.

For more information, see the "Deprecated and removed features" article in the information center.

- Be aware of the change required when creating Service Integration Bus (SIB) objects.

  For more information about the createSIBus command, see the "createSIBus command" article in the information center.

Chapter 8. Running multiple application server versions

You can create an environment in which multiple versions of WebSphere Application Server can run independently on the same system at the same time. A major consideration in coexistence is the avoidance of port conflicts.

Before you begin

Coexisting, as it applies to WebSphere Application Server products, involves multiple installations of WebSphere Application Server running independently on the same system at the same time. Multiple installations include multiple versions and multiple installations of one version. Coexistence also implies various combinations of Web server interaction.

Procedure

Read “Coexistence support.”
This article discusses which coexistence scenarios are supported.

Coexistence support

Coexistence is a state in which multiple installations and multiple nodes from different versions of WebSphere Application Server run independently in the same environment at the same time.

As it applies to WebSphere Application Server for z/OS products, coexistence primarily refers to the ability of multiple customizations (nodes) of WebSphere Application Server to run on the same z/OS image or sysplex at the same time. You can install WebSphere Application Server for z/OS once and customize it many times; that is, you can create many nodes from one installation. This installation can be shared by all LPARs (logical partitions) in the same sysplex.

When setting up your system for coexistence, you need to be aware of the points in the following table:

Table 66. WebSphere Application Server for z/OS Version 6.x,, 7.x, and Version 8.0 multiversion coexistence scenarios. The table lists the coexistence types with comments about limitations and considerations for each type.

Coexistence Type	Comments
General	Multiple WebSphere Application Server for z/OS cells can coexist in the same sysplex. Multiple WebSphere Application Serverfor z/OS nodes can coexist on the same LPAR. WebSphere Application Server for z/OS Version 6.x and 7.x can coexist in the same cell with Version 8.0 nodes. No two cells can have the same cell short name. Separate cells need separate configuration file system mount points and Job Control Language (JCL) procedures. You need to determine port conflicts that might occur when earlier versions coexist with Version 8.0. Read the "Default port assignments" article in the information center for default port information.
WebSphere Application Server for z/OS Version 6.x, 7.x, and Version 8.0	Nodes can coexist on the same LPAR in the same cell or in different cells.

© Copyright IBM Corp. 2011

Chapter 9. Interoperating multiple application server versions

WebSphere Application Server Version 8.0 is interoperable with other versions of the product under certain conditions.

Before you begin

Read “Overview of migration, coexistence, and interoperability” on page 1 and “Premigration considerations” on page 6. For resources to help you plan and perform your migration, visit Knowledge Collection: Migration planning for WebSphere Application Server.

WebSphere Application Server Version 8.0 is generally interoperable with Version 6.x and 7.0. You should apply the latest fix level to support interoperability. In order to interoperate between WebSphere Application Server for z/OS and a WebSphere Application Server product on a non-z/OS machine, however, you must follow this procedure.

You can upgrade a portion of the nodes in a cell to WebSphere Application Server Version 8.0 while leaving others at the older release level. This means that, for a period of time, you might be administering servers that are at the current release and servers that are running the newer release in the same cell.

Restrictions on using mixed-release cells:

- A mixed-release environment where some members are at an older release level might have some restrictions. For details, read the "Creating application servers" article in the information center.
- A WebSphere Application Server, Network Deployment cell can contain mixed releases of Version 6.x nodes, but there is no mixed-node management support for Version 6.0.1.x.

  The Version 8.0 migration tools still migrate these nodes during deployment-manager migration, but they issue a warning message that the nodes cannot be managed by the Version 8.0 deployment manager. You can then do one of the following based on your needs:
  - Upgrade all Version 6.0.1.x nodes to at least Version 6.0.2. This will allow them to be administered by a Version 8.0 deployment manager.
  - Migrate these nodes to Version 8.0.

Procedure

1. On the non-z/OS machine, the property `-Dcom.ibm.CORBA.ORBCharEncoding=ASCII` should be set in the administrative console under ***server_name*** **> Java and Process Management >Process Definition > Java Virtual Machine > Generic JVM arguments**.
2. On the non-z/OS machine, the property `com.ibm.CORBA.ORBWCharDefault = UCS2` should be defined in the administrative console under ***server_name*** **> Container Services > ORB Service > Custom Properties**.
3. If you are running WebSphere Application Server for z/OS with security enabled, you must set the property `com.ibm.CORBA.validateBasicAuth=false` in `sas.client.props` on the non-z/OS machine. It is set to true by default.
4. Upgrade the Software Development Kit (SDK) used to one supported by Version 8.0.

   Read Recommended fixes for WebSphere Application Server for more information.

© Copyright IBM Corp. 2011

What to do next

This information is dynamic and might be augmented by information in technical articles that are available on the IBM DeveloperWorks WebSphere site. Check the site for the latest information.

Chapter 10. Configuring port settings

When you configure WebSphere Application Server resources or assign port numbers to other applications, you must avoid conflicts with other assigned ports. In addition, you must explicitly enable access to particular port numbers when you configure a firewall.

Before you begin

You can also use the port validator tool to find port conflicts between different WebSphere Application Server profiles, products, and servers. Read the "port validator tool" article in the information center for more information.

Tip: Port conflicts might occur if you install WebSphere Application Server on multiple systems with deployment managers managing servers or clusters on different systems. The configuration-service port-resolution mechanism does not support cross profiles on different host machines.

- **Example 1:**
  1. On system A, create a cell profile that includes Dmgr and AppSrv01 (Node1).
  2. On system B, create AppSrv01 and federate AppSrv01 (Node2) to Dmgr on system A.
  3. Create server1 on Node1 and server2 on Node2.
  4. The server1 server and server2 server might contain duplicate server endpoint ports in the `serverindex.xml` file because Node1 and Node2 are located on different host systems.
- **Example 2:**
  1. On system A, create a cell profile that includes Dmgr and AppSrv01 (Node1).
  2. On system B, create AppSrv01 and federate AppSrv01 (Node2) to Dmgr on system A.
  3. On system B, create JobManager.
  4. Create a cluster and add two servers, server1 on Node1 and server2 on Node2.
  5. The server2 server and the JobManager server might contain duplicate server endpoint ports in the `serverindex.xml` file because server2 and JobManager are in cross profiles. The server2 server is under Dmgr, JobManager is under the JobManager profile. and the Dmgr and JobManager profiles are located on different machines.

About this task

Procedure

1. Review the port number settings, especially when you are planning to coexist.
2. Optional: Change the port number settings.

   You can set port numbers when configuring (customizing) the product after installation. Start thinking about port numbers during the planning phase described in the "Planning for product configuration" article in the information center.

© IBM Corporation 2004, 2008

Chapter 11. Troubleshooting migration

You might encounter problems while migrating from an older version of WebSphere Application Server.

Before you begin

Procedure

If you encounter a problem when you are migrating from a previous version of Version 8.0, check your log files and other available information.

- Job fails in the Verify step.

  This indicates that a configuration error was detected before beginning the migration process. This can be due to either incorrect data entered when you created the migration jobs or a configuration problem. Review the log output for the error detected, then correct and rerun. The logs are located in *temporary_directory_location*/nnnnn, where *temporary_directory_location* is the value that you specified when you created the migration jobs (where the default is /tmp/migrate) and *nnnnn* is a unique number that is generated and displayed during the creation of your migration jobs as well as displayed in the JESOUT DDNAME of the WROUT and WRERR steps of your batch job stream.
- Job fails after the Verify step.

  In the event of failure in the migration job after the Verify step, you can rerun the migration job; but first, you must delete the WebSphere Application Server for z/OS configuration home directory created in the CRHOME step. This corresponds to the home directory that you entered when you created the migration jobs, and it can also be found in the migration Job Control Language (JCL) environment variable V6_HomeDir. Because the migration procedure creates a new configuration file system for each node being migrated, it is a simple process to delete the configuration and start from scratch.
- Problems occur with a federated node migration.

  A federated node is the most complex node to migrate because it is essentially two migrations rolled into one. A federated node requires a migration of the node configuration information contained in the deployment manager's primary repository as well as the configuration information contained in the federated node. Federated node migration requires an active connection to the deployment manager. If you have security enabled, it is essential that you follow the instructions that were generated when you created the migration jobs. The migration job must be submitted with a WebSphere Administrator's user ID that has been properly configured for obtaining secure connections.
- Job fails during the application-installation phase of migration.

  If you select the option for the migration process to install the enterprise applications that exist in the Version 6.x configuration into the new Version 8.0 configuration, you might encounter error messages during the application-installation phase of migration.

  The applications that exist in the Version 6.x configuration might have incorrect deployment information—typically, invalid XML documents that were not validated sufficiently in previous WebSphere Application Server runtimes. The runtime now has an improved application-installation validation process and will fail to install these malformed EAR files. This results in a failure during the application-installation phase of WASPostUpgrade and produces an "E:" error message. This is considered a "fatal" migration error.

  If migration fails in this way during application installation, you can do one of the following:

  – Fix the problems in the Version 6.x applications, and then remigrate.
  – Proceed with the migration and ignore these errors.
    1. Restart the migration job in the FINISHUP step to allow the remaining migration functions to be performed.

       Do this by adding the RESTART=FINISHUP parameter to the job card and resubmitting the job.
    2. Later, fix the problems in the applications and then manually install them in the new Version 8.0 configuration using the administrative console or an install script.

© Copyright IBM Corp. 2011

- Out-of-space errors occur.

  The migration logs are located in *temporary_directory_location*/*nnnnn*, where *temporary_directory_location* is the value that you specified when you created the migration jobs (where the default is /tmp/migrate) and *nnnnn* is a unique number that was generated during the creation of your migration jobs. Normally, the space requirements for the migration logs are small. If you enable tracing, however, the log files can be quite large. The best practice is to enable tracing only after problems have been found. If tracing is required, try to only enable tracing related to the step in the process that is being debugged. This will help to reduce the space requirements.

  You can enable tracing when you create the migration jobs using the z/OS Migration Management Tool or the zmmt command; or you can change the variables in the migration JCL from disabled to enabled in the Job Control Language (JCL) member BBOWM*x*EV in the DATA dataset that is created when you use the z/OS Migration Management Tool or the zmmt command to create a migration definition.

  Note: For a deployment manager, the member name is BBOWMDEV; for a federated node, the member name is BBOWMMEV.

  During migration, a backup copy of your Version 6.x configuration is made. This backup becomes the source of the information being migrated. The default backup location is /tmp/migrate/*nnnnn*. This location can be changed when you create the migration jobs. Depending on the size of the node being migrated, this backup can be quite large. If your temporary space is inadequate, then you will need to relocate this backup.
- Out-of-memory errors occur.

  To improve memory usage, perform the following actions:

  1. Edit your job file to prevent sharing of application space, and increase the JVM heap minimum and maximum.

     The following is an example of how to edit the BBOWM3D job for a deployment manager migration so that it uses up to 768 MB of heap, which is more than the default of 256.

     ```
     BPXBATCH SH + export _BPX_SHAREAS=NO; +
     export IBM_JAVA_OPTIONS="-Xms256M -Xmx768M"; +
     /wit/bigtmp/bbomigrt2.sh WASPreUpgrade +
     /wit/bigtmp/24173105/_ +
     1>> /wit/bigtmp/24173105/BBOWMG3D.out +
     2>> /wit/bigtmp/24173105/BBOWMG3D.err;
     ```

  2. Edit the appropriate migration script.

     If you are migrating from a system where you have access to the read-only driver file system, edit the WASPreUpgrade.sh and WASPostUpgrade.sh scripts in the bin directory.

     If you cannot edit the read-only driver system, use the three migration jobs instead of the single migration job so that the migration pauses after profile creation and you can edit the profile scripts. Running the BBOWMG3* migration job is equivalent to running the following three jobs in the order listed:

     – BBOWMPRO
     – BBOWMPRE
     – BBOWMPOS

     The BBOWMG3* job and the three individual jobs are mutually exclusive—either run the single BBOWMG3* job or run the three individual jobs, but not both.

  The *PRO job performs product installation and profile creation. After this job completes successfully, you have access to the Version 8.0 profile that will be used for the migration. Go into the Version 8.0 installation directory and go to the equivalent ${WAS_HOME}/bin location. You will find the WASPreUpgrade.sh and WASPostUpgrade.sh scripts at this location. If they are symlinks back to the read-only file system, remove the symlinks and copy the original files from the read-only file system to this location. After you have actual migration script files in ${WAS_HOME}/bin, edit those files and change the lines for Performance Java Options to change the heap settings to a value appropriate for your system. For example:

```
set PERFJAVAOPTION=-Xms256M -Xmx768M
```

You can now continue your migration. If you decided to run the three individual jobs, launch the BBOWMPRE job and after that is successful (RC=0) run the BBOWMPOS job. If you edited the read-only file-system copy of the migration script files, you can run the appropriate BBOWMG3* job.

- Batch job time is exceeded.

  Each z/OS installation is different with respect to job classes and time limitations. Make sure you have specified appropriate job classes and timeout values on your job card.

- Migration of a deployment manager or stand-alone application server completes but no applications are installed.

  The log file might show messages similar to the following:

  ```
  MIGR0339I: Application WISO_wisoadmin_war.ear is deploying using the wsadmin command.
  MIGR0241I: Output of wsadmin.
  Error: unable to allocate 268435456 bytes for GC in j9vmem_reserve_memory.
  JVMJ9VM015W Initialization error for library j9gc23(2): Failed to instantiate heap. 256M requested
  Could not create the Java virtual machine.
  ```

  The problem is that the WASPostUpgrade script launched from bbomigrt2.sh does not have enough remaining address space to initialize the Java Virtual Machine (JVM). Typically, this indicates that the spawned process is running in the same address space as the WASPostUpgrade JVM.

  You can use the environment variable _BPX_SHAREAS to tell the underlying process whether or not spawned processes should share the same address space as the parent process. The default value (null) is NO, but administrators can change this to YES or MUST to get a performance benefit because the address space does not need to be copied during fork or spawn actions.

  If your system is experiencing the problem described here, run the migration job with the environment variable _BPX_SHAREAS explicitly set to NO. You can set this either in the system profile (`/etc/profile`) or in the user profile for the user running the migration job. Use the following entry to set this variable to NO:

  ```
  export _BPX_SHAREAS = NO
  ```

  After the migration job completes, you can update the profile to reset _BPX_SHAREAS to its original value.

- Failures occur during server startup after migration.

  Review the instructions that were generated when you created the migration jobs. Verify that the JCL procedures have been copied over correctly to your PROCLIB, the RACF definitions have been created, and the Version 8.0 libraries have been authorized. Make sure that the daemon process associated with your cell is at the appropriate level. The daemon process must be at the highest WebSphere Application Server for z/OS version level of all servers that it manages within the cell.

  After migrating to a Version 8.0 cell that contains or interoperates with Version 6.x nodes that are not at Version 6.0.2.11 or later, the cluster function might fail. When starting these Version 6.x application servers, you might see the following problems:

  – You might see a first failure data capture (FFDC) log that shows a ClassNotFoundException error message. This exception is thrown from the RuleEtiquette.runRules method and looks something like the following example:

    ```
    Exception = java.lang.ClassNotFoundException
    Source = com.ibm.ws.cluster.selection.SelectionAdvisor.<init>
    probeid = 133
    Stack Dump = java.lang.ClassNotFoundException: rule.local.server
    at java.net.URLClassLoader.findClass(URLClassLoader.java(Compiled Code))
    at com.ibm.ws.bootstrap.ExtClassLoader.findClass(ExtClassLoader.java:106)
    at java.lang.ClassLoader.loadClass(ClassLoader.java(Compiled Code))
    at java.lang.ClassLoader.loadClass(ClassLoader.java(Compiled Code))
    at java.lang.Class.forName1(Native Method)
    at java.lang.Class.forName(Class.java(Compiled Code))
    at com.ibm.ws.cluster.selection.rule.RuleEtiquette.runRules(RuleEtiquette.java:154)
    at com.ibm.ws.cluster.selection.SelectionAdvisor.handleNotification(SelectionAdvisor.java:153)
    at com.ibm.websphere.cluster.topography.DescriptionFactory$Notifier.run(DescriptionFactory.java:257)
    at com.ibm.ws.util.ThreadPool$Worker.run(ThreadPool.java:1462)
    ```

– You might see a java.io.IOException that looks something like the following example:

```
Exception = java.io.IOException
Source = com.ibm.ws.cluster.topography.DescriptionManagerA. update probeid = 362
Stack Dump = java.io.IOException
at com.ibm.ws.cluster.topography.ClusterDescriptionImpl.importFromStream(ClusterDescriptionImpl.java:916)
at com.ibm.ws.cluster.topography.DescriptionManagerA.update(DescriptionManagerA.java:360)
Caused by: java.io.EOFException
at java.io.DataInputStream.readFully(DataInputStream.java(Compiled Code))
at java.io.DataInputStream.readUTF(DataInputStream.java(Compiled Code))
at com.ibm.ws.cluster.topography.KeyRepositoryImpl.importFromStream(KeyRepositoryImpl.java:193)
```

During migration, Version 8.0 cluster information is distributed throughout the cell. Version 6.x nodes that are not at Version 6.0.2.11 or later fail to read this information. To avoid this problem, upgrade all Version 6.x nodes that will be contained in or interoperating with a Version 8.0 cell to Version 6.0.2.11 or later before migrating your deployment managers to Version 8.0.

After migration, carefully review the job output and log files for errors.

Note: WebSphere Application Server provides an interactive problem control system (IPCS) verb exit to help you to format information from dumps of WebSphere Application Server processes. This verb exit was named CBDATA, which was an alias of the real module name, in Version 6.x and earlier. In Version 8.0, that alias was removed. In Version 8.0 and later, therefore, you must use the real name of this verb exit, BBORDATA, instead of the alias.

If you migrate a node to Version 8.0 then discover that you need to revert back to Version 6.x, read “Rolling back environments” on page 117.

For current information available from IBM Support on known problems and their resolution, read the IBM Support page. IBM Support also has documents that can save you time gathering information needed to resolve this problem. Before opening a PMR, read the IBM Support page.

New ports that are registered on a migrated Version 8.0 node agent include: WC_defaulthost, WC_defaulthost_secure, WC_adminhost, WC_adminhost_secure SIB_ENDPOINT_ADDRESS, SIB_ENDPOINT_SECURE_ADDRESS ,SIB_MQ_ENDPOINT_ADDRESS, SIB_MQ_ENDPOINT_SECURE_ADDRESS. These ports are not needed by the node agent, and can be safely deleted.

What to do next

If you did not find your problem listed, contact IBM support.

Chapter 12. IBM Suggests

Most of the following links will take you to information that is not part of the formal product documentation and is provided "as is." Some of these links go to non-IBM Web sites and are provided for your convenience only and do not in any manner serve as an endorsement by IBM of those Web sites, the material thereon, or the owner thereof.

© IBM Corporation 2003, 2005

Chapter 13. Migrating, coexisting, and interoperating

This section covers all aspects of migration, coexistence, and interoperability. Migrating is copying the configuration from a previous release of this product into a new release. Coexisting is running a new release of WebSphere Application Server on the same machine at the same time as you run an earlier release or running two installations of the same release of WebSphere Application Server on the same machine at the same time. Interoperating is exchanging data between two different systems, such as coexisting product installations.

© Copyright IBM Corp. 2011

Appendix. Directory conventions

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

Default product locations - z/OS

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*. Corresponding product variables are *was.install.root* and *WAS_HOME*.

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

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

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

  The default is /wasv8config/*cell_name*/*node_name*.

plug-ins_root
: Refers to the installation root directory for Web Server Plug-ins.

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 always be *app_server_root*/profiles/default because only the profile name "default" is used in WebSphere Application Server for z/OS.

smpe_root
: Refers to the root directory for product code installed with SMP/E or IBM Installation Manager.

  The corresponding product variable is *smpe.install.root*.

  The default is /usr/lpp/zWebSphere/V8R0.

© IBM Corporation 2005

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

APACHE INFORMATION. This information may include all or portions of information which IBM obtained under the terms and conditions of the Apache License Version 2.0, January 2004. The information may also consist of voluntary contributions made by many individuals to the Apache Software Foundation. For more information on the Apache Software Foundation, please see http://www.apache.org. You may obtain a copy of the Apache License at http://www.apache.org/licenses/LICENSE-2.0.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Intellectual Property & Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300
2455 South Road
Poughkeepsie, NY 12601-5400
USA
Attention: Information Requests

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

© Copyright IBM Corp. 2011

Trademarks and service marks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. For a current list of IBM trademarks, visit the IBM Copyright and trademark information Web site (www.ibm.com/legal/copytrade.shtml).

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.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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

© Copyright IBM Corp. 2011

Index

A

action 108, 118, 119
admin
  migration
    cell 118
    configuration 42, 52, 79
    federated node 119
    mapping 20
    stand-alone 121
    variables 70
application 20, 25, 38, 39, 41, 42, 49, 52, 58, 60, 62, 70, 76, 79, 86, 88, 108, 113, 121
application server 38, 39, 41, 49, 52, 58, 60, 62, 108, 121

C

cell 20, 65, 118
command 19, 20, 32, 35, 52, 79, 100, 101, 118, 119
configuration 19, 20, 25, 35, 39, 42, 52, 62, 65, 70, 76, 79, 88, 93, 98, 101, 108, 113, 118, 119, 121
configuration file 42, 70, 93
controller 108

D

daemon 88, 108
data 25, 31, 32, 39, 42, 52, 65, 70, 79, 93, 101, 108
database 113
dataset 31, 32, 39, 42, 52, 65, 70, 79, 93, 101, 108
debug 113
default 20, 35, 42, 49, 52, 70, 76, 79, 93, 98
deployment 19, 20, 35, 68, 69, 70, 76, 79, 87, 88, 108, 118
directory 20, 35, 42, 49, 52, 58, 62, 70, 76, 79, 86, 88, 93, 101, 105, 118
  installation
    conventions 151

E

ear 20
enterprise application 42, 52, 70, 79

F

feature pack 20
federated node 35, 91, 92, 93, 98, 100, 107, 108, 118, 119
file 20, 32, 42, 49, 52, 58, 62, 65, 70, 76, 79, 86, 88, 93, 98, 101, 105, 108

I

install 20, 42, 52, 70, 79
installation 20, 42, 70

J

jcl 42, 52, 62, 65, 70, 79, 88, 93, 101, 108
job statement 42, 70, 93
jvm 20

L

log 20, 62, 108, 113

M

management 26, 28, 29, 30, 31, 42, 70, 93
manager 19, 20, 35, 68, 69, 70, 76, 79, 87, 88, 108, 118
migration 19, 20, 25, 26, 28, 30, 31, 32, 39, 42, 49, 52, 58, 60, 62, 65, 68, 69, 79, 86, 88, 91, 92, 93, 98, 101, 105, 108, 113, 117, 119, 121
mount 42, 52, 70, 79, 88, 93, 108

P

parameters 35, 118
path 42, 52, 70, 79, 93
planning 24, 38, 65, 91
process 31, 42, 52, 70, 79, 93, 101, 108
product 19, 20, 24, 25, 42, 70, 93
profile 20, 35, 62, 88, 108
properties 20

R

response file 32, 42, 52, 58, 70, 79, 86, 93, 101, 105
run 32, 35, 108, 118, 119

S

server 19, 20, 35, 38, 39, 41, 42, 49, 52, 58, 60, 62, 70, 88, 93, 108, 121
servers 20, 38, 121

T

tools 19, 20, 25, 113
trace 35, 42, 70, 93

U

upgrade 19, 42, 93, 113

V

values 49, 58, 76, 86, 98, 105
variables 42, 52, 70, 79, 93, 101

© Copyright IBM Corp. 2011

W

web 20, 26, 28, 32, 39, 62, 88, 108, 113
websphere 26, 28, 32, 39, 62, 88, 108
wizard 25

X

xml 118

Z

zmmt 25, 32, 52, 58, 60, 79, 86, 87, 100, 101, 105, 107