IBM Product Connectivity Scenarios information center
Version 1 Release 0

Migrating from Version 6 to Version 8.5 of WebSphere Application Server and Version 7.5 of WebSphere MQ on Linux

IBM

Contents

Figures

Chapter 1. Scenario: Migrating from Version 6 to Version 8.5 of WebSphere Application Server and Version 7.5 of WebSphere MQ

Starting with an existing WebSphere® Application Server and WebSphere MQ Version 6 installation, this scenario leads you through the key tasks that are required to migrate to Version 8.5 of WebSphere Application Server and Version 7.5 of WebSphere MQ. This scenario focuses on the migration of the product connectivity, and does not provide detail about general migration of each product beyond what is needed to migrate the product connectivity.

This scenario was developed by using a sample WebSphere Application Server and WebSphere MQ installation. If you want to try the scenario, you can set up a copy of the sample installation as described in the scenario. You can use the sample applications that are provided with the scenario to verify your progress through each stage.

Note: For the post-migration configuration, this scenario applies to the use of WebSphere Application Server Version 8.5 with WebSphere MQ version 7.5.

Notes

This scenario applies to the use of WebSphere Application Server Version 8.5 with WebSphere MQ version 7.5.
> For the post-migration configuration, this scenario applies to the use of WebSphere Application Server Version 8.5 with WebSphere MQ version 7.5.

Optional information to help you learn while you implement the solution
> The scenario contains blocks of optional information marked by Why? or What else?. You do not need to read this information to complete the scenario, but might choose to learn more:

> **Why?** Describes *why* you are instructed to do something. For example:

> ┌─ **Why am I doing this?** ──────────────────────────┐
> │ You use the **Scope** property to set the level at which the │
> │ activation specification is visible. The cell scope is the highest │
> │ level, giving the activation specification the greatest visibility. │
> └──┘

> **What else?**
> > Describes *what else* you might do, or want to learn about, related to what you are reading in the main window. For example:

> ┌─ **What else might I do or be interested in?** ──────────┐
> │ You can also create activation specifications at other levels. │
> │ For example, if you have multiple servers you might create │
> │ an activation specification for each server, using the server │
> │ scope, so that you can specify different settings to be used for │
> │ each server. │
> └──┘

Tip: Some "Why?" and "What else?" information provides links that would take you to information resources outside the scenario. To complete a scenario, you do not need to follow such links; they are provided only as optional aids for your learning.

Related information:

This scenario in IBM Knowledge Center
This scenario in IBM Knowledge Center

Product web page
WebSphere Application Server product web page

Library page
WebSphere Application Server library page

Product web page
WebSphere MQ product web page

Library page
WebSphere MQ library page

Chapter 2. Planning the solution

Review the topics in this section to understand what is covered in this scenario, the reasons why a business might want to follow the scenario, the user roles involved, and an overview of the solution proposed by the scenario.

Assumptions

This scenario makes several assumptions about the system that you are using to set up and work with the sample IT configuration. These assumptions include the operating system and version of the products that you are using, and whether you have security configured for WebSphere MQ.

This scenario assumes the following points:
- You are using a Linux based operating system.
- You are using WebSphere Application Server and WebSphere MQ on one computer. However, if you are using them in different computers, you are still able to migrate them using the instructions with a few changes.

  Note: Clustering is not described in this scenario. Instructions are provided for installing a sample WebSphere MQ and WebSphere Application Server single server configuration that you can use as a starting point for trying out the scenario in the same way as it was originally developed.
- You are using the following versions of WebSphere MQ and WebSphere Application Server:
  - For the initial Version 6 sample configuration, you are using WebSphere MQ Version 6.0.2 with Fix Pack 12, and WebSphere Application Server Version 6.1.
  - For the post-migration configuration, you are using WebSphere MQ Version 7.5, and WebSphere Application Server Version 8.5.

  > **Why am I doing this?**
  > Earlier versions of the products might require configuration that is not covered in this scenario.

  Note: For the post-migration configuration, this scenario applies to the use of WebSphere Application Server Version 8.5 with WebSphere MQ version 7.5.
- You do not have security configured for WebSphere MQ. This scenario does not describe security configuration. However, if you do have security configured for WebSphere MQ you are still able to complete the scenario.
- You have root access to the computer that you are using.
- You use both the graphical user interfaces and the command-line equivalents of WebSphere MQ and WebSphere Application Server.

Business overview

A company wants to migrate an existing IT environment provided by WebSphere MQ Version 6 integrated with WebSphere Application Server Version 6 to an equivalent IT environment provided by WebSphere MQ Version 7.5 and WebSphere Application Server Version 8.5.

3

To date, company A has a business solution that uses a messaging infrastructure that is provided by WebSphere MQ Version 6. WebSphere MQ is connected to WebSphere Application Server Version 6 so that Java EE applications on WebSphere Application Server can consume and work with messages from WebSphere MQ. Through the WebSphere MQ messaging provider in WebSphere Application Server, Java EE applications can exploit the Java Message Service (JMS) with WebSphere MQ as an external provider of JMS messaging resources.

The company decides to migrate their business solution to WebSphere MQ Version 7.5 connected to WebSphere Application Server Version 8.5 to gain business value, including the following:

- Maintaining the software environment at a recent supported version
- Reducing software operation and maintenance costs
- Benefiting from the enhanced JMS provided by a re-architecture of WebSphere MQ to be a more-integrated, higher-performing JMS provider
- Using new and updated functions available in Version 7.5 of WebSphere MQ and Version 8.5 of WebSphere Application Server.

Related information:

➡ WebSphere Application Server features and benefits web page

User roles and interactions

This scenario focuses on migrating an existing solution that involves migrating WebSphere Application Server from Version 6 to Version 8.5, and WebSphere MQ from Version 6 to Version 7.5. The user roles and interactions described are the main ones involved in migration activities for the product connectivity.

Migrate the solution

Migrate the solution for production use, by installing, configuring, and testing new versions of the components that provide the solution.

Administrator
> The Administrator installs and configures the components that support the solution, in this case WebSphere MQ and WebSphere Application Server. The business might have a different Administrator for each product, or one Administrator might perform installation and configuration for all components. If the Administrators are different, they might need to exchange information. For example, in this scenario the WebSphere MQ administrator must give the name of the queue and queue manager to the WebSphere Application Server administrator.
>
> The Administrator also deploys the software components given to them by the Developer.
>
> The maintenance of components used for messaging between WebSphere Application Server and WebSphere MQ has changed:
>
> - In Version 6, the WebSphere MQ messaging provider installed as part of WebSphere Application Server includes a cut-down version of the WebSphere MQ Version 6 JMS Client, which Java™ EE applications can use to create client-mode connections to either a local or remote WebSphere MQ queue manager.
> - In Version 6, the WebSphere MQ messaging provider uses WebSphere MQ client classes for JMS included in WebSphere Application Server.

Therefore, the WebSphere Application Server administrator is responsible for maintaining those classes, by applying WebSphere Application Server fix packs.

- In Version 7.5, the WebSphere MQ messaging provider uses the WebSphere MQ Resource Adapter included in WebSphere Application Server. Therefore, the WebSphere Application Server administrator is responsible for maintaining that resource adapter, by applying WebSphere Application Server fix packs.

Test Implementer

The Test Implementer runs the tests to validate the solution and that the solution is ready for production use. For example, can the solution be started, stopped, backed up, and recovered after a system failure as well as be maintained?
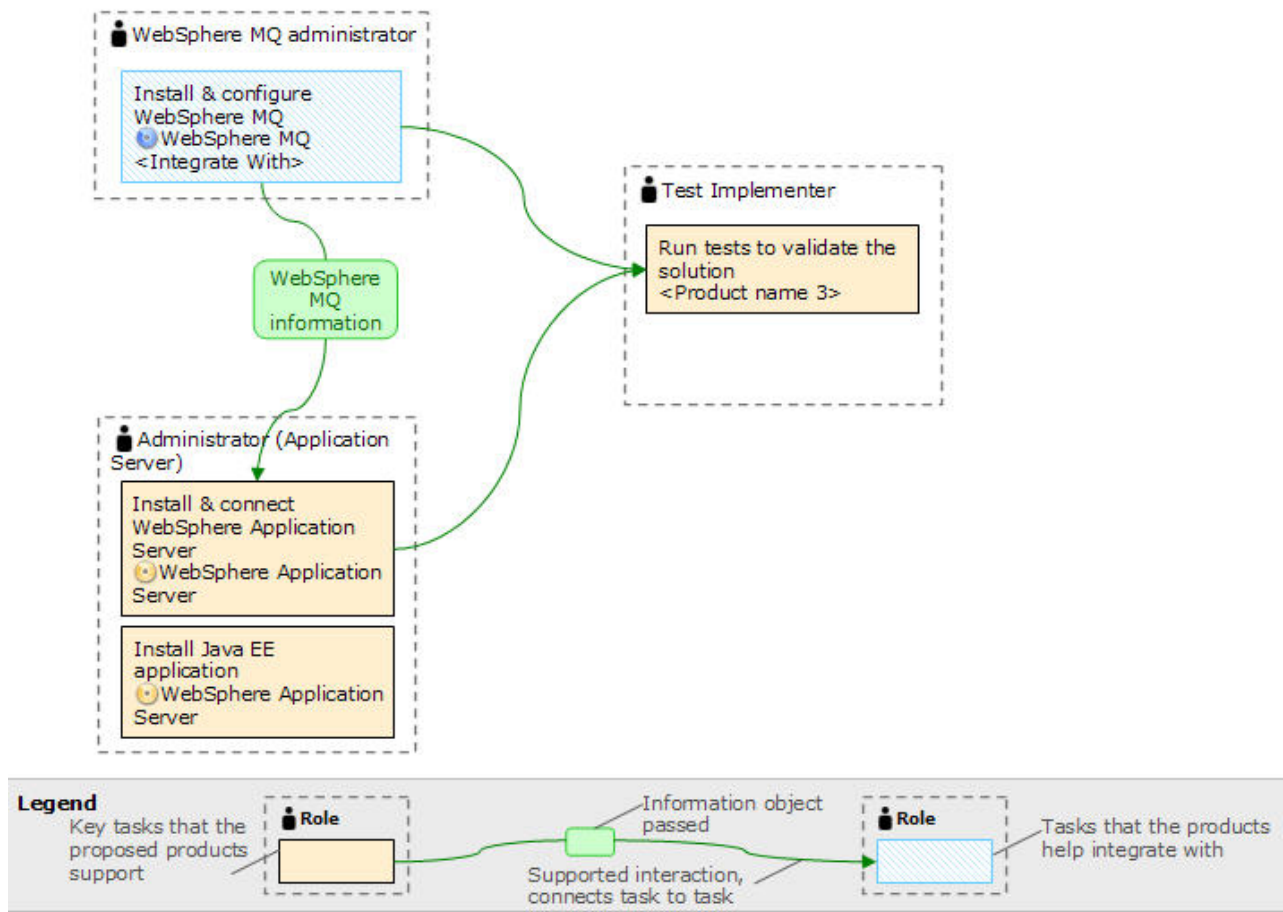
Figure 1. User role interactions to deploy the solution

Technical solution

This scenario describes one way of migrating an existing WebSphere Application Server and WebSphere MQ solution from Version 6 to Version 8.5 of WebSphere Application Server and Version 7.5 of WebSphere MQ.

Overview: Initial IT configuration

Company A uses an existing IT configuration provided by WebSphere MQ Version 6 integrated with WebSphere Application Server Version 6. This scenario describes migrating that initial IT configuration to an equivalent IT configuration provided by Version 7.5 of WebSphere MQ and Version 8.5 of WebSphere Application Server.

In this scenario, the initial configuration that is to be migrated to an equivalent configuration provided by Version 7.5 of WebSphere MQ and Version 8.5 of WebSphere Application Server includes a Java EE application on WebSphere Application Server Version 6 that uses a message-driven bean (MDB) to consume and work with messages received on a queue provided by WebSphere MQ Version 6. The Java EE application also sends reply messages back to the requester client of the JMS application.

Tip: This scenario was developed by using a sample version of the initial IT configuration that is described in this topic. If you want to try the scenario, follow the instructions in "Creating a sample initial Version 6 IT configuration" on page 18 to set up a copy of the sample IT configuration.

The initial multi-product configuration includes several components that an administrator configures or uses, as shown in Figure 2 on page 7, and described after the figure.
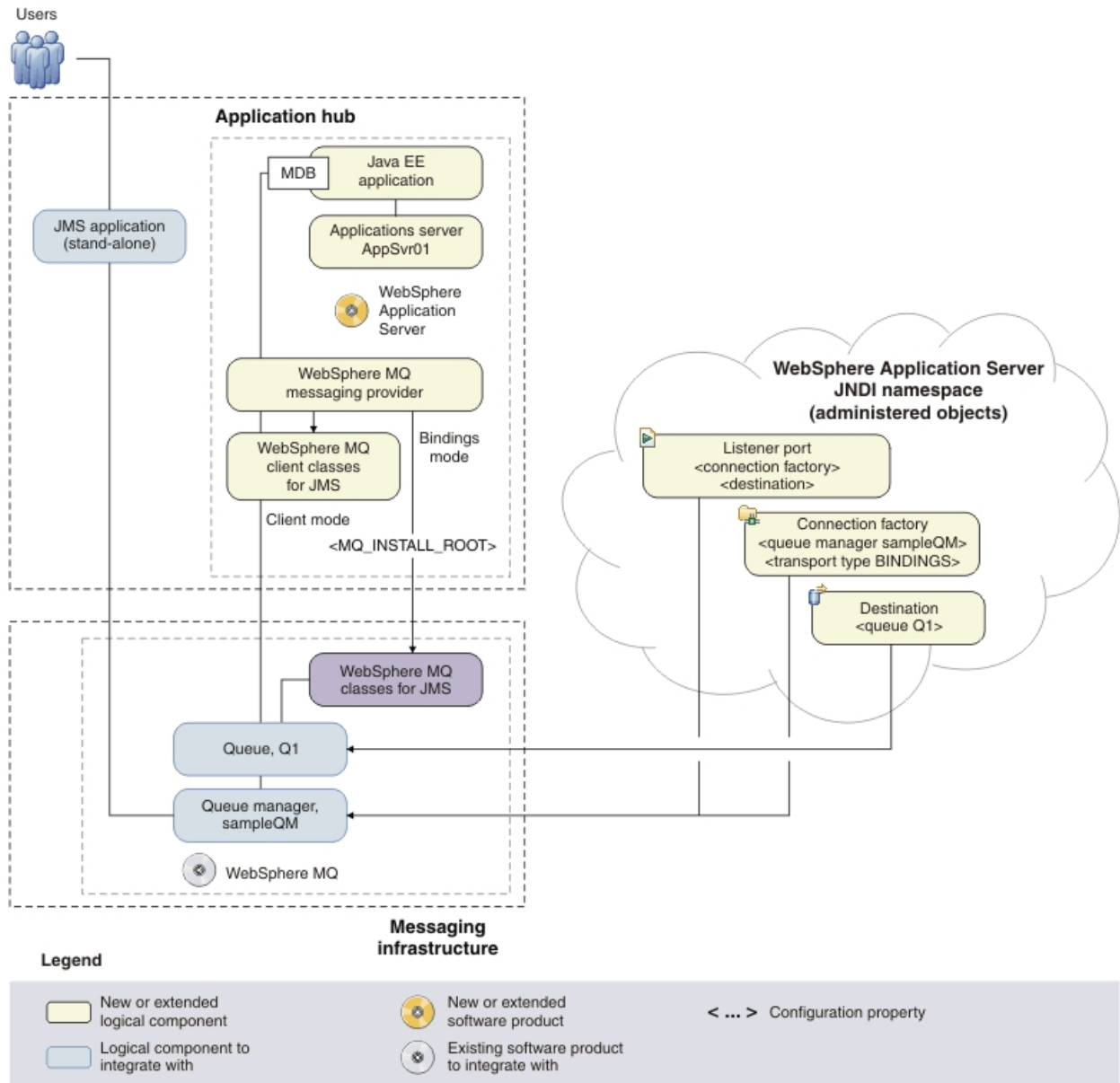
Figure 2. The initial multi-product configuration. The high-level logical topology diagram for software functions delivered by integrating WebSphere Application Server into the initial WebSphere MQ messaging infrastructure. This logical topology shows the new functions as new or extended logical components, integrated with the software functions and products from the initial logical topology.

WebSphere MQ as a *messaging provider* for WebSphere Application Server

The WebSphere MQ messaging provider in WebSphere Application Server makes JMS messaging available to WebSphere Application Server applications by using the existing capabilities in the WebSphere MQ environment.

> **Why am I doing this?**
> WebSphere Application Server applications can interact with WebSphere MQ destinations to send and receive messages in the same way as any JMS application in the WebSphere MQ environment.

> **What else might I do or be interested in?**
>
> - You can connect a WebSphere MQ network to a service integration bus within WebSphere Application Server, by using WebSphere MQ links. A WebSphere MQ link provides support for sender-receiver channels between the service integration bus and a WebSphere MQ queue manager or queue-sharing group. This option requires more complex configuration in WebSphere Application Server; you must configure a service integration bus and messaging engines.
>
> - You can add a WebSphere MQ server (representing a queue manager or queue-sharing group) as a member of a service integration bus within WebSphere Application Server. A WebSphere MQ server provides a direct client or bindings connection between a service integration bus and queues on a WebSphere MQ queue manager. This option also requires you to configure a service integration bus and messaging engines in WebSphere Application Server.

Bindings mode transport between WebSphere Application Server and WebSphere MQ

In this scenario, bindings mode transport is used for communication between WebSphere Application Server and WebSphere MQ. Bindings mode transport uses WebSphere MQ classes for JMS, included in WebSphere MQ.

> > **Why am I doing this?**
> >
> > Bindings mode usually has better performance than client mode because communication between the two products is done using shared memory. Bindings mode can be used only when WebSphere Application Server and WebSphere MQ are installed on the same computer and operating system image.

> > **What else might I do or be interested in?**
> >
> > Client mode is used for communication between WebSphere Application Server and WebSphere MQ installed on different computers, and can be used for communication if both products are installed on the same computer and operating system image. Client mode transport generally offers lower performance than bindings mode. By default, client mode transport uses the WebSphere MQ client classes for JMS, included in WebSphere Application Server.

WebSphere MQ client classes for JMS, included in WebSphere Application Server

The WebSphere MQ client classes for JMS are installed as part of WebSphere Application Server into `WAS_INSTALL_ROOT\lib\WMQ\ java\lib`. Using these classes, JMS applications on WebSphere Application Server can connect to WebSphere MQ in client mode.

The WebSphere MQ client classes for JMS included in WebSphere Application Server are maintained by applying WebSphere

Application Server fix packs. For information about which version of WebSphere MQ is included, see Which version of WebSphere MQ is shipped with WebSphere Application Server.

WebSphere MQ classes for JMS, included in WebSphere MQ
The WebSphere MQ classes for JMS installed as part of WebSphere MQ include Java Native Interface (JNI) libraries that are needed for bindings mode transport. When WebSphere MQ is installed on the same computer as WebSphere Application Server, JMS applications on WebSphere Application Server can use these classes to connect to WebSphere MQ in bindings mode. The location of the WebSphere MQ classes for JMS is specified by the WebSphere Application Server environment variable MQ_INSTALL_ROOT.

The WebSphere MQ classes for JMS included in WebSphere MQ are maintained by applying WebSphere MQ fix packs.

Java EE application
The application consumes and works with messages on the WebSphere MQ queue, Q1. This application runs on an *application server* of the WebSphere Application Server product that has been installed and connected with WebSphere MQ.

The sample application in this scenario provides a *message-driven bean (MDB)* as an asynchronous message consumer. When a message arrives at the queue, the MDB automatically processes the message without the application having to explicitly poll the queue.

Why am I doing this?
MDBs are activated by the EJB container in WebSphere Application Server on receipt of a message. A typical MDB performs messaging functions, and calls one or more session beans to perform business logic. Because of this separation of function, you can easily change and reuse units of business logic without affecting the messaging function of the application.

Application server
A server program in WebSphere Application Server that provides the environment in which Java EE application programs run.

WebSphere Application Server JNDI namespace
WebSphere Application Server includes a name server that provides access to the following JMS administered objects through the Java Naming and Directory Interface (JNDI). The use of JNDI, the connection factory, and the destination, are the same as described for the initial IT configuration in "Overview: Initial IT configuration" on page 6.

Listener port, PiShopReplyMDBLstener
A listener port defines the association between a connection factory, destination, and a deployed MDB. The listener port provides the configuration necessary for the MDB to listen for messages arriving at a destination. The listener port processes inbound messages to the MDB.

> **Why am I doing this?**
> The WebSphere MQ messaging provider in WebSphere Application Server Version 6 does not have a Java EE Connector architecture (JCA) 1.5 resource adapter. If you want to use MDBs with a messaging provider that does not have a JCA 1.5 resource adapter, you cannot use activation specifications and therefore you must configure your MDBs against a listener port.

> **What else might I do or be interested in?**
> Activation specifications are part of the JCA 1.5 standard, which provides a standard way to integrate JMS providers, such as WebSphere MQ Version 7, with Java EE application servers such as WebSphere Application Server.

Connection factory, myCF

A JMS connection factory object defines a set of standard configuration properties for connections. An application uses a connection factory to create a connection to WebSphere MQ.

If your application uses an MDB, as the sample application does, the connection factory is used for outbound messages only; an inbound message is received by the listener port, which passes the message to a new instance of a message-driven bean for processing.

In this scenario, when creating JMS objects in WebSphere Application Server, you specify full details rather than use a *client channel definition table (CCDT)* that has been exported from WebSphere MQ.

> **Why am I doing this?**
> A CCDT makes configuring objects in WebSphere Application Server easier because it contains information that is required to connect to WebSphere MQ. If you do not specify a CCDT, you must enter this information yourself. This scenario does not use a CCDT so that you can see more clearly what information is required to make a connection.

> **What else might I do or be interested in?**
> A CCDT is particularly useful if your client applications might need to connect to a number of alternative queue managers. This scenario does not cover that situation.

Overview: The delivered logical topology

The company has migrated WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5.

The delivered IT configuration includes several components that an administrator configures or uses differently in newer versions when compared to Version 6, as shown in Figure 3. The description of components in this topic builds on the description given for the equivalent Version 6 components in "Overview: Initial IT configuration" on page 6.
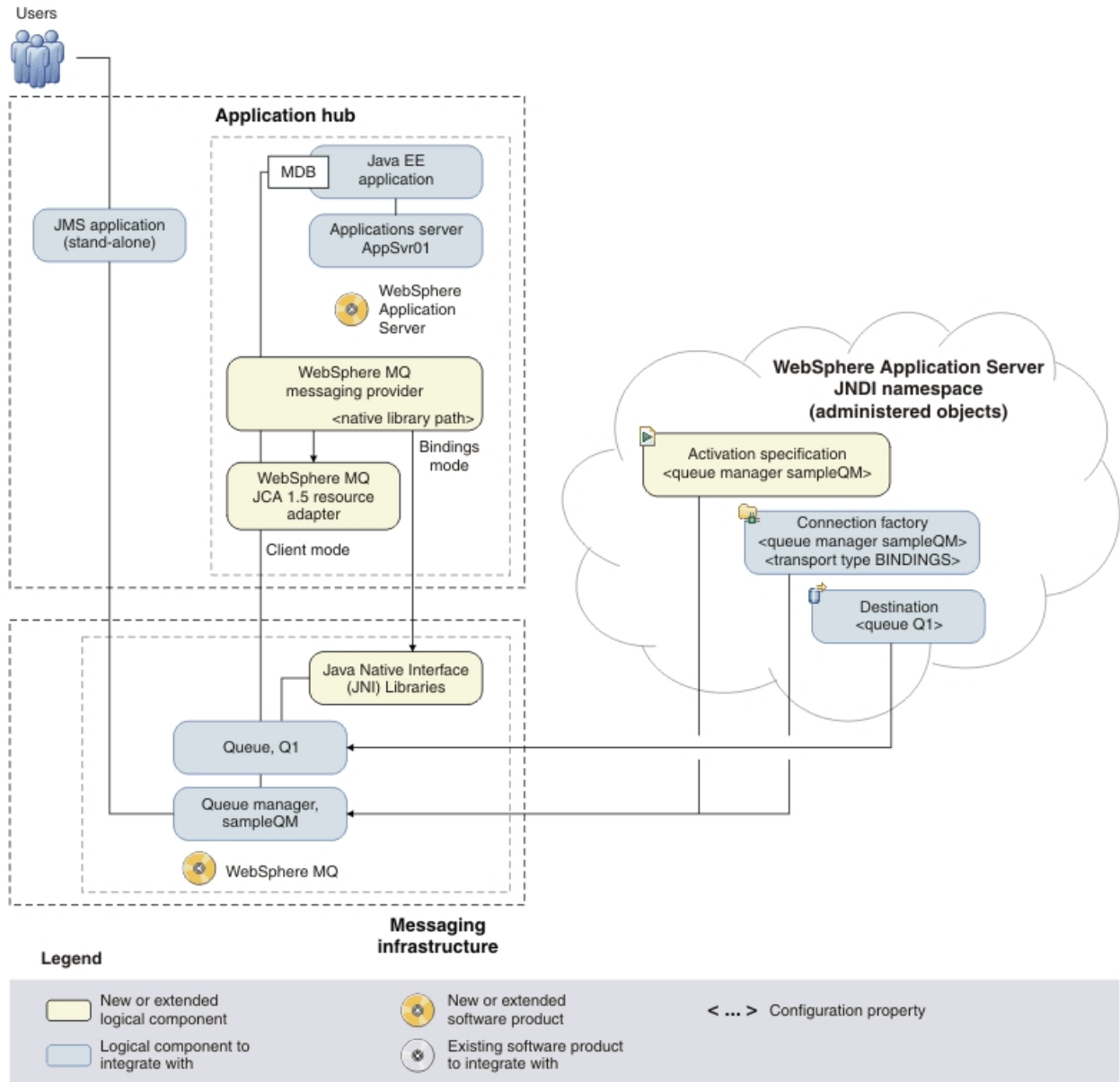
Figure 3. Delivered logical topology. The high-level logical topology diagram for software functions delivered by migrating WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5. This logical topology shows the new functions as new or extended logical components, integrated with the software functions and products from the initial logical topology.

WebSphere MQ as a *messaging provider* **for WebSphere Application Server**
> The WebSphere MQ messaging provider in WebSphere Application Server makes JMS messaging available to WebSphere Application Server applications by using the existing capabilities in the WebSphere MQ environment.

Bindings mode transport between WebSphere Application Server and WebSphere MQ

In this scenario, bindings mode transport is used for communication between WebSphere Application Server and WebSphere MQ. Bindings mode transport uses WebSphere MQ classes for JMS, included in WebSphere MQ.

> **Why am I doing this?**
> Bindings mode usually has better performance than client mode because communication between the two products is done using shared memory. Bindings mode can be used only when WebSphere Application Server and WebSphere MQ are installed on the same computer and operating system image.

> **What else might I do or be interested in?**
> Client mode is used for communication between WebSphere Application Server and WebSphere MQ installed on different computers, and can be used for communication if both products are installed on the same computer and operating system image. Client mode transport generally offers lower performance than bindings mode. By default, client mode transport uses the WebSphere MQ client classes for JMS, included in WebSphere Application Server.

WebSphere MQ JCA 1.5 Resource Adapter, included in WebSphere Application Server

WebSphere MQ Version 7.5 classes for JMS are built in to WebSphere Application Server Version 8.5 as a JCA 1.5 resource adapter, wmq.jmsra.rar. By configuring activation specifications and connection factories to use the WebSphere MQ messaging provider, JMS applications on WebSphere Application Server can use the resource adapter to connect to WebSphere MQ in client mode.

The WebSphere MQ JCA 1.5 resource adapter included in WebSphere Application Server is maintained by applying WebSphere Application Server fix packs. For information about which version of the WebSphere MQ JCA resource adapter is included, see Which version of WebSphere MQ is shipped with WebSphere Application Server.

WebSphere MQ classes for JMS, included in WebSphere MQ

The Java Native Interface (JNI) libraries that are needed for bindings mode transport are included in WebSphere MQ. When WebSphere MQ is installed on the same computer as WebSphere Application Server, JMS applications on WebSphere Application Server can use these classes to connect to WebSphere MQ in bindings mode. The location of the WebSphere MQ native libraries is specified in WebSphere Application Server, by the WebSphere MQ messaging provider property Native library path.

> **Why am I doing this?**
>
> The WebSphere MQ messaging provider property **Native library path** is used instead of the *MQ_INSTALL_ROOT* variable, which is no longer required and by default is not migrated from Version 6.

The WebSphere MQ native libraries are maintained by applying WebSphere MQ fix packs.

WebSphere Application Server JNDI namespace

WebSphere Application Server includes a name server that provides access to the following JMS administered objects through the Java Naming and Directory Interface (JNDI). The use of JNDI, the connection factory, and the destination, are the same as described for the initial IT configuration in "Overview: Initial IT configuration" on page 6.

Activation specification, myActSpec

A JMS activation specification can be associated with one or more message-driven beans (MDBs) and provides the configuration necessary for them to listen for messages arriving at a destination. Activation specifications process inbound messages to the MDB.

> **Why am I doing this?**
>
> Activation specifications are part of the Java EE Connector architecture (JCA) 1.5 standard. JCA 1.5 provides a standard way to integrate JMS providers, such as WebSphere MQ, with Java EE application servers such as WebSphere Application Server.

> **What else might I do or be interested in?**
>
> Use of listener ports is an older configuration method for MDBs to listen for messages arriving at a destination. The use of listener ports is stabilized in WebSphere Application Server Version 7 and later. If you are using WebSphere Application Server for the first time, avoid using listener ports.

Resources for this scenario

This scenario provides sample resources that you can use to verify each task phase. Instructions for downloading resources, and creating resources, are given at appropriate steps in the tasks for this scenario.

Tip: You can download resources when instructed in the scenario tasks, or before then when you read this topic.

Resources for Version 6

1. The following download files are required for "Creating a sample initial Version 6 IT configuration" on page 18:
   - To download the sample JMS application package, click the following link and save the file to the machine that hosts WebSphere Application Server:

sampleJMSApp_V6.zip, then extract the contents. The package contains a
sample JMS application `.jar` file and shell scripts for running the
application.

- To download the sample MDB application, click the following link and save
  the file to the machine that hosts WebSphere Application Server:
  sampleMDBv6.ear.

2. When Configuring the JNDI namespace and administered objects, you create
   the following resources by using the command line:

- A queue manager, `sampleQM`, for the sample application.
- The JNDI namespace context to use a directory on the local file system
  `/JNDI-Directory`:
- A connection factory administered object, `myCF`, to connect to the base queue
  manager, `sampleQM`.
- A destination administered object, `Q1`, for the queue, `myQueue`.
- The WebSphere MQ queue, `myQueue`.

3. When Configuring WebSphere Application Server Version 6, you create the
   following resources in the WebSphere Application Server administrative
   console:

- A listener port, `PiShopReplyMDBListener`, to find a connection factory and
  destination configured in the WebSphere Application Server JNDI
  namespace.
- A connection factory, `PiShopReplyCF`, with JNDI name `jms/PiShopReplyCF`,
  configured to connect to the queue manager `sampleQM`.
- A queue, `wasQueue`, with the JNDI name `jms/wasQueue`, WebSphere MQ
  queue, `Q1`.

Resources for Version 7 & 8

1. The following download files are required for "Option 1: Migrating WebSphere
   Application Server before WebSphere MQ" on page 39 and "Option 2:
   Migrating WebSphere MQ before WebSphere Application Server" on page 68:

- Sample JMS application package

  To download the sample JMS application package, click the following link
  and save the file to the machine that hosts WebSphere Application Server:
  sampleJMSApp_V7.zip, then extract the contents. The package contains a
  sample JMS application `.jar` file and shell scripts for running the
  application.

- Sample MDB application

  To download the sample MDB application, click the following link and save
  the file to the machine that hosts WebSphere Application Server:
  sampleMDB.ear

2. Resources created on Version 7.5 of WebSphere MQ and Version 8.5 of
   WebSphere Application Server during this scenario:

- When migrating your configuration, you create the same resources as for
  Version 6, to show that the applications and resources work on the newer
  versions.
- On WebSphere Application Server, you can convert the listener port to an
  activation specification. For more information, see Migrating WebSphere
  Application Server listener ports to activation specifications.

  Note: No additional resources are created in WebSphere MQ Explorer for this
  stage of the scenario.

Choosing which product to migrate first

When migrating WebSphere Application Server to Version 8.5 and WebSphere MQ to Version 7.5, you cannot migrate both products simultaneously. You must migrate one product before the other. Your choice depends on several considerations, and includes a period of time with an intermediate migration mode configuration.

You need to understand any functional and performance implications of running with an intermediate migration mode configuration.

Considerations for migrating WebSphere Application Server to Version 8.5 before WebSphere MQ

You need to understand several considerations for choosing to migrate WebSphere Application Server to Version 8.5 before WebSphere MQ, and the consequences of running with an intermediate migration mode configuration of WebSphere Application Server Version 8.5 with WebSphere MQ Version 6.

Use the intermediate migration mode configuration only as a short-term solution

WebSphere Application Server Version 8.5 uses its WebSphere MQ resource adapter to connect to a queue manager running on WebSphere MQ Version 6, so Version 7 features of WebSphere MQ are not available. As a result, the WebSphere MQ classes for JMS in the resource adapter run in migration mode, which has a performance and function cost.

This migration mode configuration facilitates migration from Version 6 to Version 8.5, but should not be considered as a long term solution if performance is important.

For more information about the performance impact, see the section "Avoiding running in Migration/Compatibility Mode" in the SupportPac MP07: WebSphere MQ - JMS V7 Performance Evaluations

Migration mode is set automatically.

Considerations for migrating WebSphere Application Server first

Have a minimum fix pack level before starting migration.
> In general, you should be at the latest fix pack level to benefit from the latest maintenance available. In some circumstances, you need to be at a minimum fix pack level:
>
> - For WebSphere Application Server Version 8.5 to connect in bindings mode to WebSphere MQ Version 6, WebSphere MQ must be at level 6.0.1.1 or later. This scenario was developed by using WebSphere MQ Version 6.0.2 with Fix Pack 12 (6.0.2.12)
> - This scenario was developed by using WebSphere Application Server Version 6.1.

Install WebSphere Application Server Version 8.5, then migrate the Version 6 configuration to Version 8.5
> You can install WebSphere Application Server Version 8.5 to coexist with WebSphere Application Server Version 6.

Considerations for migrating WebSphere MQ to Version 7.5 before WebSphere Application Server

You need to understand several considerations for choosing to migrate WebSphere MQ to Version 7.5 before WebSphere Application Server, and the consequences of running with an intermediate migration mode configuration of WebSphere MQ Version 7.5 with WebSphere Application Server Version 6.

You can use the intermediate migration mode configuration as a long-term solution

WebSphere Application Server Version 6 uses WebSphere MQ Version 7.5 JMS Classes to connect to a queue manager running on WebSphere MQ Version 7.

This migration mode configuration can be used as a long-term solution if it meets business requirements. This configuration is stable and supported.

Considerations for migrating WebSphere MQ first

Can migrate WebSphere MQ only in place
> MQ_INSTALL_ROOT will then point to WebSphere MQ Version native libraries

> For bindings connection, the application server must also be configured to locate the WebSphere MQ V7 native libraries by using the JVM argument -Djava.library.path set to the location of the libraries. Use the property on the Java virtual machine settings page of the administrative console (see http://www14.software.ibm.com/webapp/wsbroker/redirect?version=pix &product=was-nd-dist&topic=urun_rconfproc_jvm)

Changes to WebSphere MQ classes for JMS
> You might encounter JMS application migration problems, as a result of changes to the WebSphere MQ classes for JMS. If your solution requires connection to a V7.5 queue manager, you might be able to bypass some problems by continuing to use the version 6.0 WebSphere MQ classes for JMS. Alternatively, install the version 6 WebSphere MQ client on the same server as the application server and set $MQ_INSTALL_ROOT to the WebSphere MQ installation directory. Connect the JMS application to the version 7.5 queue manager in client mode.

Have a minimum fix pack level before starting migration.
> In general, you should be at the latest fix pack level to benefit from the latest maintenance available. In some circumstances, you need to be at a minimum fix pack level:

> • This scenario was developed by using WebSphere Application Server Version 6.1.

For client mode applications, everything is the same as for the starting environment.

Chapter 3. Implementing the solution

Implementing the solution in this scenario involves migrating WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5.

Before you begin

If you want to try out the scenario, you first need to follow the instructions to set up a copy of the sample messaging infrastructure as described in "Creating a sample initial Version 6 IT configuration" on page 18. This sample configuration is based on WebSphere Application Server and WebSphere MQ Version 6.

About this task
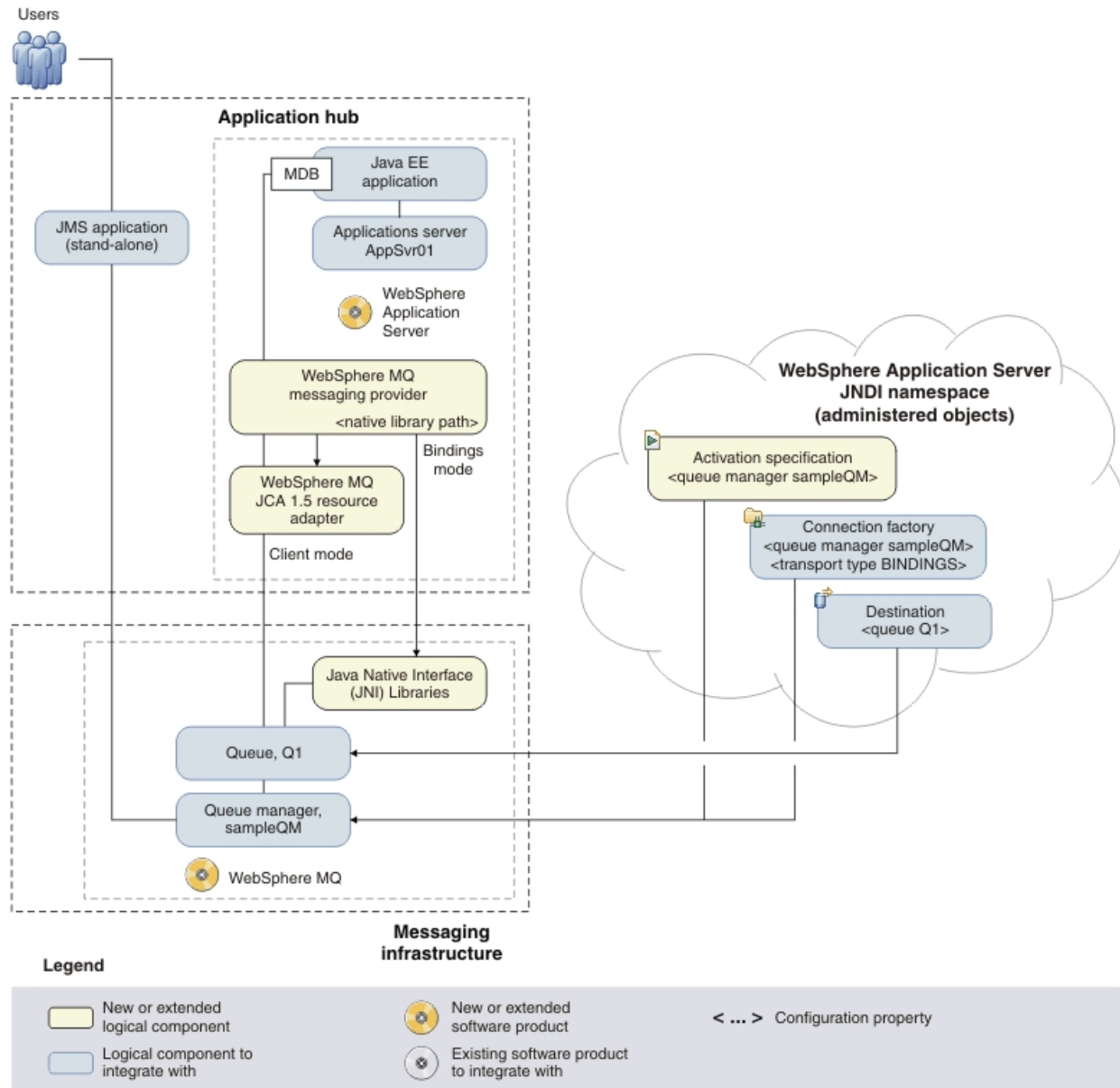
Figure 4. Delivered logical topology, including WebSphere Application Server. The high-level logical topology diagram for new software functions delivered by integrating WebSphere Application Server into the initial WebSphere MQ messaging infrastructure. This delivered logical topology shows the new functions as new or extended logical components, integrated with the software functions and products from the initial logical topology.

After connecting WebSphere Application Server to WebSphere MQ, you can use the connection by deploying a Java EE application to run on WebSphere Application Server. This scenario describes use of a sample Java EE application that is deployed onto WebSphere Application Server and used to verify the solution.

Creating a sample initial Version 6 IT configuration

This scenario was developed using a sample initial IT configuration. Follow the instructions to set up this sample configuration to try out the scenario in the same way as it was originally developed.

Installing WebSphere MQ Version 6.0

Create a user and group to install WebSphere MQ with. Then, install WebSphere MQ and apply the fix pack by using the command line.

Before you begin

In order for the sample MDB application to connect to WebSphere MQ, the minimum required level for WebSphere MQ is Version 6.0.1.1. This scenario was tested using WebSphere MQ Version 6.0 with Fix Pack 6.0.2.12 on Red Hat 64-bit Linux.

Before starting this task, complete the following checks:

- You must have local root access to install WebSphere MQ.
- Ensure that the host name does not contain any spaces.
- You must have permission to create a user and group on the host.
- Ensure that you have sufficient disk space, up to 600 MB to install WebSphere MQ for Linux.

If you want to investigate the requirements for installing WebSphere MQ in more detail, see System Requirements for IBM® MQ.

You can download WebSphere MQ V6.0 Fix Pack 6.0.2.12 from IBM support page.

Procedure

1. Log in as the root user.
2. Create a user and group called **mqm**.
   a. Create the **mqm** user, in a terminal enter `useradd mqm`.
   b. Add the **mqm** user to the **mqm** group, enter `usermod -g mqm mqm`.
3. Install WebSphere MQ 6.0.
   a. Insert the WebSphere MQ for Linux Server DVD into the DVD drive.
   b. Accept the WebSphere MQ license. In a terminal, navigate to the root directory of the DVD drive. Enter `./mqlicense.sh -accept`. The following message is displayed:

```
Licensed Materials - Property of IBM

 5724-H72

 (C) Copyright IBM Corporation 1994, 2019 All rights reserved.

US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.



Agreement accepted:  Proceed with install.
```

   c. Install the prerequisite of WebSphere MQ. Enter `rpm -ivh gsk7bas*` The following message is displayed:

```
Preparing...              ######################################### [100%]
   1:gsk7bas              ######################################### [ 50%]
   2:gsk7bas64            ######################################### [100%]
```

   d. Install the WebSphere MQ packages. Enter the following command:

```
rpm -ivh MQSeriesRuntime-*.rpm MQSeriesServer-*.rpm MQSeriesSamples-*.rpm MQSeriesJava-*.rpm
```

The following is displayed throughout the installation process:

```
Preparing...                  ######################################### [100%]
   1:MQSeriesRuntime           ######################################### [ 25%]
   2:MQSeriesServer            ######################################### [ 50%]
   3:MQSeriesJava              ######################################### [ 75%]
   4:MQSeriesSamples           ######################################### [100%]
```

WebSphere MQ is successfully installed to `/opt/mqm` directory.

4. Install WebSphere MQ V6.0 Fix Pack.

   a. Extract the archive file of fix pack. Enter the command:

```
tar -zxvf 6.0.2-WS-MQ-LinuxX64-FP0012.tar.gz
```

   b. Find out which packages are installed on your machine. Enter the following command:

```
rpm -qa | grep MQSeries
```

   If you have a minimum WebSphere MQ installation for this scenario, the message will be like:

```
MQSeriesJava-6.0.1-0.x86_64
MQSeriesSamples-6.0.1-0.x86_64
MQSeriesRuntime-6.0.1-0.x86_64
MQSeriesServer-6.0.1-0.x86_64
```

   c. Install all available updates for the packages you have on your system:

```
rpm -ivh MQSeriesRuntime-*.rpm MQSeriesServer-*.rpm MQSeriesSamples-*.rpm MQSeriesJava-*.rpm
```

   Note: You must install all packages in the fix pack that apply to those packages that are currently installed on your system.

   d. Repeat Step 4b, and you will see that the packages of WebSphere MQ are now at level 6.0.2.12:

```
MQSeriesSamples-U842215-6.0.2-12.x86_64
MQSeriesJava-6.0.1-0.x86_64
MQSeriesSamples-6.0.1-0.x86_64
MQSeriesRuntime-U842215-6.0.2-12.x86_64
MQSeriesRuntime-6.0.1-0.x86_64
MQSeriesJava-U842215-6.0.2-12.x86_64
MQSeriesServer-6.0.1-0.x86_64
MQSeriesServer-U842215-6.0.2-12.x86_64
```

You have successfully applied the fix pack.

Results

You have installed WebSphere MQ and are ready to configure objects such as queues and queue managers.

What to do next

Follow the instructions in "Configuring the JNDI namespace and administered objects" to configure WebSphere MQ for use by the sample application.

Configuring the JNDI namespace and administered objects

Configure the JNDI namespace to administer objects that the sample application can use; a queue, two channels and a listener.

About this task

Before an application can retrieve administered objects from a JNDI namespace, an administrator must first create the administered objects. The administrator can use the WebSphere® MQ JMS administration tool or WebSphere MQ Explorer to create and maintain administered objects in a JNDI namespace.

This scenario demonstrates the following aspects of configuring WebSphere MQ:
- Use of a JNDI namespace located in a local file system. A file system is used, because it is the simplest JNDI mechanism for a sample scenario.

> **What else might I do or be interested in?**
> The JNDI namespace can be on a file system, Lightweight Directory Access Protocol (LDAP) server, or on another JNDI implementation. If you want to use a JNDI namespace on an LDAP server or another JNDI implementation, you must configure the JNDI namespace and modify the sample application to reference the JNDI namespace, as required by the implementation.

- Use of the WebSphere MQ Explorer to create administered objects in the JNDI namespace. The JMS application can look up the administered objects to connect to WebSphere MQ and access destinations with which to send or receive messages.

  Remember: In this example, you use the JMS Administered Objects functions of WebSphere MQ Explorer. The JMS administration support in WebSphere MQ Explorer is only available if you have installed the required WebSphere MQ Fix Pack.

In this task, you create the following objects in WebSphere MQ:

WebSphere MQ

Queue manager
sampleQM

Queue Q1

JMS

Connection
factory
<queue manager
sampleQM>

Destination
<queue Q1>

**WebSphere MQ
JNDI namespace**

Connection
factory
<queue manager
sampleQM>

Destination
<queue Q1>

Legend

< ... > Configuration property
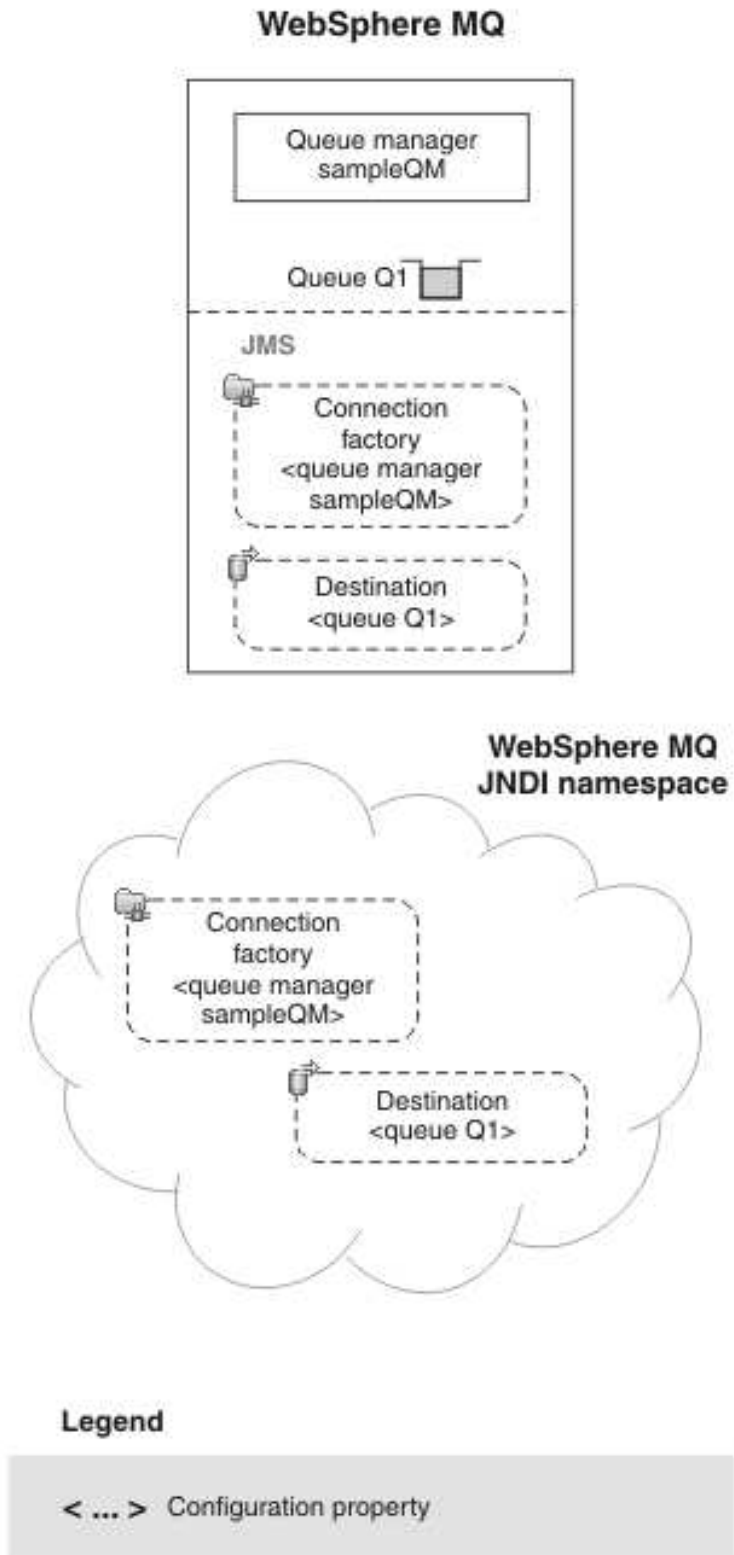
Figure 5. Objects created in WebSphere MQ

Procedure

1. Log in as the root user.

2. Create a JNDI-Directory. Navigate to the / directory, enter `mkdir JNDI-Directory`.

3. Log in as the mqm user.

4. Create a queue manager.

   a. Navigate to the /opt/mqm/bin directory.

   b. Enter `./crtmqm -ld /var/mqm/log sampleQM`. The following message is displayed:

```
WebSphere MQ queue manager created.
Creating or replacing default objects for sampleQM.
Default objects statistics : 40 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
```

5. Start the queue manager, enter `./strmqm sampleQM`. The following message is displayed:

```
WebSphere MQ queue manager 'sampleQM' starting.
5 log records accessed on queue manager 'sampleQM' during the log replay phase.
Log replay for queue manager 'sampleQM' complete.
Transaction manager state recovered for queue manager 'sampleQM'.
WebSphere MQ queue manager 'sampleQM' started.
```

6. Create a configuration file to connect WebSphere MQ to the initial context of the *JNDI namespace*.

   a. Create a file called `MyAdmin.config`. Navigate to the /opt/mqm/java/bin directory. Enter `vi MyAdmin.config`.

   > **Why am I doing this?**
   > This file creates a separate configuration specifically for JMS administration of the administered objects in the JNDI Namespace.

   b. Type the following into the file:

```
INITIAL_CONTEXT_FACTORY=com.sun.jndi.fscontext.RefFSContextFactory
PROVIDER_URL=file:///JNDI-Directory
SECURITY_AUTHENTICATION=none
export classpath="/opt/mqm/java/lib/com.ibm.mq.jar:/opt/mqm/java/lib/com.ibm.mq.jms.Nojndi.jar:/opt/mqm/java/lib/com.ibm
```

   c. Press Escape then type `:wq`.

   d. In the terminal, enter `chmod u+x MyAdmin.config`. The file now has execute permissions.

7. Create the queue manager administered objects.

   a. Start the *MQSC* commands on sampleQM. Enter `runmqsc sampleQM`.

   b. Create a local queue, enter `DEFINE QLOCAL('myQueue')`.

   c. Create a server-connection channel, enter `DEFINE CHANNEL(JMS.CHANNEL) CHLTYPE(SVRCONN) TRPTYPE(TCP) MCAUSER('mqm')`.

   d. Create a client-connection channel, enter `DEFINE CHANNEL(JMS.CHANNEL) CHLTYPE(CLNTCONN) TRPTYPE(TCP) CONNAME(`*hostNameOfMQ*`) QMNAME('sampleQM')`.

   e. Create a listener object, enter `DEFINE LISTENER('listener1') TRPTYPE(TCP) CONTROL(QMGR)`.

   f. Enter end. The following message is displayed:

```
4 MQSC commands read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

8. Set the JMS environment.

    a. Log out of the mqm user and the login as root.

    b. Navigate to the /opt/mqm/java/bin directory. Enter `. ./setjmsenv` to set the file paths that are associated with Java. The following message is displayed:

```
MQ_JAVA_INSTALL_PATH is /opt/mqm/java
MQ_JAVA_DATA_PATH is /var/mqm
MQ_JAVA_LIB_PATH is /opt/mqm/java/lib
CLASSPATH is :/opt/mqm/java/lib/com.ibm.mq.jar:/opt/mqm/java/lib/com.ibm.mqjms.jar:/opt/mqm/samp/java/base:/opt/mqm/samp/ja
```

Note: If there is a Java error, repeat this step.

9. Start the JMSAdmin. Enter `./JMSAdmin -cfg MyAdmin.config`. The `InitCtx>` prompt is displayed.

10. Define a JMS context and change to it.

    a. Type `define ctx(JMS)`.

    b. Type `change ctx(JMS)`. The prompt changes to `InitCtx/JMS>`.

11. Create a connection factory that is called myCF, enter `define cf(myCF) tran(CLIENT) host(`*hostNameOfMQ*`) chan(JMS.CHANNEL) qmgr(sampleQM)`.

12. Create a queue called myQueue. Enter `define q(Q1) queue(mkyQueue) qmgr(sampleQM)`.

13. Verify the initial context. Enter `display ctx`. The output provides the following information:

```
Contents of InitCtx/JMS

    .bindings              java.io.File
  a  myCF                  com.ibm.mq.jms.MQConnectionFactory
  a  Q1                    com.ibm.mq.jms.MQQueue

3 Object(s)
  0 Context(s)
  3 Binding(s), 2 Administered
```

14. Enter `end` to stop WebSphere MQ classes for Java Message Service Administration.

15. Restart queue manager sampleQM

    a. Enter `endmqm sampleQM` to stop sampleQM.

    b. Enter `strmqm sampleQM` to start sampleQM.

Results

You have now created the WebSphere MQ objects that are required to use the sample JMS application.

What to do next

Follow the steps in "Testing the WebSphere MQ Version 6 configuration" to verify your configuration.

Related information:

➡ WebSphere MQ V6 documentation

Testing the WebSphere MQ Version 6 configuration

Run the sample JMS application to send and receive messages through WebSphere MQ, and verify that you configured WebSphere MQ correctly for use with the sample application.

Before you begin

Download the sample application package. Click the following link and save the file to the machine that hosts WebSphere MQ: sampleJMSApp_V6.zip, then extract the contents. The package contains a sample JMS application .jar file and shell script files for running the application.

- The sample sampleJMSApp.jar file and the .sh files must be in the same directory.
- The .sh files use environment variables to set the class path for running the JMS application. When the JMS application is run, if you see a Java java.lang.NoClassDefFoundError, you might be required to adjust the class path line in the command file.

About this task

The JMS application comprises a requester client, which sends the initial message, and a responder client, which receives the message and sends a reply.

- In one terminal run the runresponder.sh, responder client then waits for a message.
- In another terminal, run the runrequester.sh, the requester client starts and sends a request message and receives a reply.

With two command prompt windows, you can see the actions of the requester and responder separately and more clearly.

Procedure

1. Log in as the mqm user.
2. If the queue manager sampleQM is not started, enter strmqm sampleQM in a terminal.
3. Run the responder client.
    a. In a terminal window, navigate to the directory that contains the sampleJMSApp.jar and the runresponder.sh files.
    b. Add execute permissions to the shell script files. In a terminal window, navigate to the sample JMS application folder, and then enter chmod u+x *.sh.
    c. Enter ./runres.sh. The responder client waits for a message.

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Created connection.

> Waiting for message.
```

4. Run the requestor client.
    a. In another terminal window, navigate to the directory that contains the sampleJMSApp.jar and the runrequestor.sh files.
    b. Enter ./runreq.sh.

Results

The requestor client terminal, shows the connection status, the message it sent, and the reply message that it received from the responder client:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
```

```
> Connection created.

> Sending stock request for 'BakedBeans'
> Sent Message ID=ID:414d512073616d706c65514d20202020c615815102120020

> Received Message ID=ID:414d512073616d706c65514d20202020c615815102110020 for 'BakedBeans - 15 tins in stock'

> Closing connection to QueueManager.
> Closed Connection.
----------------------
In this window, observe the messages sent through WebSphere MQ.
- The request message sent
- The reply message received
---
When ready, press any key to close the window.
```

In the Responder window, observe the updated responder messages; the message it received (from the requester client) and the reply message that it sent:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Created connection.

> Waiting for message.

> Received Message ID=ID:414d512073616d706c65514d20202020c615815102120020 for 'BakedBeans'

> Sending Reply Message 'BakedBeans - 15 tins in stock'
> Sent Message ID=ID:414d512073616d706c65514d20202020c615815102110020

> Closing connection to QueueManager.
> Closed connection.
---------------------------------------------------------
In this window, observe the updated responder messages
- The request message received from the requester
- The reply message sent
-----
When ready, press any key to close this window
```

The messages that are shown in the two terminals verify that the requester and responder clients of the sample application can communicate with each other through WebSphere MQ.

What to do next

Follow the instructions in "Installing WebSphere Application Server Version 6.1" to begin the WebSphere Application Server part of the scenario.

Installing WebSphere Application Server Version 6.1

Install WebSphere Application Server, using the graphical user interface to select the installation options and create the stand-alone application server profile required for this scenario so that you are then ready to configure WebSphere Application Server to communicate with WebSphere MQ.

Before you begin

- You must have root access to the host.
- Ensure that the default shell for your Linux operating system is /bin/bash. Use the following command readlink /bin/sh to ensure that your default shell is bash not dash.
- Make sure that you have X Window System installed on your Linux system.

- Ensure that you have sufficient disk space, up to 2 GB for all components.
- Make sure that you have Firefox/Mozilla whose version number is between 4 to 9 (including). If you do not have, you can get one from Mozilla website.
- Verify that prerequisites and corequisites are at the required release levels, which can be reviewed on the System Requirements for WebSphere Application Server Deployment Manager V6.1 on Linux x86-64 .

About this task

This task describes the basic steps to install WebSphere Application Server 6.1 on the Red Hat 64-bit Linux operating system.

Procedure

1. Set the *BROWSER* variable to point to your browser. Enter

```
export BROWSER=/path/to/your/browser
```

> **Note:** The browser that you are pointing to must be Firefox/Mozilla Version 4 ~ 9.

2. Insert the disk labeled WebSphere Application Server Network Deployment into the disk drive, or access the root directory of the downloaded Passport Advantage image on your hard disk.

3. If you are using 64-bit Linux operating system, edit the `thisDisk.properties` file, which is in the *installation_image_root*/launchpad/diskinfo directory.

    a. Change the value for the *compatible[top.ARCHITECTURE]* property to x86_64.

4. Launch the launchpad. In a terminal window, navigate to the installation image root and enter `./launchpad.sh`.

5. On the Welcome page, click **Launch the installation wizard for WebSphere Application Server Network Deployment**. After a short delay, the installation wizard starts.

6. Follow the instructions in the wizard to move through the pages, choosing whether to install sample applications and language packs, and specifying an installation directory, until you reach the WebSphere Application Server environments page.

    Note: Although you can install the sample applications and language packs later, install them now if you might use them in the future. These features can be corrupted if you install them after applying maintenance.

7. On the WebSphere Application Server environments page, select the **Application server** option, then click **Next**.

    Why am I doing this?
    This option creates the simplest profile, for a stand-alone application server.
    ---

8. On the Enable Administrative Security page, choose whether to enable administrative security. If you accept the default option, to enable security, record the user name and password that you choose; without this information you will not be able to use the product.

> **Why am I doing this?**
>
> Administrative security activates the settings, such as the authentication of users, the use of Secure Sockets Layer (SSL), and the choice of user account repository, that protect your server from unauthorized users. Some environments, such as development systems, might not require security. However, in most environments enable administrative security to prevent unauthorized users from accessing the administrative console.

Click **Next**.

9. On the Installation Summary page, verify your choices, then click **Next** to install the product code. The installation wizard creates the installation program and then displays a progress page that shows which components are being installed. This process might take several minutes.

10. The Installation results page is displayed. Examine the information on the page to check that there were no file system or other unusual errors while installing. If there were problems during installation, use the `logs/install/log.txt` file in the installation directory to investigate.

11. Clear the **Launch the First steps console** check box, then click **Finish** to close the installation wizard. Close the launchpad.

Results

WebSphere Application Server is installed with the latest level of maintenance.

If you encountered a problem when installing WebSphere Application Server, see the topic Troubleshooting installation in the WebSphere Application Server information center, to resolve the issue.

What to do next

Follow the instructions in "Configuring WebSphere Application Server Version 6.1" to configure WebSphere Application Server to communicate with WebSphere MQ.

Configuring WebSphere Application Server Version 6.1

Create a listener port, connection factory, and queue for WebSphere Application Server Version 6.1 applications to use to communicate with WebSphere MQ Version 6.0.

About this task

To use JMS messaging between WebSphere Application Server and WebSphere MQ, you create the following objects in WebSphere Application Server.

• Listener port

> **Why am I doing this?**
>
> Needed to use a message-driven bean with WebSphere MQ as a JMS provider to WebSphere Application Server.

• Connection factory
• Queue (a type of JMS destination)

These objects are described in "Overview: Initial IT configuration" on page 6.

The following image shows these objects in WebSphere Application Server, and the corresponding set of objects in WebSphere MQ that you created when you configured WebSphere MQ for use by the sample application. When you create one of these objects in WebSphere Application Server, it is stored in the WebSphere Application Server JNDI namespace.
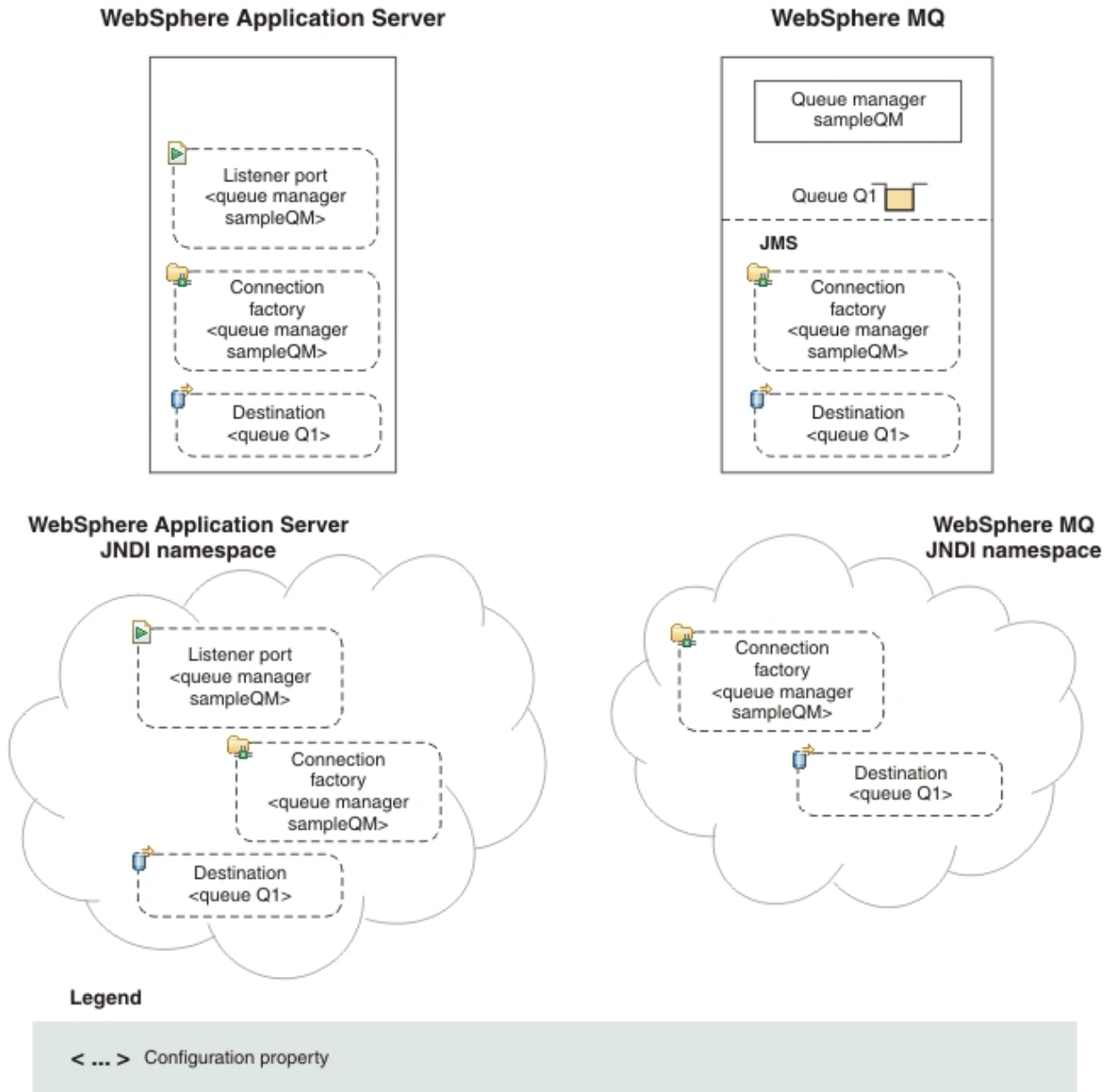

Figure 6. Objects created in WebSphere Application Server, and the corresponding objects in WebSphere MQ

Procedure

1. Log in as the root user.
2. If WebSphere Application Server is not already started, open a terminal, navigate to /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin, and then enter ./startServer.sh server1. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 77062
```

3. Start a web browser, in the address field enter *host_name*:9060/ibm/console. The *host_name* is the name of your WebSphere Application Server host. If the browser is on the same computer as WebSphere Application Server, the value of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when you installed the product. If you did not enable administrative security, the user ID is used only to distinguish changes that you make in the administrative console from changes that are made by other users.

4. Create a listener port.

   Why am I doing this?
   If you want to use message-driven beans with a messaging provider that does not have a JCA 1.5 resource adapter (for example the WebSphere MQ messaging provider), you cannot use activation specifications and therefore you must configure your beans against a listener port.
   ---

   a. In the administrative console, click **Servers** > **Application Servers** > *server_name* > **[Communications] Messaging** > **Message Listener Service** > **[Additional Properties] Listener Ports**.
   b. Click **New**.
   c. Specify the following required properties for the listener port:

      Name   A name of your choice; for example, PiShopReplyMDBListener

      Initial state
            Started

      Connection factory JNDI name
            The JNDI name for the JMS connection factory to be used by the listener port: jms/PiShopReplyCF.

            Tip: This name must match what you specify later when you configure the connection factory.

      Destination JNDI name
            The JNDI name for the destination to be used by the listener port:jms/wasQueue.

            Tip: This name must match what you specify later when you configure the queue.
   d. Click **OK**.

5. Create a connection factory.

   Why am I doing this?
   A JMS connection factory contains information about how to create a connection. When an application needs a JMS connection, the factory creates a connection instance. The connection factory is used for outbound messages from the MDB, whereas the listener port is used for inbound messages.
   ---

a. In the administrative console, click **Resources** > **JMS** > **Connection
   factories**.

b. Ensure that the scope is set to **Cell=*cell_name***, then click **New**.

c. Select **WebSphere MQ messaging provider**, then click **OK**. The
   Connection Factory Settings page is displayed.

d. On the Connection Factory Settings page, under General Properties,
   complete the required fields as follows:

┌─ **Why am I doing this?** ─────────

The name you choose for the connection factory is used to display
the connection factory in the administrative console. The JNDI name
is used to reference the object in the JNDI namespace. The JNDI
name is also used in the sample MDB application code; if you want
to use a different JNDI name you must change the MDB to match,
then redeploy the application.

Name Specify a name of your choice; for example: `PiShopReplyCF`

JNDI name
        Specify `jms/PiShopReplyCF`

        Tip: This name must match what you specified earlier when you
        configured the listener port.

Queue manager
        Specify `sampleQM`

Host The host name of the first WebSphere MQ host. If you are using
          WebSphere Application Server and WebSphere MQ on the same
          computer, enter `localhost`.

Port Default value is **1414**, if you specified a port number during the
          WebSphere MQ configuration, enter this value here.

Channel
        Enter `JMS.CHANNEL`.

┌─ **Why am I doing this?** ─────────

Defines the name of the channel a client can use to connect to
the queue manager, sampleQM.

Transport type
        Change to **CLIENT**.

> **Why am I doing this?**
>
> In previous step of configuring WebSphere MQ, you use client as the transport type of your connection factory. In order to connect your WebSphere Application Server to WebSphere MQ, you need to set the transport type as client as well.

    e. Leave the default settings for all other fields on the Connection Factory Settings page, then click **OK**

6. Create a queue.

> **Why am I doing this?**
>
> A queue is a type of JMS destination, used for point-to-point messaging. A destination identifies the messaging target or source.

> **What else might I do or be interested in?**
>
> In a publish/subscribe scenario, you create a topic instead of a queue.

    a. In the administrative console, click **Resources** > **JMS** > **Queues**.

    b. Ensure that the scope is set to **Cell=*cell_name***, then click **New**.

    c. Select **WebSphere MQ messaging provider** then click **OK**.

    d. Under General Properties, enter the following values for the required fields.

      Name   A name of your choice; for example: `wasQueue`

           This name is used to display the queue in the administrative console.

      JNDI name
           `jms/wasQueue`

           The JNDI name is used to reference this queue in the WebSphere Application Server JNDI namespace.

           Tip: This name must match what you specified earlier when you configured the listener port.

      Base queue name
           `myQueue`

           This value is the administrative name of the queue in WebSphere MQ that you want to communicate with. This is different from the JNDI name of the queue.

> **What else might I do or be interested in?**
> Specify a **Queue manager or Queue-sharing group name** if you want
> WebSphere Application Server to send messages to a queue that is on
> a different queue manager from the one connected to by the listener
> port or connection factory. The queue managers must be configured
> to communicate with each other, as described in the topic ic10240_
> "What is intercommunication?" in the WebSphere MQ V6
> downloadable Information Center.

   e. Click **OK**

7. Configure the WebSphere MQ messaging provider with native libraries
   information.

> **Why am I doing this?**
> When you created the connection factory you specified a transport mode
> of **Bindings**. To use bindings mode, the WebSphere MQ messaging
> provider within WebSphere Application Server requires native libraries,
> for example `.so` files, from WebSphere MQ. You must configure the
> WebSphere Application Server environment variable
> *MQ_INSTALL_ROOT* to the directory where WebSphere MQ is installed.

   a. In the administrative console, click **Environment** > **WebSphere variables**.

   b. Make sure that the **Scope** property is set to Node. The Scope value should
   contain only `Node=`*nodename*

   c. In the table of variables, make sure that the value for
   MQ_INSTALL_ROOT is the full path to the directory where WebSphere
   MQ is installed. For a default installation of WebSphere MQ, the path is
   `/opt/mqm`.

     If you need to change the value, click the **MQ_INSTALL_ROOT** link, type
   the correct value into the Value field, then click **OK** to return to the table
   of variables.

8. Click the **Save** link in the Messages section at the top of the pane to save your
   changes to the master configuration.

9. Stop the application server. In a terminal, navigate to `/opt/IBM/WebSphere/`
   `AppServer/profiles/AppSrv01/bin`, and enter `./stopServer.sh server1`.

10. Start the application server. In the same terminal, enter `./startServer.sh`
    `server1`

Results

You have configured WebSphere Application Server with the necessary objects to
enable communication with WebSphere MQ.

What to do next

Follow the instructions in "Deploying the sample MDB application to WebSphere
Application Server Version 6.1" on page 34 to deploy the sample MDB application.

Deploying the sample MDB application to WebSphere Application Server Version 6.1

Using the WebSphere Application Server administrative console, install the Version 6 sample message driven bean (MDB) application so that you can use it to verify that WebSphere Application Server is communicating with WebSphere MQ. When you install the MDB application, you associate it with the listener port name and connection factory JNDI name that you specified when you configured WebSphere Application Server.

Before you begin

Before starting this task, download the Version 6 sample MDB application. Click the following link and save the file to the machine that hosts WebSphere Application Server: sampleMDBv6.ear.

About this task

The sample MDB application, `sampleMDBv6.ear`, is designed to use the objects that you created earlier in WebSphere Application Server. The MDB application uses these objects to send a message to WebSphere MQ, for receipt by the sample JMS application requester client that you used in "Testing the WebSphere MQ Version 6 configuration" on page 24.

The following diagram shows a message traveling from the sample JMS client to WebSphere MQ, and then on to WebSphere Application Server, where it is passed to the MDB running within WebSphere Application Server. A response message travels from the MDB to WebSphere MQ, and then on to the JMS client.
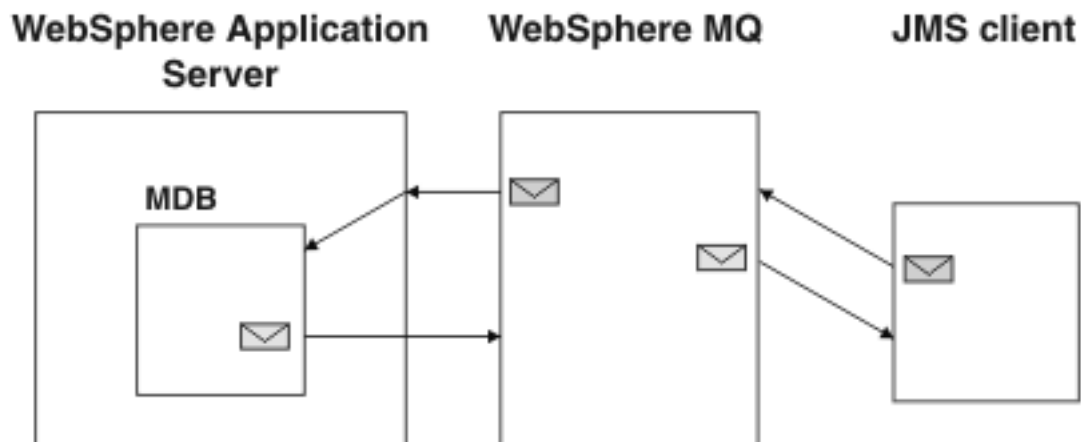
Figure 7. A message traveling from the sample JMS client to WebSphere MQ, then onward to the MDB on WebSphere Application Server

Procedure

1. Log in as root user.
2. If WebSphere Application Server is not already started, open a terminal, navigate to /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin, and then enter `./startServer.sh server1`.
3. Start a web browser, in the address field enter *host_name*:9060/ibm/console.
4. Click **Applications** > **Install New Application**.

5. On the Preparing for the application installation page, select **Local file system**, then browse to the folder where you downloaded the sample MDB application, `sampleMDBv6.ear`.

   a. Select the application then click **Open**.

   b. Leave all other fields unchanged and click **Next**.

   What else might I do or be interested in?
   Unless you specify otherwise, the name of the installed application is derived from the name of the MDB EAR file, which in the case of this scenario is `sampleMDBv6.ear`. If you prefer, you can give the application a different name. To do this, click the **Step 1** link to go to the Select installation options page then specify your chosen **Application name**.
   ---

6. On the Install New Application page, click the **Step 3** link to go to the Bind listeners for message-driven beans page.

   What else might I do or be interested in?
   To save having to specify the binding whenever you deploy the MDB EAR file, you can bind the MDB to a listener port name using an assembly tool. For more information, see the topic Preparing for application installation binding settings in the WebSphere Application Server information center.
   ---

7. On the Bind listeners for message-driven beans page specify the listener port name to which the sample MDB application must be bound.

   Why am I doing this?
   An MDB must be associated with either a listener port name or activation specification JNDI name. Since this stage of this scenario is using a listener port, the MDB must be bound to the listener port name
   ---

   a. Under **Listener Bindings**, select **Listener port**.

      Tip: You do not need to select the EJB module, was61MDB, because it is the only EJB module in the EAR file.

   b. Type the **Name** of the listener port that you created as described in "Configuring WebSphere Application Server Version 6.1" on page 28: `PiShopReplyMDBListener`.

   c. Click **Next**.

8. On the Map resource references to resources page, map the connection factory resource reference defined in the sample MDB application to the JNDI name of the connection factory.

> **Why am I doing this?**
>
> Each resource reference that is defined in an application must be mapped to an actual resource that has been configured for use by the application. A resource reference allows an application to access a resource by using a logical name rather than the actual name that is defined for the resource in the runtime environment. Resource references eliminate the need to change your application code when you change resource runtime configurations.

  a. Under **Target Resource JNDI Name**, click **Browse** to display a list of resources that are available to be used to bind the resource reference of the MDB.

  Tip: For this scenario, the resource that you need to select is the connection factory PiShopReplyCF that you created as described in "Configuring WebSphere Application Server Version 6.1" on page 28.

  b. Select `PiShopReplyCF` then click **Apply**.

  c. Leave all other fields unchanged and click **Next**.

 9. On the Summary page, click **Finish**. The program displays messages to indicate that the application is being installed.

10. When the application installs successfully, click the **Save** link that is displayed at the end of the messages, to save your changes to the master configuration. If the application does not install successfully, examine the messages to identify why the installation failed. Correct the problems as needed and try installing the application again.

11. Stop the application server. In a terminal, navigate to `/opt/IBM/WebSphere/ AppServer/profiles/AppSrv01/bin`, and enter `./stopServer.sh server1`.

12. Start the application server. In the same terminal, enter `./startServer.sh server1`

Results

You have deployed the MDB application, which is now visible in the Enterprise Applications page. To view this page, click **Applications** > **Enterprise Applications**.

What to do next

Follow the instructions in "Verifying the sample IT configuration" to run the sample applications and verify that messages can be passed between the two products.

Verifying the sample IT configuration

Use the Version 6 sample JMS and message driven bean (MDB) applications to send and receive messages so that you can verify that WebSphere MQ and WebSphere Application Server can communicate with each other.

Before you begin

You must have configured WebSphere Application Server and WebSphere MQ as described in "Configuring WebSphere Application Server Version 6.1" on page 28 and "Configuring the JNDI namespace and administered objects" on page 21.

About this task

In "Testing the WebSphere MQ Version 6 configuration" on page 24, you ran the supplied JMS sample application to verify that the requester and responder clients of the application could communicate through WebSphere MQ. In "Deploying the sample MDB application to WebSphere Application Server Version 6.1" on page 34, you installed an MDB application. In this task, you run the requester application as before, but this time the reply comes from the MDB application instead of the previous responder application, verifying that messages are being passed between WebSphere MQ and WebSphere Application Server.
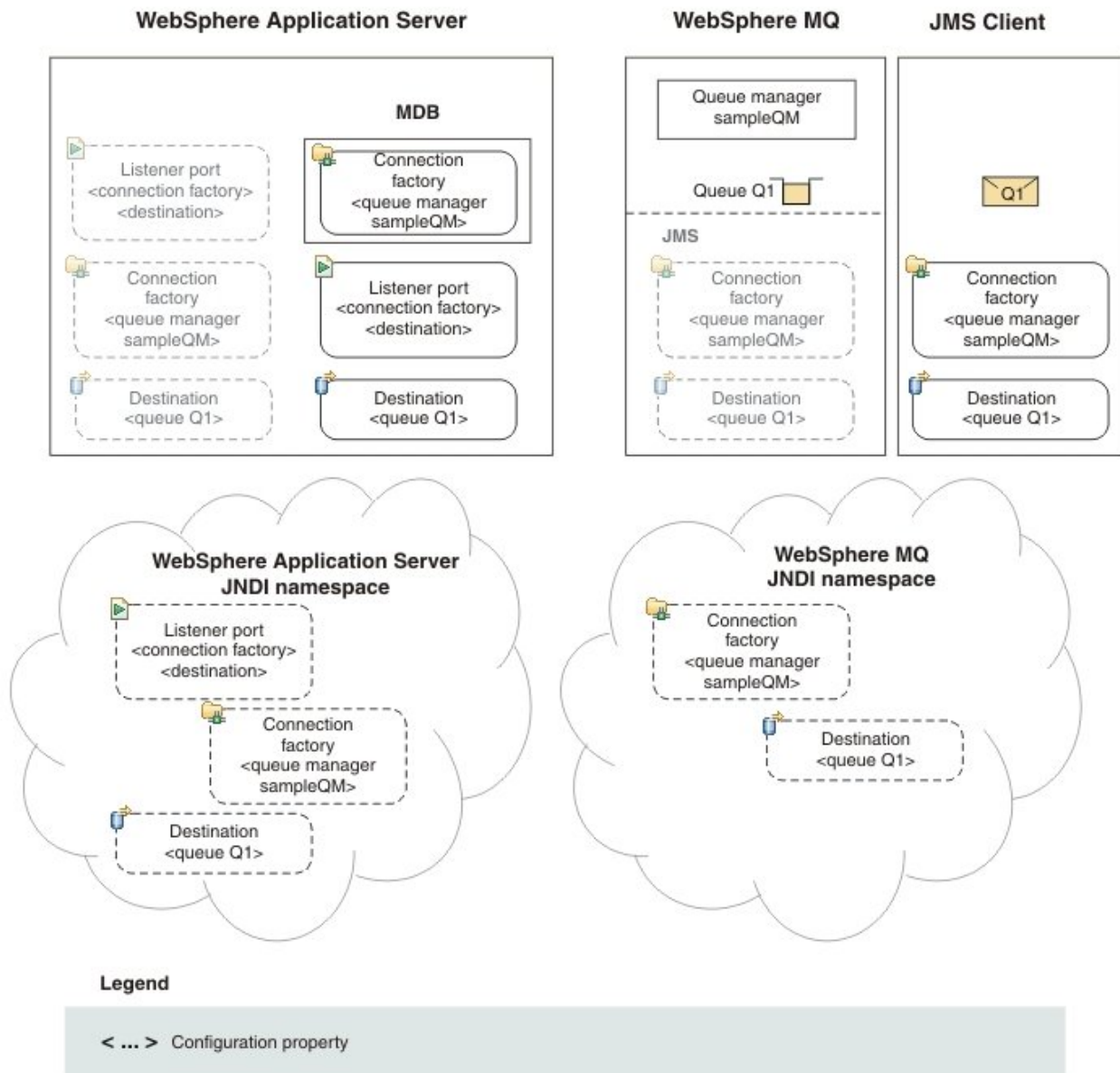
Procedure

1. If sampleQM is not started, log in as the mqm user and then enter `strmqm sampleQM`.

2. Log in as the root user.

3. If WebSphere Application Server is not already started, open a terminal, navigate to /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin, and then enter ./startServer.sh server1.

4. Start a web browser, in the address field enter *host_name*:9060/ibm/console.

5. Check the status of the listener port PiShopReplyMDBListener then, if it is not already started, start it.

   a. In the administrative console, click **Servers** > **Application Servers** > *server_name* > **[Communications] Messaging** > **Message Listener Service** > **Listener Ports**.

   b. Check whether the status of the listener port is started. If listener port is running, the status is shown as a green arrow. If the listener port is not running, the status is shown as a red cross.

   c. If the listener port is not already started, select it then click **Start**. The status of the listener port changes to a green arrow.

6. Check whether the Version 6 sample MDB application is already running then, if it is not running, start it.

   a. Click **Applications** > **Enterprise Applications** to display a list of installed applications.

   b. Check whether the application status of sampleMDBv6 is started. If sampleMDBv6 is running, the application status is shown as a green arrow. If sampleMDBv6 is not running, the application status is shown as a red cross.

   c. If sampleMDBv6 is not already running, select it then click **Start**. The application status changes to a green arrow. The MDB application is now waiting to receive a message.

   If you see an error message similar to the following message, there might be a mismatch between your WebSphere Application Server configuration and the information contained in the sample application. Check the WebSphere Application Server error logs in /opt/IBM/WebSphere/AppServer/AppSrv01/logs/server1 for more information.

```
Error sampleMDB failed to start. Check the logs for server server1 on node
    mymachineNode01 for details.
Error An error occurred while starting sampleMDBv6. Check the logs for
    server server1 on node mymachineNode01 for more information.
```

7. Log off as root and log in as mqm.

8. Run the requestor client.

   a. On the command line, navigate to the directory that contains the sampleJMSApp.jar and the runreq.sh files.

   b. Enter ./runreq.sh.

Results

The sample requester application sends the same request message as before, but the response now comes from the MDB application. The following output, showing sent and received messages, is displayed in the requester application Command Prompt window:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Connection created.

> Sending stock request for 'BakedBeans'
> Sent Message ID=ID:414d512073616d706c65514d20202020362c8251020d0020
```

```
> Received Message ID=ID:414d512073616d706c65514d20202020362c8251020f0020 for 'From V6 MDB: BakedBeans - 6 tins in stock

> Closing connection to QueueManager.
> Closed Connection.
-----------------------
In this window, observe the messages sent through WebSphere MQ.
- The request message sent
- The reply message received
---
When ready, press any key to close the window.
```

> **Note:** If you update the WebSphere Application Server configuration, you must restart the application server for the changes to take effect.

Option 1: Migrating WebSphere Application Server before WebSphere MQ

In Option 1, you first migrate your WebSphere Application Server Version 6 configuration to Version 8.5 and use the Version 6 sample applications to check that your migrated configuration can still communicate with WebSphere MQ, which is still at Version 6. You then migrate WebSphere MQ to Version 7.5 and use the sample Version 7 applications to check that messages can still be passed between WebSphere MQ and WebSphere Application Server.

Before you begin

The starting point for this scenario is the sample Version 6 IT configuration described in "Overview: Initial IT configuration" on page 6.

- If you have not already created the sample Version 6 IT configuration, complete the tasks described in "Creating a sample initial Version 6 IT configuration" on page 18.
- If you have already completed "Option 2: Migrating WebSphere MQ before WebSphere Application Server" on page 68, and you now want to try "Option 1: Migrating WebSphere Application Server before WebSphere MQ," you first must uninstall both products used in the other option (to reset the installation), and then set up the sample Version 6 IT configuration again as described in the following topics:
  1. "Uninstalling WebSphere MQ" on page 64
  2. "Uninstalling WebSphere Application Server Version 8.5" on page 65
  3. "Uninstalling WebSphere Application Server Version 6.1" on page 67
  4. "Creating a sample initial Version 6 IT configuration" on page 18

Phase 1: Migrating WebSphere Application Server to Version 8.5

Make a backup copy of your WebSphere Application Server Version 6.1 configuration, and then install Version 8.5. After completing the installation, migrate the WebSphere Application Server resources to Version 8.5, and then check that the migration of these resources and of the sample MDB application has been successful.

Before you begin

The starting point for this scenario is a clean sample Version 6 IT configuration. If you have not already done so, read the section "Before you begin" in the topic "Option 1: Migrating WebSphere Application Server before WebSphere MQ," and

complete any appropriate actions described there.

About this task

Phase 1 of this scenario covers the migration of WebSphere Application Server from Version 6 to Version 8.5. At the end of this phase, your WebSphere MQ configuration is still at Version 6 but messages can be passed between WebSphere MQ Version 6.0 and WebSphere Application Server Version 8.5 using the sample applications.

The migration process for WebSphere Application Server involves installing a Version 8.5 environment alongside the existing Version 6 environment, then migrating your Version 6 configuration to the Version 8.5 environment.

Procedure

1. Follow the instructions in "Preparing to migrate to WebSphere Application Server Version 8.5" on page 41 to prepare for migrating to WebSphere Application Server Version 8.5.
2. Follow the instructions in "Installing WebSphere Application Server Version 8.5" on page 43 to install WebSphere Application Server Version 8.5.
3. Follow the instructions in "Migrating your WebSphere Application Server configuration to Version 8.5" on page 46 to migrate your WebSphere Application Server Version 6.1 configuration to Version 8.5, using the main migration wizard.

> **What else might I do or be interested in?**
> This scenario does not require the use of any migration tools other than the migration wizard. However, in more complex scenarios, you might need to use additional migration tools to complete any upgrades that are not done automatically by the migration wizard; for example, upgrade of JSP objects or Derby databases.

4. Follow the steps in "Confirming migration of WebSphere Application Server resources to Version 8.5" on page 49 to check that the listener port, connection factory, and queue from your WebSphere Application Server Version 6 installation have successfully migrated to WebSphere Application Server Version 8.5.
5. Follow the steps in "Verifying the migrated sample MDB application" on page 52 to test that the migrated MDB application in WebSphere Application Server Version 8.5 can communicate with WebSphere MQ Version 6.

Results

You have migrated WebSphere Application Server from Version 6 to Version 8.5. At the end of this phase, your WebSphere MQ configuration is still at Version 6, but you have confirmed that the migrated MDB application in WebSphere Application Server Version 8.5 can communicate with WebSphere MQ Version 6.

What to do next

When you have confirmed that your migrated WebSphere Application Server is working as required in Version 8.5, you can, if required, remove WebSphere Application Server Version 6.1 using the uninstaller program that is created during installation. The uninstaller program removes registry entries, uninstalls the

product, and removes all related features. You might also need to do some manual cleaning up, as the uninstaller program does not remove log files in the installation root directory. For more information, see "Uninstalling WebSphere Application Server Version 6.1" on page 67.

Note: If you migrate any profiles to a new version of the product, you must either shut down the new migrated deployment managers before uninstalling the product, or include the `removeProfilesOnUninstall="false"` parameter on the uninstall command. Either of these options prevents the migrated profiles from being deleted when you uninstall the old version of the product.

After completing the phase 1 tasks, move on to "Phase 2: Migrating WebSphere MQ to Version 7.5" on page 54.

Preparing to migrate to WebSphere Application Server Version 8.5

Prepare to migrate your WebSphere Application Server Version 6.1 configuration to Version 8.5 by reviewing premigration considerations and making a backup copy of your Version 6.1 configuration. At the end of this task, you will be ready to install WebSphere Application Server Version 8.5 with the latest level of maintenance.

Before you begin

- You must have local root access when you are installing.
- Ensure that you have sufficient disk space, up to 2 GB for all components.
- Ensure that the default shell for your Linux operating system is `/bin/bash`. Use the following command `readlink /bin/sh` to ensure that your default shell is `bash` not `dash`.
- Check for any post release technical notes available for your operating system at the product support site Support home - WebSphere Application Server.
- Make sure that your Linux system has the following items:
  - Kernel and C runtime library
  - Current® and all compatibility versions of the C++ runtime library
  - X Window System libraries
  - GTK runtime libraries
- Ensure that there is adequate disk space. Installing all features and products requires approximately 2 GB of disk space. This estimate includes the following products, components, and features:
  - Main application server product installation
  - Profiles
  - Sample applications
  - IBM HTTP Server
  - Web Server plug-ins
  - Application Client for WebSphere Application Server
- The backup configuration utility creates an archive file into a folder of your choice. If the folder does not already exist, create it now before continuing.

For more details about requirements for installation, see the IBM Support pages.

About this task

This task describes the steps for preparing to install WebSphere Application Server on the Red Hat 64-bit Linux operating system.

Procedure

1. Review the Premigration considerations topic in the WebSphere Application Server Version 8.5 information center.

   Tip: You do not need to complete this step for this scenario because the information and actions needed are provided within the scenario to make it easier for you to proceed. However, it is good general practice to review the latest premigration considerations, especially if preparing to migrate your own WebSphere Application Server Version 6 environment.

2. Log in as the root user.

3. Make a backup copy of your existing Version 6.1 configuration and make a note of the exact name and location of the backed-up configuration.

   ┌─ **Why am I doing this?** ───┐
   │ The migration tools create a backup copy of the configuration from the │
   │ previous version, as a file in zip format. Backing up your existing │
   │ configuration before you start the migration allows you to restore the │
   │ configuration to its previous state after migration. │
   └──┘

   ┌─ **What else might I do or be interested in?** ────────────────────┐
   │ You can use either the **backupConfig** utility or your own preferred backup │
   │ utility to create the backup. The backupConfig command is a simple │
   │ utility to back up the configuration of your node to a file. By default, all │
   │ servers on the node stop before the backup is made so that partially │
   │ synchronized information is not saved. If you do not have root authority, │
   │ you must specify a path for the backup file in a location where you have │
   │ write permission. The backup file will be in zip format and a .zip │
   │ extension is recommended. │
   └──┘

   Prerequisite: The directory into which you want to create the backup file must exist before running the **backupConfig** command.

   a. Open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin`.

   b. Enter the following command to run the backupConfig utility, specifying the target directory and file name.

```
./backupConfig.sh migbackupdir/backupfile.zip
```

   ┌─ **Why am I doing this?** ───┐
   │ If you want to save the backup archive file in a specific location of │
   │ your own choice, you must include the complete pathname when you │
   │ run `backupConfig.sh`. If you do not specify a pathname the backup │
   │ archive file is placed in the same directory as the one from which you │
   │ ran `backupConfig.sh`. │
   └──┘

> **What else might I do or be interested in?**
> To avoid user ID and password prompts when you use this command, add the user ID and password information to the soap.client.props file located in `profile_root/properties/soap.client.props`.

For example:

```
./backupConfig.bat /opt/migbackupdir/myBackup.zip
```

After a short pause, the backup process starts, and information messages, similar to those in the following example, are shown in the Command Prompt window.

```
ADMU0116I: Tool information is being logged in file
           /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/backupConfig.log
ADMU0128I: Starting tool with the AppSrv01 profile
ADMU5001I: Backing up config directory
           /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config to file
           /opt/migbackupdir/myBackup.zip
ADMU0505I: Servers found in configuration:
ADMU0506I: Server name: server1
ADMU2010I: Stopping all server processes for node mylocalNode01
ADMU0510I: Server server1 is now STOPPED
```

When the backup is complete, a message confirming this is displayed in the Command Prompt window:

```
ADMU5002I: 358 files successfully backed up
```

4. Ensure that prerequisites and corequisites are at the required release levels. On the WebSphere Application Server detailed system requirements web page, click the link for your platform to see information about supported operating systems, browsers, and hardware requirements.

   Tip: You do not need to complete this step for this scenario because the information and actions needed are provided within the scenario to make it easier for you to proceed. However, it is good general practice to verify prerequisites and corequisites, especially if preparing to migrate your own WebSphere Application Server V6 environment.

Results

You are now ready to install WebSphere Application Server Version 8.5.

What to do next

Follow the instructions in "Installing WebSphere Application Server Version 8.5" to install WebSphere Application Server Version 8.5.

Installing WebSphere Application Server Version 8.5

Install WebSphere Application Server 8.5 by using Installation Manager. At the end of this task, you are ready to configure WebSphere Application Server to communicate with WebSphere MQ.

Before you begin

- You must have root access when you are installing.
- Ensure that you have sufficient disk space. You require up to 2 GB for all components.
- Ensure that prerequisites and corequisites are at the required release levels. On the WebSphere Application Server detailed system requirements page, click the

link for your operating system and the link for software to see information about supported operating systems and browsers, and hardware requirements.

About this task

This task describes the basic steps to install WebSphere Application Server 8.5 on Red Hat 64-bit Linux operating system by using Installation Manager.

Procedure

1. Log in as root user.
2. If you have not installed IBM Installation Manager before, install it by completing the following steps:
   a. Extract the Installation Manager archive file.
   b. In a terminal window, navigate to the Installation Manager archive directory and enter ./installc -c. The following message is displayed:

```
Preprocessing the input.
Loading repositories...
Preparing and resolving the selected packages...


=====> IBM Installation Manager> Install

Select packages to install:
    1. [X] IBM® Installation Manager 1.5.2

    O. Check for Other Versions, Fixes, and Extensions

    N. Next,      C. Cancel
-----> [N]
```

   c. Enter N then A to accept the license agreement.
   d. Enter N until the installation summary like following is displayed:

```
=====> IBM Installation Manager> Install> Licenses> Location> Summary

Target Location:
  Package Group Name      :  IBM Installation Manager
  Installation Directory   :  /opt/IBM/InstallationManager/eclipse

Packages to be installed:
       IBM® Installation Manager 1.5.2

Options:
    G. Generate an Installation Response File

    B. Back,      I. Install,     C. Cancel
-----> [I]
```

   Press Enter. The installation process begins. When the installation is complete, the following message is displayed:

```
=====> IBM Installation Manager> Install> Licenses> Location> Summary>
  Completion

The install completed successfully.

Options:
    R. Restart Installation Manager
-----> [R]
```

   e. Press Enter, and then enter X to exit the Installation Manager.

3. In a terminal window, navigate to `/opt/IBM/InstallationManager/eclipse`, and then enter `./launcher`. The IBM Installation Manager window is displayed.

4. Click **Install**.

5. If a message is displayed that states `Installation Manager cannot find any packages to install`, add WebSphere Application Server to the repository.

   a. Click the **Repositories** link in the **Repository connections** field.

   b. Click **Add Repository** in the upper right of the Preference window.

   c. Click **Browse** and find the `repository.config` file for WebSphere Application Server.

   d. Click **OK**.

   e. Click **Test Connections** on the right of the Preference window. If a message is displayed that states `All the selected repositories are connected`, WebSphere Application Server is added to the repository.

   f. Click **OK** to close the window and click **OK** again to confirm adding the repository.

6. Select **Version 8.5.0.0**, and then click **Next**.

7. Click **I accept the terms in the license agreement**, and then click **Next.**

8. Click **Next**, then click **Next** again.

9. Select the languages that you want and click **Next**.

10. Select the components that you want and click **Next**.

11. Click **Install**. A progress bar is displayed, which shows that the installation is in progress.

12. When the installation is complete, a window is displayed, which indicates that the packages are installed. Click **Profile Management Tool to create a profile**, and then click **Finish**. WebSphere Application Server is successfully installed and "WebSphere Customization Toolbox" is displayed.

    If "WebSphere Customization Toolbox" window is not displayed, open a terminal and navigate to `/opt/IBM/WebSphere/AppServer1/bin/ProfileManagement`, and then enter `./pmt.sh` to start it.

13. Create an application server profile

    a. In the "WebSphere Customization Toolbox" window, click **Profile Management Tool** among the three buttons on the top.

    b. In the Profile Management Tool, click **Create**.

    c. In Environment Selection window, expand **WebSphere Application Server**, and click **Application server**, and then click **Next**.

    d. In Profile Creation Options window, leave the option as **Typical profile creation**, and then click **Next**.

    e. Choose whether to enable administrative security and click **Next**. If you choose to enable it, every time you use the Administrative Console, you must prove your identity.

    f. Confirm your settings and click **Create**.

    g. When the profile creation is complete, the Profile Creation Complete window is displayed. Clear **Launch the First steps console** and click **Finish**. An application server profile is successfully created.

Results

WebSphere Application Server is installed and a profile is created. If you encountered a problem when installing, see the IBM Installation Manager product

documentation. For further information about where to find this information center, see IBM Installation Manager product documentation.

What to do next

Follow the instructions in "Migrating your WebSphere Application Server configuration to Version 8.5" to migrate your Version 6.1 configuration to Version 8.5.

Related information:

➡ Installing the Update Installer for WebSphere Software

➡ Installing WebSphere Application Server

Migrating your WebSphere Application Server configuration to Version 8.5

Use the migration wizard to migrate the listener port, connection factory, and queue from WebSphere Application Server Version 6.1 to Version 8.5 so that you can then view and configure these resources in the Version 8.5 administrative console.

Before you begin

Before starting this task, follow the instructions in "Installing WebSphere Application Server Version 8.5" on page 43 to install WebSphere Application Server Version 8.5.

About this task

This task covers migration of the stand-alone application server profile that you created in "Installing WebSphere Application Server Version 6.1" on page 26.

┌─ **What else might I do or be interested in?** ─────────────────┐
The wizard is the graphical interface to the primary Version 8.5 command-line migration tools, which are the **WASPreUpgrade** and **WASPostUpgrade** commands. You can also use those commands directly to complete a migration.
└──┘

The migration wizard includes an option to back up your Version 6.1 configuration before migrating to Version 8.5. This enables you to restore the configuration to its previous state after migration.

┌─ **What else might I do or be interested in?** ─────────────────┐
Before migrating a WebSphere Application Server Version 5.1.x or Version 6.x application server, you can use the backupConfig command or your own preferred backup utility to back up your existing configuration.
└──┘

Procedure

1. Log in as root user.

2. If you have not already done so, stop the WebSphere Application Server Version 6.1 application server. In a terminal window, navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and enter `./stopServer.sh server1`.

3. Start the migration wizard. Navigate to `/opt/IBM/WebSphere/AppServer1/bin/ProfileManagement`, and then enter `./wct.sh`.

4. Click the **Migration Sources** tab, and then click **New**.

5. On the "Detected versions of WebSphere Application Server" page, from the list of detected versions of WebSphere Application Server that exist on your computer, select your Version 6.1 installation. For example, `IBM WebSphere Application Server Network Deployment Version 6.1.0.0`.

   Click **Next**.

6. On the "Source profile selection" page, select the source profile that you want to migrate, for example `AppSrv01`, then click **Next**.

7. On the "Profile Migration Output" page, specify the location of the migration backup directory in which you want to place a backup copy of the configuration from the previous version, then click **Next**.

> **What else might I do or be interested in?**
> If the directory does not already exist, it is created. If the directory already exists, it must be empty because otherwise the backup installation might overwrite existing backup files that are still needed.

8. On the "Target profile selection" page, complete the following steps:

   a. Click **Migrate to an existing profile**, and then select the target profile to which you want to migrate the selected Version 6 source profile, for example `AppSrv01`.

   > **What else might I do or be interested in?**
   > This page also offers you the option to create a new profile. If you did not create a profile when you installed WebSphere Application Server Version 8.5 you must select **Create new profile** here, then enter the parameters for creating the new profile.

   b. Select **Create a backup of the target profile**.

   > **Why am I doing this?**
   > If necessary you can then use the `restoreConfig` command to restore the configuration after migration.

   c. Click **Next**.

9. On the "Application Migration" page, leave the default option **Migrate and install the applications as part of the migration process** selected.

> **Why am I doing this?**
> In this scenario, the sample applications used on Version 6 are to be used on Version 8.5 without any change, so are included as part of the migration.

10. In the **Application installation directory** field, leave the default option **Install the applications in the default directory of the target profile** selected, and then click **Next**.

> **What else might I do or be interested in?**
> The other options are to keep the same application installation directories as the previous version or to install the applications into the specified application installation directory. If you select the option to install your applications in the specified application installation directory, you also need to specify where the migrated applications should be located.

11. On the "Port value" page, select the option **Use the port values assigned to the target profile**, then click **Next**.

> **Why am I doing this?**
> Because WebSphere Application Server Version 8.5 was installed on the same computer as the existing WebSphere Application Server Version 6, the port numbers used by the Version 8.5 profile were incremented to avoid port conflicts with the Version 6 profile. Selecting the option **Use the port values assigned to the target profile** causes the migration wizard to use the port numbers already assigned to the Version 8.5 profile.

> **What else might I do or be interested in?**
> You could select the option **Use the port values assigned to the previous (source) profile** to cause the migration wizard to try to use the port numbers from the Version 6 profile. By default, any port conflict is resolved by incrementing the port number of the target profile by one until an unused port number is found. You can specify a starting port number to be used when a conflict is detected. If the starting port number is in use, it is incremented by one until an unused port number is found.

12. On the "Additional Migration Options" page, complete the following steps:

   a. In the **Scripting compatibility** field, clear **Migrate object types to support script compatibility**.

   > **Why am I doing this?**
   > This is a temporary transition option that minimizes the impact of the migration on existing administrative scripts. This is meant to provide a temporary transition until all of the nodes in the environment are at the Version 8.5 level.

   b. In the **Administrative console "My Tasks"** field, click **Use the default workspace user root directory**.

> **Why am I doing this?**
>
> In this scenario, you did not create a special task view for the `My Tasks` view in the administrative console navigation. By default, "My Tasks" user information in the Version 6 installation would be stored in the default workspace user root location.

> **What else might I do or be interested in?**
>
> If you had changed the location of the workspace root location, you would use the option **Specify the workspace user root location**. For more information about changing the location, see Changing the location of the wstemp temporary workspace directory in the WebSphere Application Server information center.

13. Check the information on the "Migration summary" page, then click **Migrate** to start the migration.

14. When the wizard indicates that each stage of the process has been successful, click **Next** to move to the next stage.

    Tip: In the output from the post-upgrade processing, you should see a message that the sample application has been migrated to the Version 8.5 profile: `MIGR0499I: Application sampleMDBv6.ear successfully deployed using the wsadmin command.`

    If any problems occur, correct them and then start the migration process again.

15. On the "Migration Output" page, click **Next**.

16. On the "Migration Results" page, click **Finish**.

Results

You have migrated the listener port, connection factory, and queue from WebSphere Application Server Version 6.1 to Version 8.5, and are ready to check that you can view these resources in the Version 8.5 administrative console.

What to do next

To confirm that the migration of the Version 6.1 resources has been successful, follow the instructions in "Confirming migration of WebSphere Application Server resources to Version 8.5."

Confirming migration of WebSphere Application Server resources to Version 8.5

Using the administrative console, check that the listener port, connection factory, and queue have been successfully migrated from WebSphere Application Server Version 6.1 to Version 8.5.

About this task

To use JMS messaging between WebSphere Application Server Version 8.5 and WebSphere MQ Version 6, you must check that following WebSphere Application Server objects have been successfully migrated from Version 6.1.

- Listener port

- Connection factory
- Queue (a type of JMS destination)

These objects are described in "Overview: Initial IT configuration" on page 6.

Procedure

1. Log in as root user.
2. If WebSphere Application Server is not already started, open a terminal
   window and navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/`
   `bin`, and then enter `./startServer.sh server1`. The following message is
   displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

3. Start a web browser, in the address field enter *host_name*`:9061/ibm/console`.
   The *host_name* is the name of your WebSphere Application Server host. If the
   browser is on the same computer as WebSphere Application Server, the value
   of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when
   you installed the product. If you did not enable administrative security, the
   user ID is used only to distinguish changes that you make in the administrative
   console from changes that are made by other users.

4. Check the listener port settings.
   a. In the administrative console, click **Servers** > **[Server Types] WebSphere
      application servers** > *server_name* > **[Communications] Messaging** >
      Message Listener Service > **[Additional Properties] Listener Ports**.
   b. Click **PiShopReplyMDBListener**. The Listener Port settings page is
      displayed.
   c. Check that the following values appear in the required fields:

      Name   `PiShopReplyMDBListener`

      Initial state
              Started

      Connection factory JNDI name
              `jms/PiShopReplyCF`

Destination JNDI name
> jms/wasQueue

    d. If everything is set up correctly, click **Cancel**, otherwise make any required changes, then click **OK**.

5. Check the connection factory settings.

    a. In the administrative console, click **Resources** > **JMS** > **Connection factories**.

    b. Click **PiShopReplyCF**. The Connection factories settings page is displayed.

    c. On the Connection factories settings page, under General Properties, check that the following values appear in the required fields:

Name   PiShopReplyCF

JNDI name
> jms/PiShopReplyCF

Queue manager
> sampleQM

Hostname
> The host name of the first WebSphere MQ host. If you are using WebSphere Application Server and WebSphere MQ on the same computer, it is localhost.

Port   1414 by default. If you specified a port number during the WebSphere MQ configuration, it is the number you specified.

Server connection channel
> JMS.CHANNEL

Transport type
> CLIENT

    d. If everything is set up correctly, click **Cancel**, otherwise make any required changes, then click **OK**.

6. Check the queue settings.

    a. In the administrative console, click **Resources** > **JMS** > **Queues**. The Queues settings page is displayed.

    b. Click **wasQueue**.

    c. Check that the following values appear in the required fields.

Name   wasQueue

JNDI name
> jms/wasQueue

Base queue name
> myQueue

    d. If everything is set up correctly, click **Cancel**, otherwise make any required changes, then click **OK**.

7. If you have made any changes, save these changes to the master configuration, then stop and then restart the application server.

    a. Click the **Save** link in the Messages section at the top of the pane to save your changes to the master configuration.

    b. Stop the application server. In a terminal, navigate to /opt/IBM/WebSphere/ AppServer1/profiles/AppSrv01/bin, and enter ./stopServer.sh server1.

    c. Start the application server. In the same terminal, enter ./startServer.sh server1

Results

You have confirmed that the listener port, connection factory, and queue from your WebSphere Application Server Version 6.1 installation have successfully migrated to WebSphere Application Server Version 8.5.

What to do next

Follow the instructions in "Verifying the migrated sample MDB application" to verify that messages are being passed between WebSphere MQ Version 6.0 and WebSphere Application Server Version 8.5.

Verifying the migrated sample MDB application

After migrating to WebSphere Application Server to Version 8.5, use the Version 6 sample applications to verify that messages are being passed between WebSphere MQ Version 6.0 and WebSphere Application Server Version 8.5.

About this task

The migration of your WebSphere Application Server Version 6.1 configuration to Version 8.5 included the Version 6 sample MDB application. In this task, you use the Version 6 sample JMS requester application running in WebSphere MQ Version 6.0 to communicate with the Version 6 sample MDB application running in WebSphere Application Server Version 8.5. If the requester application successfully receives a reply from the MDB application, it confirms that messages are being passed between WebSphere MQ Version 6.0 and WebSphere Application Server Version 8.5.

Procedure

1. If sampleQM is not started, log in as the mqm user and then enter `strmqm sampleQM`.

2. Log in as root user.

3. If WebSphere Application Server is not already started, open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and then enter `./startServer.sh server1`. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

4. Start a web browser, in the address field enter *host_name*`:9061/ibm/console`. The *host_name* is the name of your WebSphere Application Server host. If the browser is on the same computer as WebSphere Application Server, the value of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when you installed the product. If you did not enable administrative security, the user ID is used only to distinguish changes that you make in the administrative console from changes that are made by other users.

5. Check the status of the listener port PiShopReplyMDBListener then, if it is not already started, start it:

   a. In the administrative console, click **Servers** > **Server Types** > **WebSphere application servers** > *server_name* > **[Communications] Messaging** > **Message Listener Service** > **Listener Ports**.

  b. Check whether the status of the listener port is started. If listener port is running, the status is shown as a green arrow. If the listener port is not running, the status is shown as a red cross.

  c. If the listener port is not already started, select it then click **Start**. The status of the listener port changes to a green arrow.

6. Check whether the sample MDB application, `sampleMDBv6`, is already running then, if it is not running, start it:

  a. Click **Applications** > **Application Types** > **WebSphere enterprise applications** to display a list of installed applications.

  b. Check whether the application status of `sampleMDBv6` is started. If `sampleMDBv6` is running, the application status is shown as a green arrow. If `sampleMDBv6` is not running, the application status is shown as a red cross.

  c. If `sampleMDBv6` is not already running, select it then click **Start**. The application status changes to a green arrow. The MDB application is now waiting to receive a message.

  If you see an error message similar to the following message, there might be a mismatch between your WebSphere Application Server configuration and the information contained in the sample application. Check the WebSphere Application Server error logs in *install_directory*/profiles/AppSrv01/logs/server1 for more information.

```
Error sampleMDB failed to start. Check the logs for server server1 on node
    mymachineNode01 for details.
Error An error occurred while starting sampleMDB. Check the logs for
    server server1 on node mymachineNode01 for more information.
```

  Note: If you update the WebSphere Application Server configuration, you must restart the application server for the changes to take effect.

7. Log off as root and log in as mqm.

8. Run the requestor client.

  a. On the command line, navigate to the directory that contains the `sampleJMSApp.jar` and the `runreq.sh` files.

  b. Enter `./runreq.sh`.

Results

The sample requester application sends the request message, and then waits for a response from the sample MDB application. The following output, showing sent and received messages, is displayed in the requester application Command Prompt window:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Connection created.

> Sending stock request for 'BakedBeans'
> Sent Message ID=ID:414d512073616d706c65514d20202020c690835102160020

> Received Message ID=ID:414d512073616d706c65514d20202020c690835105130020 for 'From V6 MDB: BakedBeans - 6 tins in stock

> Closing connection to QueueManager.
> Closed Connection.
```

You have finished verifying that your WebSphere Application Server configuration can still connect to WebSphere MQ Version 6.0 for the successful transmission of messages, following the migration of WebSphere Application Server to Version 8.5.

What to do next

Follow the instructions in "Phase 2: Migrating WebSphere MQ to Version 7.5" to begin phase 2 of this scenario.

Phase 2: Migrating WebSphere MQ to Version 7.5

Make a backup copy of the queue manager data, then migrate WebSphere MQ from Version 6.0 to Version 7.5. After completing the migration, check that you can send and receive messages through WebSphere MQ Version 7.5 using the provided sample JMS application.

About this task

In this scenario, WebSphere MQ Version 6.0 is not uninstalled before the migration starts. You can also migrate to WebSphere MQ Version 7.5 on a system where a previous version has been uninstalled but where the queue manager data has been retained.

Procedure

1. Follow the instructions in "Preparing for then migrating to WebSphere MQ Version 7.5" to back up the queue manager data and migrate WebSphere MQ from Version 6.0 to Version 7.5.
2. Follow the instructions in "Testing the Version 7 sample JMS application" on page 57 to run the Version 7 sample JMS application to send and receive messages through WebSphere MQ, and confirm that your migrated WebSphere MQ configuration works correctly with the Version 7 sample application.

Results

WebSphere MQ Version 7.5 is now installed, and all persistent items have been migrated from the previous version of WebSphere MQ. You have successfully run the Version 7 sample JMS application to send and receive messages through WebSphere MQ.

What to do next

After completing the phase 2 tasks, move on to "Phase 3: Verifying the solution" on page 59.

Related information:

WebSphere MQ V6 documentation

Preparing for then migrating to WebSphere MQ Version 7.5

Back up the queue manager data, then migrate WebSphere MQ from Version 6.0 to Version 7.5 by using the command line.

Before you begin

The general prerequisites for installing WebSphere MQ Version 7.5 are much the same as for installing WebSphere MQ Version 6. If you created the sample initial Version 6 IT configuration, then you have already satisfied these prerequisites. For more information, see the "Before you begin" section in "Installing WebSphere MQ Version 6.0" on page 19.

About this task

- This task describes the steps for migrating using a WebSphere MQ 7.5 on the Red Hat 64-bit Linux operating system.
- You must back up your queue manager data before you migrate to version 7.5. You cannot revert a queue manager to version 6.0 once you start it on version 7.5.
- You must uninstall WebSphere MQ Version 6.0 before installing WebSphere MQ version 7.1

Procedure

1. Log in as mqm.
2. End all WebSphere activity.
   a. Use the `dspmq` command to display the state of all the queue managers on the system.
   b. Use the `endmqm` command to stop all running queue managers.
   c. Stop any listeners associated with the queue managers, using the command:

```
endmqlsr -m QMgrName
```

3. Log out mqm and log in as root.
4. Check that you have stopped all of the necessary processes. In a terminal, enter the following command:

```
ps aug | grep mq
```

   Make sure no processes listed begin with amq or runmq; otherwise kill those processes.

5. Back up the queue manager data for the queue manager sampleQM, and the default queue manager, if you have created one. Take copies of all of the following data, making sure that you include all backup directories:
   - The queue manager data in `/var/mqm/qmgrs`.
   - The log file directories for the queue managers in `/var/mqm/log`, including the log control file `amqhlctl.lfh`.
   - The configuration files in the `/var/mqm/config`.
   - The default configuration file, `JMSAdmin.config` in `/opt/mqm/java/bin`. This file is deleted when WebSphere MQ Version 6.0 is uninstalled, so must be backed up before starting the migration.

   Some of the directories might be empty, but you need them all if you need to restore the backup later, so save them too.

   > **Why am I doing this?**
   > If you migrate from a previous level of WebSphere MQ without first backing up your system, you cannot revert to your previous level. Backing up your system before you install the new version enables you to back out the upgrade if necessary. If you back out the upgrade, however, you cannot recover any work, such as changes to messages and objects, carried out by WebSphere MQ Version 7.5.

6. Uninstall WebSphere MQ using the `rpm` command.
   a. Find out the names of the packages (components) currently installed on your system, by entering the following command:

```
rpm -qa | grep MQSeries
```

b. Remove all the components including fix packs at the same time by appending all the package names to the rpm command arguments. For example, to remove all the components displayed in the previous step, enter the following command:

```
rpm -ev `rpm -qa | grep MQSeries`
```

c. Check the packages are successfully uninstalled. Enter the following command again:

```
rpm -qa | grep MQSeries
```

If no package is displayed, you've uninstalled WebSphere MQ successfully.

7. Insert the WebSphere MQ Version 7.5 Server CD into the CD-ROM drive.

8. Accept the WebSphere MQ license.

a. In a terminal, navigate to the / directory. Enter ./mqlicense.sh -accept. The following message is displayed:

```
Licensed Materials - Property of IBM

 5724-H72

 (C) Copyright IBM Corporation 1994, 2019 All rights reserved.

US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.



Agreement accepted:  Proceed with install.
```

9. If /opt/mqm is not empty, delete this folder. Enter the following command:

```
rm -rf /opt/mqm
```

10. Install the WebSphere MQ packages. Enter the following command:

```
rpm -ivh MQSeriesRuntime-*.rpm MQSeriesServer-*.rpm MQSeriesSamples-*.rpm MQSeriesJava-*.rpm
```

Why am I doing this?

To complete this scenario, The minimum components that you must install are as follows:

- MQSeriesRuntime
- MQSeriesServer
- MQSeriesSamples
- MQSeriesJava

What else might I do or be interested in?

- To install all the components of WebSphere MQ, enter `rpm -ivh MQSeries*.rpm`.

- If you want to install to a non-default location, use the **rpm --prefix** command. The installation path must be to an empty directory, the root of an unused file system, or a path that does not exist. The length of the path is limited to 256 bytes and must not contain spaces. For example, to install the runtime and server components to /opt/customLocation, use the following command:

  `rpm --prefix /opt/customLocation -ivh MQSeries*.rpm`

The following is displayed throughout the installation process:

```
Preparing...                    ######################################## [100%]
   1:MQSeriesRuntime            ######################################## [ 25%]
   2:MQSeriesServer             ######################################## [ 50%]
   3:MQSeriesSamples            ######################################## [ 75%]
   4:MQSeriesJava               ######################################## [100%]
```

WebSphere MQ Version 7.5 is successfully installed to /opt/mqm directory.

Results

WebSphere MQ Version 7.5 is now installed, and all persistent items have been migrated from the previous version of WebSphere MQ.

What to do next

Follow the instructions in "Testing the Version 7 sample JMS application" to test that the Version 7 sample application will run in your migrated configuration.

Testing the Version 7 sample JMS application

Run the Version 7 sample JMS application to send and receive messages through WebSphere MQ, and confirm that your migrated WebSphere MQ configuration works correctly with the Version 7 sample application.

Before you begin

If you have not already done so, download the Version 7 sample JMS application package. Click the following link and save the file to the machine that hosts WebSphere Application Server: sampleJMSApp_V7.zip, then extract the contents. The package contains a sample JMS application .jar file and batch files for running the application. The package contains a sample JMS application .jar file and shell script files for running the application.

- The sample sampleJMSApp.jar file and the .sh files must be in the same directory.
- The .sh files use environment variables to set the class path for running the JMS application. When the JMS application is run, if you see a Java java.lang.NoClassDefFoundError, you might be required to adjust the class path line in the command file.

About this task

The JMS application comprises a requester client, which sends the initial message, and a responder client, which receives the message and sends a reply.

- In one terminal run the runresponder.sh, responder client then waits for a message.
- In another terminal, run the runrequester.sh, the requester client starts and sends a request message and receives a reply.

With two terminals, you can see the actions of the requester and responder separately and more clearly.

Procedure

1. Log in as the mqm user.
2. Start queue manager sampleQM.

a. In a terminal, navigate to /opt/mqm/bin.

b. Enter the command: ./strmqm sampleQM. The following message is displayed:

```
WebSphere MQ queue manager 'sampleQM' starting.
The queue manager is associated with installation 'Installation2'.
5 log records accessed on queue manager 'sampleQM' during the log replay phase.
Log replay for queue manager 'sampleQM' complete.
Transaction manager state recovered for queue manager 'sampleQM'.
Migrating objects for queue manager 'sampleQM'.
Default objects statistics : 35 created. 3 replaced. 0 failed.
WebSphere MQ queue manager 'sampleQM' started using V7.5.0.1.
```

3. Run the responder client.

a. On the command line, navigate to the directory that contains the sampleJMSApp.jar and the runresponder.sh files.

b. Enter ./runresponder.sh. The responder client waits for a message.

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Created connection.

> Waiting for message.
```

4. Run the requester client.

a. On the command line, navigate to the directory that contains the sampleJMSApp.jar and the runrequestor.sh files.

b. Enter ./runrequestor.sh.

Results

The requester client terminal, shows the connection status, the message it sent, and the reply message that it received from the responder client:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Connection created.

> Sending stock request for 'BakedBeans'
> Sent Message ID=ID:414d512073616d706c65514d202020204fb8835102170020

> Received Message ID=ID:414d512073616d706c65514d202020204fb8835102150020 for 'BakedBeans - 15 tins in stock'

> Closing connection to QueueManager.
> Closed Connection.
-----------------------
In this window, observe the messages sent through WebSphere MQ.
- The request message sent
- The reply message received
---
When ready, press any key to close the window.
```

The responder client terminal, shows the updated responder messages; the message it received (from the requester client) and the reply message that it sent:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Created connection.

> Waiting for message.

> Received Message ID=ID:414d512073616d706c65514d202020204fb8835102170020 for 'BakedBeans'
```

```
> Sending Reply Message 'BakedBeans - 15 tins in stock'
> Sent Message ID=ID:414d512073616d706c65514d202020204fb8835102150020

> Closing connection to QueueManager.
> Closed connection.
-------------------------------------------------------
In this window, observe the updated responder messages
- The request message received from the requester
- The reply message sent
-----
When ready, press any key to close this window
```

The messages that are shown in the two terminals verify that the requester and responder clients of the sample application can communicate with each other through WebSphere MQ.

What to do next

Follow the instructions in "Phase 3: Verifying the solution" on page 86 to confirm that WebSphere Application Server and WebSphere MQ can still communicate with each other following the migration of both products.

Phase 3: Verifying the solution

Verify that WebSphere Application Server and WebSphere MQ can still communicate with each other following the migration of WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5.

Procedure

1. Follow the instructions in "Deploying the Version 8 sample MDB application" to deploy the sample MDB that you need to use to complete the final phase of testing.
2. Follow the instructions in "Verifying the migrated IT configuration" on page 62 to complete the final stage of testing the migrated WebSphere Application Server and WebSphere MQ configuration.

Results

Following the migration of WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5, WebSphere Application Server can still connect to WebSphere MQ and messages can be transmitted successfully between the two products.

What to do next

After completing phase 3, you are ready to move on to the final stage of the scenario, which involves replacing the migrated listener port with an activation specification. To complete this final stage, follow the instructions in "Migrating WebSphere Application Server listener ports to activation specifications" on page 91.

Deploying the Version 8 sample MDB application

Using the WebSphere Application Server administrative console, install the Version 8 sample message driven bean (MDB) application so that you can use it to verify that WebSphere Application Server is communicating with WebSphere MQ following the migration of WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5.

Before you begin

If you have not already done so, download the Version 8 sample MDB application. Click the following link and save the file to the machine that hosts WebSphere Application Server: sampleMDB.ear">sampleMDB.ear.

About this task

The Version 8 sample MDB application, `sampleMDB.ear`, should now be able to use the objects that you migrated from WebSphere Application Server Version 6 to Version 8.5. The MDB application uses these objects to send a message to WebSphere MQ, for receipt by the sample JMS application requester client that you used in "Testing the Version 7 sample JMS application" on page 57.

Procedure

1. Log in as root user.
2. If WebSphere Application Server is not already started, open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and then enter `./startServer.sh server1`. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

3. Start a web browser, in the address field enter *host_name*`:9061/ibm/console`. The *host_name* is the name of your WebSphere Application Server host. If the browser is on the same computer as WebSphere Application Server, the value of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when you installed the product. If you did not enable administrative security, the user ID is used only to distinguish changes that you make in the administrative console from changes that are made by other users.
4. Click **Applications** > **New Application** > **[Install a New Application] New Enterprise Application**.
5. On the first Preparing for the application installation page, select the MDB application that you want to upload and install:
   a. Select **Local file system**, then click **Browse**. The File Upload window opens.
   b. In the File Upload window, browse to the folder where you downloaded the sample MDB application, select `sampleMDB.ear`, then click **Open**.
   c. Click **Next**.

   > **What else might I do or be interested in?**
   > Unless you specify otherwise, the name of the installed application is derived from the name of the MDB EAR file, which in the case of this scenario is `sampleMDB.ear`. If you prefer, you can give the application a different name. To do this, click the **Step 1** link to go to the Select installation options page then specify your chosen **Application name**.

6. On the second Preparing for the application installation page, leave the default setting **Fast Path**, then click **Next**.
7. On the Install New Application page, click the **Step 3** link to go to the Bind listeners for message-driven beans page.

> **What else might I do or be interested in?**
> To save having to specify the binding whenever you deploy the MDB
> EAR file, you can bind the MDB to a listener port name by using an
> assembly tool. For more information, see the topic Preparing for
> application installation binding settings in the WebSphere Application
> Server information center.

8. On the Bind listeners for message-driven beans page specify the listener port
   name to which the sample MDB application must be bound:

   a. Under **Listener Bindings**, select **Listener port**.

   b. Type the **Name** of the listener port: `PiShopReplyMDBListener`

      Tip: This is the listener port that you migrated from WebSphere
      Application Server Version 6.1 as described in "Migrating your WebSphere
      Application Server configuration to Version 8.5" on page 46.

   c. Click **Next**.

9. On the Map resource references to resources page, map the connection factory
   resource reference defined in the sample MDB application to the JNDI name
   of the connection factory.

   a. Under **Target Resource JNDI Name**, click **Browse** to display a list of
      resources that are available to be used to bind the resource reference of the
      MDB.

   b. Select `PiShopReplyCF` then click **Apply**.

      Tip: For this scenario, the resource that you select is the connection factory
      PiShopReplyCF that you migrated from WebSphere Application Server
      Version 6.1 as described in "Migrating your WebSphere Application Server
      configuration to Version 8.5" on page 46.

   c. Leave all other fields unchanged and click **Next**.

10. Click **Next** again to go to the Summary page.

11. On the Summary page, click **Finish**. The program displays messages to
    indicate that the application is being installed. If the application does not
    install successfully, examine the messages to identify why the installation
    failed. Correct the problems as needed and try installing the application again.

12. When the application installs successfully, click the **Save** link that is displayed
    at the end of the messages, to save your changes to the master configuration.

13. Restart the application server.

    a. Stop the application server. In a terminal, navigate to `/opt/IBM/WebSphere/`
       `AppServer/profiles/AppSrv01/bin`, and enter `./stopServer.sh server1`.

    b. Start the application server. In the same terminal, enter `./startServer.sh`
       `server1`

Results

You have deployed the MDB application, which is now visible in the Enterprise
Applications panel. To view this panel, click **Applications** > **Application Types** >
WebSphere enterprise applications.

What to do next

Follow the instructions in "Verifying the migrated IT configuration" to run the sample applications and verify that messages can be passed between the two products.

Verifying the migrated IT configuration

Run the sample JMS and message drive bean (MDB) applications to verify that your migrated WebSphere Application Server and WebSphere MQ configurations can communicate with each other.

About this task

In "Testing the Version 7 sample JMS application" on page 57, you ran the supplied JMS sample application to verify that the requester and responder clients of the application could communicate through WebSphere MQ. In this task, you run the requester application as before, but this time the reply comes from the MDB application that you installed in "Deploying the Version 8 sample MDB application" on page 59, instead of from the responder application, verifying that messages are being passed between WebSphere MQ and WebSphere Application Server.

Procedure

1. If sampleQM is not started, log in as the mqm user and then enter `strmqm sampleQM`.
2. Log in as root user.
3. If WebSphere Application Server is not already started, open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and then enter `./startServer.sh server1`. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

4. Start a web browser, in the address field enter *host_name*`:9061/ibm/console`. The *host_name* is the name of your WebSphere Application Server host. If the browser is on the same computer as WebSphere Application Server, the value of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when you installed the product. If you did not enable administrative security, the user ID is used only to distinguish changes that you make in the administrative console from changes that are made by other users.

5. Check the status of the listener port PiShopReplyMDBListener then, if it is not already started, start it:
   a. In the administrative console, click **Servers** > **Server Types** > **WebSphere application servers** > *server_name* > **[Communications] Messaging** > **Message Listener Service** > **[Additional Properties] Listener Ports**.
   b. Check whether the status of the listener port is started. If listener port is running, the status is shown as a green arrow.
   c. If the listener port is not already started, select it then click **Start**. The status of the listener port changes to a green arrow.
6. Check whether the Version 8 sample MDB application, sampleMDB, is already running then, if it is not running, start it:
   a. Click **Applications** > **Application Types** > **WebSphere enterprise applications** to display a list of installed applications.

b. Check whether the application status of sampleMDB is started. If sampleMDB is running, the application status is shown as a green arrow.

c. If the sampleMDB is not already running, start it. Select the check box next to sampleMDB, then click **Start**. The application status changes to a green arrow. The MDB application is now waiting to receive a message.

If you see an error message similar to the following message, there might be a mismatch between your WebSphere Application Server configuration and the information contained in the sample application. Check the WebSphere Application Server error logs in *install_directory*/profiles/AppSrv01/logs/server1 for more information.

```
Error sampleMDB failed to start. Check the logs for server server1 on node
    mymachineNode01 for details.
Error An error occurred while starting sampleMDB. Check the logs for
    server server1 on node mymachineNode01 for more information.
```

7. Optional: Stop the Version 6 MDB application, sampleMDBv6, if it is running. Select the check box next to sampleMDBv6, then click **Stop**.

> **Why am I doing this?**
> You do not need to stop the Version 6 MDB application, sampleMDBv6, to be able to run the Version 8 MDB. However, stopping the Version 6 MDB better makes it clear that the message handling is being done by the Version 8 MDB.

8. Log in to the WebSphere MQ host as the mqm user.

9. Run the requester client.

   a. On the command line, navigate to the directory that contains the sampleJMSApp.jar and the runreq.sh files.

   b. Enter ./runreq.sh.

Results

The sample requester application sends the same request message as before, but the response now comes from the MDB application. The following output, showing sent and received messages, is displayed in the requester application Command Prompt window:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Connection created.

> Sending stock request for 'BakedBeans'
> Sent Message ID=ID:414d512073616d706c65514d202020204fb88351021a0020

> Received Message ID=ID:414d512073616d706c65514d202020204fb88351041c0020 for 'From MDB: BakedBeans - 15 tins in stock. |

> Closing connection to QueueManager.
> Closed Connection.
----------------------
In this window, observe the messages sent through WebSphere MQ.
- The request message sent
- The reply message received
---
When ready, press any key to close the window.
```

If you see errors similar to the following error in the WebSphere Application Server logs, check that the name of the queue in WebSphere Application Server matches the administrative name (not the JNDI name) of the queue in WebSphere MQ:

```
WebSphere MQ call failed with compcode '2' ('MQCC_FAILED') reason '2085'
    ('MQRC_UNKNOWN_OBJECT_NAME')
com.ibm.msg.client.jms.DetailedInvalidDestinationException: JMSWMQ2008:
    Failed to open MQ queue 'myQueue'. JMS attempted to perform an MQOPEN,
    but WebSphere MQ reported an error. Use the linked exception to determine
    the cause of this error. Check that the specified queue and queue manager
    are defined correctly.
```

Note: If you update the WebSphere Application Server configuration, you must restart the application server for the changes to take effect.

You have finished migrating WebSphere Application Server from Version 6 to Version 8.5 and then WebSphere MQ from Version 6 to Version 7, and verified that, following the migration, WebSphere Application Server can still connect to WebSphere MQ for the successful transmission of messages.

You have now completed Option 1 of this scenario.

What to do next

At this point, the Version 8.5 sample MDB application is still using the listener port that you migrated from WebSphere Application Server Version 6 to connect to WebSphere MQ. To complete the delivered logical topology described in "Overview: The delivered logical topology" on page 10, you now need to create an activation specification and configure the Version 8 sample MDB application to use this activation specification instead of the listener port to listen for and respond to messages received from WebSphere MQ. To complete this final stage of the scenario, follow the instructions in "Migrating WebSphere Application Server listener ports to activation specifications" on page 91.

Uninstalling WebSphere MQ and WebSphere Application Server

If you want to try one migration option then the other, you should uninstall the WebSphere MQ and WebSphere Application Server files that are used by the first option before starting the second option. After finishing this scenario, you might also want to uninstall all WebSphere MQ and WebSphere Application Server files used for the scenario.

About this task

The subtopics provide instructions to uninstall all applications that are involved in the migration. The subtopics can be used both individually or as a complete uninstallation procedure. The topics ensure that other programs and platforms installed throughout the Migration process, such as IBM Installation Manager is removed from your system. There are details about folders that should be deleted, which are left behind after the uninstallation is complete, which is important to prevent problems occurring when reinstalling.

Uninstalling WebSphere MQ

Stop, and then uninstall WebSphere MQ Version 7.5. Finally, delete any folders or files that are left behind. At the end of this task, you are ready to reinstall WebSphere MQ.

About this task

This task describes the basic steps to remove WebSphere MQ for Red Hat 64-bit Linux operating system.

Procedure

1. Before starting to uninstall, end all WebSphere activity.
   a. Log in as mqm.
   b. Use the `dspmq` command to display the state of all the queue managers on the system.
   c. Use the `endmqm` command to stop all running queue managers.
   d. Stop any listeners associated with the queue managers, using the command:

```
endmqlsr -m QMgrName
```

2. Log out mqm and log in as root.
3. Check that you have stopped all of the necessary processes. In a terminal, enter the following command:

```
ps aug | grep mq
```

   Make sure no processes listed begin with amq or runmq; otherwise kill those processes.
4. Uninstall WebSphere MQ using the `rpm` command.
   a. Find out the names of the packages (components) currently installed on your system, by entering the following command:

```
rpm -qa | grep MQSeries
```

   b. Remove all the components at the same time by appending all the package names to the rpm command arguments. For example, to remove the runtime, Server, Samples, and Java components enter the following command:

```
rpm -ev MQSeriesRuntime MQSeriesServer MQSeriesSamples MQSeriesJava
```

   c. Check the packages are successfully uninstalled. Enter the following command again:

```
rpm -qa | grep MQSeries
```

   If no package is displayed, you have uninstalled WebSphere MQ successfully.
5. After uninstalling, you might want to delete the contents of `/var/mqm`, which is not removed by the uninstallation.

   Note: Do this only if you do not need your queue manager data.

Results

WebSphere MQ Version 7.5 have been uninstalled. Any folders that were left behind from the uninstallation process have been removed.

Uninstalling WebSphere Application Server Version 8.5

Stop, and then uninstall WebSphere Application Server Version 8.5. Delete any folders or files that are left behind. At the end of this task, you will be ready to reinstall WebSphere Application Server Version 8.5.

About this task

This task describes the steps to uninstall WebSphere Application Server Version 8.5 on the Rat Hat 64-bit Linux operating system.

Procedure

1. Log in as root.
2. Stop the WebSphere Application Server Version 8.5 application server. In a terminal window, navigate to `/opt/IBM/WebSphere/AppServer1/bin`, and then enter the command:

```
./stopServer.sh server1
```

   If the following information is displayed, it means the server is stopped successfully:

```
Server server1 stop completed.
```

3. Remove WebSphere Application Server Version 8.5.

   a. Start IBM Installation Manager. In a terminal, navigate to `/opt/IBM/InstallationManager/eclipse`, and then enter `./launcher` Then the IBM Installation Manager window is displayed.

   b. Click **Uninstall**

   c. In the Uninstall Packages windows, select **IBM WebSphere Application Server V8.5**, and then click **Next**.

   d. If the following message is displayed, stop the servers in all profiles of WebSphere Application Server V8.5, and click **Recheck Status**. .

```
Running processes have been detected that may interfere with the current operation. Stop all WebSphere and related processe
```

   When the following message is displayed, click **Next**.

```
All prerequisites are met.
```

   e. Check the uninstallation summary, and then click **Uninstall** to start the uninstallation process. When the uninstallation process is completed successfully, a success message is displayed.

   f. Click the **View Log File** link to view the details of the uninstallation, and then click **Finish**.

4. Remove IBM Installation Manager. Remove this software only if you are not using any product of IBM that is Version 8 or later.

   a. From a terminal window, go to the uninstallation directory of Installation Manager. By default, it is `/var/ibm/InstallationManager/uninstall`.

   b. Enter the `./uninstall` command. The Uninstall Package window is displayed.

   c. Select **IBM Installation Manager**, and then click **Next**.

   d. Check the uninstallation summary, and then click **Uninstall** to start the uninstallation process. When the uninstallation process is completed successfully, a success message is displayed.

   e. Click the **View Log File** link to view the details of the uninstallation, and then click **Finish**.

5. Delete the WebSphere Application Server Version 8.5 folder. By default, it is `/opt/IBM/WebSphere/AppServer1`.

Results

WebSphere Application Server Version 8.5 and IBM Installation Manager Version 1.5.2 are uninstalled.

Uninstalling WebSphere Application Server Version 6.1

Stop, and then uninstall WebSphere Application Server Version 6.1 and delete any folders or files that are left behind. At the end of this task, you will be ready to reinstall WebSphere Application Server Version 6.1.

About this task

This task describes the steps to uninstall WebSphere Application Server on the Red Hat 64-bit Linux operating system.

Procedure

1. Log in as root.
2. Stop the WebSphere Application Server Version 6.1 application server. Navigate to `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin`, and then enter the command `./stopServer.sh server1`.
3. Issue the uninstall command.
   a. In a terminal window, navigate to `/opt/IBM/WebSphere/AppServer/uninstall`, and then enter `./uninstall`. The uninstall window is displayed.
   b. Click **Next**.
   c. Select **Remove all profiles**, and then click **Next**.
   d. If the following message is displayed, click **Cancel**, stop all the servers in the application server, and then restart the process of Step 3.

`Running processes have been detected that may interfere with the current operation. Before installing or uninstalling ma`

   e. Check the uninstallation summary, and then click **Next** to start the uninstallation process. When the uninstallation process is completed successfully, a success message is displayed.
4. Remove any folders left behind.
   a. Delete the WebSphere Application Server V6.1 folder. By default, it is `/opt/IBM/WebSphere/AppServer`.
   b. Delete the WebSphere Application Server migration folder. By default, it is `/opt/IBM/WebSphere/WSMigration`
   c. Now all WebSphere Application Server packages have been unistalled from your computer. You can delete the `WebSphere` folder. You should only delete this folder if there are no other WebSphere products installed in this folder.

Results

WebSphere Application Server Version 6.1 has been uninstalled.

What else might I do or be interested in?

For general help with problems uninstalling WebSphere Application Server, see the Uninstalling the product topic in the WebSphere Application Server information center.

Option 2: Migrating WebSphere MQ before WebSphere Application Server

In Option 2, you first migrate your WebSphere MQ Version 6 configuration to Version 7.5 and use the Version 6 sample applications to check that your migrated configuration can still communicate with WebSphere Application Server, which is still at Version 6. You then migrate WebSphere Application Server to Version 8.5 and use the sample Version 8 applications to check that messages can still be passed between WebSphere MQ and WebSphere Application Server.

Before you begin

The starting point for this scenario is the sample Version 6 IT configuration described in "Overview: Initial IT configuration" on page 6.

- If you have not already created the sample Version 6 IT configuration, complete the tasks described in "Creating a sample initial Version 6 IT configuration" on page 18.
- If you have already completed "Option 1: Migrating WebSphere Application Server before WebSphere MQ" on page 39, and you now want to try "Option 2: Migrating WebSphere MQ before WebSphere Application Server," you first must uninstall both products used in the other option (to reset the installation), and then set up the sample Version 6 IT configuration again as described in the following topics:
  1. "Uninstalling WebSphere MQ" on page 64
  2. "Uninstalling WebSphere Application Server Version 8.5" on page 65
  3. "Uninstalling WebSphere Application Server Version 6.1" on page 67
  4. "Creating a sample initial Version 6 IT configuration" on page 18

Phase 1: Migrating WebSphere MQ to Version 7.5

Make a backup copy of the queue manager data, and then migrate WebSphere MQ from Version 6.0 to Version 7.5. After completing the migration, check that you can send and receive messages through WebSphere MQ Version 7.5 using the provided sample JMS application.

Before you begin

The starting point for this scenario is a clean sample Version 6 IT configuration. If you have not already done so, read the section "Before you begin" in the topic "Option 2: Migrating WebSphere MQ before WebSphere Application Server," and complete any appropriate actions described there.

About this task

In this scenario, WebSphere MQ Version 6.0 is not uninstalled before the migration starts. You can also migrate to WebSphere MQ Version 7.5 on a system where a previous version has been uninstalled but where the queue manager data has been retained.

Procedure

1. Follow the instructions in "Preparing for then migrating to WebSphere MQ Version 7.5" on page 69 to back up the queue manager data and migrate WebSphere MQ from Version 6.0 to Version 7.5.

2. Follow the instructions in "Testing the Version 7 sample JMS application" on page 72 to run the Version 7 sample JMS application to send and receive messages through WebSphere MQ, and confirm that your migrated WebSphere MQ configuration works correctly with the Version 7 sample application.

Results

WebSphere MQ Version 7.5 is now installed, and all persistent items have been migrated from the previous version of WebSphere MQ. You have successfully run the Version 7 sample JMS application to send and receive messages through WebSphere MQ.

What to do next

After completing the phase 1 tasks, move on to "Phase 2: Migrating WebSphere Application Server to Version 8.5" on page 74.

Related information:

➦ WebSphere MQ V6 documentation

Preparing for then migrating to WebSphere MQ Version 7.5

Back up the queue manager data, then migrate WebSphere MQ from Version 6.0 to Version 7.5 by using the Launchpad and selecting the **Update** option of the installation wizard.

Before you begin

The general prerequisites for installing WebSphere MQ Version 7.5 are much the same as for installing WebSphere MQ Version 6, such as you would already have satisfied if you created the sample initial Version 6 IT configuration. For more information, see the "Before you begin" section in "Installing WebSphere MQ Version 6.0" on page 19.

About this task

- This task describes the steps for migrating using a WebSphere MQ 7.5 on the Red Hat 64-bit Linux operating system.
- You must back up your queue manager data before you migrate to version 7.5. You cannot revert a queue manager to version 6.0 once you start it on version 7.5.
- You must uninstall WebSphere MQ Version 6.0 before installing WebSphere MQ version 7.1

Procedure

1. Log in as mqm.
2. End all WebSphere activity.
   a. Use the `dspmq` command to display the state of all the queue managers on the system.
   b. Use the `endmqm` command to stop all running queue managers.
   c. Stop any listeners associated with the queue managers, using the command:

```
endmqlsr -m QMgrName
```

3. Log out mqm and log in as root.
4. Check that you have stopped all of the necessary processes. In a terminal, enter the following command:

```
ps aug | grep mq
```

> Make sure no processes listed begin with amq or runmq; otherwise kill those processes.

5. Back up the queue manager data for the queue manager sampleQM, and the default queue manager, if you have created one. Take copies of all of the following data, making sure that you include all backup directories:
   - The queue manager data in `/var/mqm/qmgrs`.
   - The log file directories for the queue managers in `/var/mqm/log`, including the log control file `amqhlctl.lfh`.
   - The configuration files in the `/var/mqm/config`.
   - The default configuration file, `JMSAdmin.config` in `/opt/mqm/java/bin`. This file is deleted when WebSphere MQ Version 6.0 is uninstalled, so must be backed up before starting the migration.

   Some of the directories might be empty, but you need them all if you need to restore the backup later, so save them too.

   ┌─ **Why am I doing this?** ─────────────────────────────
   │ If you migrate from a previous level of WebSphere MQ without first
   │ backing up your system, you cannot revert to your previous level.
   │ Backing up your system before you install the new version enables you
   │ to back out the upgrade if necessary. If you back out the upgrade,
   │ however, you cannot recover any work, such as changes to messages and
   │ objects, carried out by WebSphere MQ Version 7.5.
   └──

6. Uninstall WebSphere MQ using the `rpm` command.
   a. Find out the names of the packages (components) currently installed on your system, by entering the following command:

```
rpm -qa | grep MQSeries
```

   b. Remove all the components including fix packs at the same time by appending all the package names to the rpm command arguments. For example, to remove all the components displayed in the previous step, enter the following command:

```
rpm -ev `rpm -qa | grep MQSeries`
```

   c. Check the packages are successfully uninstalled. Enter the following command again:

```
rpm -qa | grep MQSeries
```

   If no package is displayed, you've uninstalled WebSphere MQ successfully.

7. Insert the WebSphere MQ Version 7.5 Server CD into the CD-ROM drive.

8. Accept the WebSphere MQ license.
   a. In a terminal, navigate to the / directory. Enter `./mqlicense.sh -accept`. The following message is displayed:

```
Licensed Materials - Property of IBM

 5724-H72

 (C) Copyright IBM Corporation 1994, 2019 All rights reserved.

US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
```

```
Agreement accepted:  Proceed with install.
```

9. If /opt/mqm is not empty, delete this folder. Enter the following command:

```
rm -rf /opt/mqm
```

10. Install the WebSphere MQ packages. Enter the following command:

```
rpm -ivh MQSeriesRuntime-*.rpm MQSeriesServer-*.rpm MQSeriesSamples-*.rpm MQSeriesJava-*.rpm
```

Why am I doing this?

To complete this scenario, The minimum components that you must install are:

- MQSeriesRuntime
- MQSeriesServer
- MQSeriesSamples
- MQSeriesJava

What else might I do or be interested in?

- To install all the components of WebSphere MQ, enter `rpm -ivh MQSeries*.rpm`.

- If you want to install to a non-default location, use the **`rpm --prefix`** command. The installation path must be to an empty directory, the root of an unused file system, or a path that does not exist. The length of the path is limited to 256 bytes and must not contain spaces. For example, to install the runtime and server components to /opt/customLocation, use the following command:

  ```
  rpm --prefix /opt/customLocation -ivh MQSeries*.rpm
  ```

The following is displayed throughout the installation process:

```
Preparing...               ######################################### [100%]
   1:MQSeriesRuntime        ######################################### [ 25%]
   2:MQSeriesServer         ######################################### [ 50%]
   3:MQSeriesSamples        ######################################### [ 75%]
   4:MQSeriesJava           ######################################### [100%]
```

WebSphere MQ Version 7.5 is successfully installed to /opt/mqm directory.

Results

WebSphere MQ Version 7.5 is now installed, and all persistent items have been migrated from the previous version of WebSphere MQ.

What to do next

Follow the instructions in "Testing the Version 7 sample JMS application" on page 72 to test that the Version 7 sample application will run in your migrated configuration.

Testing the Version 7 sample JMS application

Run the Version 7 sample JMS application to send and receive messages through WebSphere MQ, and confirm that your migrated WebSphere MQ configuration works correctly with the Version 7 sample application.

Before you begin

If you have not already done so, download the Version 7 sample JMS application package. Click the following link and save the file to the machine that hosts WebSphere Application Server: sampleJMSApp_V7.zip, then extract the contents. The package contains a sample JMS application `.jar` file and batch files for running the application. The package contains a sample JMS application `.jar` file and shell script files for running the application.

- The sample `sampleJMSApp.jar` file and the `.sh` files must be in the same directory.
- The `.sh` files use environment variables to set the class path for running the JMS application. When the JMS application is run, if you see a Java `java.lang.NoClassDefFoundError`, you might be required to adjust the class path line in the command file.

About this task

The JMS application comprises a requester client, which sends the initial message, and a responder client, which receives the message and sends a reply.

- In one terminal run the `runresponder.sh`, responder client then waits for a message.
- In another terminal, run the `runrequester.sh`, the requester client starts and sends a request message and receives a reply.

With two terminals, you can see the actions of the requester and responder separately and more clearly.

Procedure

1. Log in as the `mqm` user.
2. Start queue manager sampleQM.
   a. In a terminal, navigate to `/opt/mqm/bin`.
   b. Enter the command: `./strmqm sampleQM`. The following message is displayed:

```
WebSphere MQ queue manager 'sampleQM' starting.
The queue manager is associated with installation 'Installation2'.
5 log records accessed on queue manager 'sampleQM' during the log replay phase.
Log replay for queue manager 'sampleQM' complete.
Transaction manager state recovered for queue manager 'sampleQM'.
Migrating objects for queue manager 'sampleQM'.
Default objects statistics : 35 created. 3 replaced. 0 failed.
WebSphere MQ queue manager 'sampleQM' started using V7.5.0.1.
```

3. Run the responder client.
   a. On the command line, navigate to the directory that contains the `sampleJMSApp.jar` and the `runresponder.sh` files.
   b. Enter `./runresponder.sh`. The responder client waits for a message.

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Created connection.

> Waiting for message.
```

4. Run the requester client.

   a. On the command line, navigate to the directory that contains the
      sampleJMSApp.jar and the runrequestor.sh files.

   b. Enter ./runrequestor.sh.

Results

The requester client terminal, shows the connection status, the message it sent, and
the reply message that it received from the responder client:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Connection created.

> Sending stock request for 'BakedBeans'
> Sent Message ID=ID:414d512073616d706c65514d202020204fb8835102170020

> Received Message ID=ID:414d512073616d706c65514d202020204fb8835102150020 for 'BakedBeans - 15 tins in stock'

> Closing connection to QueueManager.
> Closed Connection.
----------------------
In this window, observe the messages sent through WebSphere MQ.
- The request message sent
- The reply message received
---
When ready, press any key to close the window.
```

The responder client terminal, shows the updated responder messages; the
message it received (from the requester client) and the reply message that it sent:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Created connection.

> Waiting for message.

> Received Message ID=ID:414d512073616d706c65514d202020204fb8835102170020 for 'BakedBeans'

> Sending Reply Message 'BakedBeans - 15 tins in stock'
> Sent Message ID=ID:414d512073616d706c65514d202020204fb8835102150020

> Closing connection to QueueManager.
> Closed connection.
--------------------------------------------------------
In this window, observe the updated responder messages
- The request message received from the requester
- The reply message sent
-----
When ready, press any key to close this window
```

The messages that are shown in the two terminals verify that the requester and
responder clients of the sample application can communicate with each other
through WebSphere MQ.

What to do next

Follow the instructions in "Phase 2: Migrating WebSphere Application Server to Version 8.5."

Phase 2: Migrating WebSphere Application Server to Version 8.5

Make a backup copy of your WebSphere Application Server Version 6.1 configuration, then install Version 8.5. After completing the installation, migrate the WebSphere Application Server resources to Version 8.5, then check that the migration of these resources and of the sample MDB application has been successful.

About this task

Phase 2 of this scenario covers the migration of WebSphere Application Server from Version 6 to Version 8.5. At the end of this phase, your WebSphere Application Server is at Version 8.5 and WebSphere MQ is at Version 7.5, and you are ready to test that messages can be passed between WebSphere MQ Version 7.5 and WebSphere Application Server Version 8.5 using the sample applications.

The migration process for WebSphere Application Server involves installing a Version 8.5 environment alongside the existing Version 6 environment, then migrating your Version 6 configuration to the Version 8.5 environment.

Procedure

1. Follow the instructions in "Preparing to migrate to WebSphere Application Server Version 8.5" on page 75 to prepare for migrating to WebSphere Application Server Version 7.0 and the latest fix pack.

2. Follow the instructions in "Migrating your WebSphere Application Server configuration to Version 8.5" on page 80 to migrate your WebSphere Application Server Version 6.1 configuration to Version 8.5, using the main migration wizard.

> **What else might I do or be interested in?**
> This scenario does not require the use of any migration tools other than the migration wizard. However, in more complex scenarios, you might need to use additional migration tools to complete any upgrades that are not done automatically by the migration wizard; for example, upgrade of JSP objects or Derby databases. For more information about migration tools, see the Migrating product configurations with migration tools topic in the WebSphere Application Server Version 8.5 information center.

3. Follow the steps in "Confirming migration of WebSphere Application Server resources to Version 8.5" on page 83 to check that the listener port, connection factory, and queue from your WebSphere Application Server Version 6.1 installation have successfully migrated to WebSphere Application Server Version 8.5.

Results

You have installed WebSphere Application Server Version 8.5 and migrated your WebSphere Application Server resources from Version 6 to Version 8.5. WebSphere

MQ is already at Version 7.5, so you are now ready to complete the final testing of your migrated IT configuration.

What to do next

When you have confirmed that your migrated WebSphere Application Server is working as required in Version 8.5, you can, if required, remove WebSphere Application Server Version 6.1 using the uninstaller program that is created during installation. The uninstaller program removes registry entries, uninstalls the product, and removes all related features. You might also need to do some manual cleaning up, as the uninstaller program does not remove log files in the installation root directory. For more information, see "Uninstalling WebSphere Application Server Version 6.1" on page 67.

Note: If you migrate any profiles to a new version of the product, you must either shut down the new migrated deployment managers before uninstalling the product, or include the `removeProfilesOnUninstall="false"` parameter on the uninstall command. Either of these options prevents the migrated profiles from being deleted when you uninstall the old version of the product.

After completing the phase 2 tasks, move on to "Phase 3: Verifying the solution" on page 86.

Preparing to migrate to WebSphere Application Server Version 8.5

Prepare to migrate your WebSphere Application Server Version 6.1 configuration to Version 8.5 by reviewing premigration considerations and making a backup copy of your Version 6.1 configuration. At the end of this task, you will be ready to install WebSphere Application Server Version 8.5 with the latest level of maintenance.

Before you begin
- You must have local root access when you are installing.
- Ensure that you have sufficient disk space, up to 2 GB for all components.
- Ensure that the default shell for your Linux operating system is `/bin/bash`. Use the following command `readlink /bin/sh` to ensure that your default shell is `bash` not `dash`.
- Check for any post release technical notes available for your operating system at the product support site Support home - WebSphere Application Server.
- Make sure that your Linux system has the following items:
  - Kernel and C runtime library
  - Current and all compatibility versions of the C++ runtime library
  - X Window System libraries
  - GTK runtime libraries
- Ensure that there is adequate disk space. Installing all features and products requires approximately 2 GB of disk space. This estimate includes the following products, components, and features:
  - Main application server product installation
  - Profiles
  - Sample applications
  - IBM HTTP Server
  - Web Server plug-ins

– Application Client for WebSphere Application Server
- The backup configuration utility creates an archive file into a folder of your choice. If the folder does not exist, create it now before continuing.

For more details about requirements for installation, see the IBM Support pages.

About this task

This task describes the steps for preparing to install WebSphere Application Server on the Red Hat 64-bit Linux operating system.

Procedure

1. Review the Premigration considerations topic in the WebSphere Application Server Version 8.5 information center.

   Tip: You do not need to complete this step for this scenario because the information and actions needed are provided within the scenario to make it easier for you to proceed. However, it is good general practice to review the latest premigration considerations, especially if preparing to migrate your own WebSphere Application Server Version 6 environment.

2. Log in as the root user.
3. Make a backup copy of your existing Version 6.1 configuration and make a note of the exact name and location of the backed-up configuration.

   Why am I doing this?

   The migration tools create a backup copy of the configuration from the previous version, as a file in zip format. Backing up your existing configuration before you start the migration allows you to restore the configuration to its previous state after migration.

   What else might I do or be interested in?

   You can use either the **backupConfig** utility or your own preferred backup utility to create the backup. The backupConfig command is a simple utility to back up the configuration of your node to a file. By default, all servers on the node stop before the backup is made so that partially synchronized information is not saved. If you do not have root authority, you must specify a path for the backup file in a location where you have write permission. The backup file will be in zip format and a .zip extension is recommended.

   Prerequisite: The directory into which you want to create the backup file must exist before running the **backupConfig** command.

   a. Open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin`.
   b. Enter the following command to run the backupConfig utility, specifying the target directory and file name.

```
./backupConfig.sh migbackupdir/backupfile.zip
```

For example:

```
./backupConfig.bat /opt/migbackupdir/myBackup.zip
```

After a short pause, the backup process starts, and information messages,
similar to those in the following example, are shown in the Command
Prompt window.

```
ADMU0116I: Tool information is being logged in file
          /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/backupConfig.log
ADMU0128I: Starting tool with the AppSrv01 profile
ADMU5001I: Backing up config directory
          /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config to file
          /opt/migbackupdir/myBackup.zip
ADMU0505I: Servers found in configuration:
ADMU0506I: Server name: server1
ADMU2010I: Stopping all server processes for node mylocalNode01
ADMU0510I: Server server1 is now STOPPED
```

When the backup is complete, a message confirming this is displayed in the
Command Prompt window:

```
ADMU5002I: 358 files successfully backed up
```

4. Ensure that prerequisites and corequisites are at the required release levels. On
   the WebSphere Application Server detailed system requirements web page,
   click the link for your platform to see information about supported operating
   systems, browsers, and hardware requirements.

   Tip: You do not need to complete this step for this scenario, because the
   information and actions needed are provided within the scenario to make it
   easier for you to proceed. However, it is good general practice to verify
   prerequisites and corequisites, especially if preparing to migrate your own
   WebSphere Application Server V6 environment.

Results

You are now ready to install WebSphere Application Server Version 8.5.

What to do next

Follow the instructions in "Installing WebSphere Application Server Version 8.5" on
page 78 to install WebSphere Application Server Version 8.5.

Installing WebSphere Application Server Version 8.5

Install WebSphere Application Server 8.5 by using Installation Manager. At the end of this task, you are ready to configure WebSphere Application Server to communicate with WebSphere MQ.

Before you begin

- You must have root access when you are installing.
- Ensure that you have sufficient disk space. You require up to 2 GB for all components.
- Ensure that prerequisites and corequisites are at the required release levels. On the WebSphere Application Server detailed system requirements page, click the link for your operating system and the link for software to see information about supported operating systems and browsers, and hardware requirements.

About this task

This task describes the basic steps to install WebSphere Application Server 8.5 on Red Hat 64-bit Linuxoperating system by using Installation Manager.

Procedure

1. Log in as root user.
2. If you haven't installed IBM Installation Manager before, install it.
   a. Extract the Installation Manager archive file.
   b. In a terminal, navigate to Installation Manager archive directory enter `./installc -c`. The following message is displayed:

```
Preprocessing the input.
Loading repositories...
Preparing and resolving the selected packages...


=====> IBM Installation Manager> Install

Select packages to install:
     1. [X] IBM® Installation Manager 1.5.2

     O. Check for Other Versions, Fixes, and Extensions

     N. Next,     C. Cancel
-----> [N]
```

   c. Enter N then A to accept the license agreement.
   d. Enter N until the installation summary like the following is displayed:

```
=====> IBM Installation Manager> Install> Licenses> Location> Summary

Target Location:
  Package Group Name         :  IBM Installation Manager
  Installation Directory     :  /opt/IBM/InstallationManager/eclipse

Packages to be installed:
        IBM® Installation Manager 1.5.2

Options:
     G. Generate an Installation Response File

     B. Back,      I. Install,      C. Cancel
-----> [I]
```

   Press Enter. The installation process begins. When the installation is complete, a the following message is displayed:

```
=====> IBM Installation Manager> Install> Licenses> Location> Summary>
  Completion

The install completed successfully.

Options:
    R. Restart Installation Manager
-----> [R]
```

     e. Press Enter, and then enter X to exit the Installation Manager.

3. In a terminal window, navigate to /opt/IBM/InstallationManager/eclipse, and then enter ./launcher. The IBM Installation Manager window is displayed.

4. Click **Install**.

5. If a message is displayed that states Installation Manager cannot find any packages to install, add WebSphere Application Server to the repository.

    a. Click the **Repositories** link in the **Repository connections** field.

    b. Click **Add Repository** in the upper right of the Preference window.

    c. Click **Browse** and find the repository.config file for WebSphere Application Server.

    d. Click **OK**.

    e. Click **Test Connections** on the right of the Preference window. If a message is displayed that states All the selected repositories are connected, WebSphere Application Server is added to the repository.

    f. Click **OK** to close the window and click **OK** again to confirm adding the repository.

6. Select **Version 8.5.0.0**, and then click **Next**.

7. Click **I accept the terms in the license agreement**, and then click **Next.**

8. Click **Next**, then click **Next** again.

9. Select the languages that you want and click **Next**.

10. Select the components that you want and click **Next**.

11. Click **Install**. A progress bar is displayed, which shows that the installation is in progress.

12. When the installation is complete, a window is displayed, which indicates that the packages are installed. Click **None**, and then click **Finish**. WebSphere Application Server is successfully installed.

Results

WebSphere Application Server is installed and a profile is created. If you encountered a problem when installing, see the IBM Installation Manager Information Center. For further information about where to find this information center, see About IBM Installation Manager Information Center.

What to do next

Follow the instructions in "Migrating your WebSphere Application Server configuration to Version 8.5" on page 80 to migrate your Version 6.1 configuration to Version 8.5.

Migrating your WebSphere Application Server configuration to Version 8.5

Use the migration wizard to migrate the listener port, connection factory, and queue from WebSphere Application Server Version 6.1 to Version 8.5 so that you can then view and configure these resources in the Version 8.5 administrative console.

Before you begin

Before starting this task, follow the instructions in "Installing WebSphere Application Server Version 8.5" on page 78 to install WebSphere Application Server Version 7 and the latest fix pack.

About this task

This task covers migration of the stand-alone application server profile that you created in "Installing WebSphere Application Server Version 6.1" on page 26.

> **What else might I do or be interested in?**
> The wizard is the graphical interface to the primary Version 8.5 command-line migration tools, which are the **WASPreUpgrade** and **WASPostUpgrade** commands. You can also use those commands directly to complete a migration.

The migration wizard includes an option to back up your Version 6.1 configuration before migrating to Version 8.5. This enables you to restore the configuration to its previous state after migration.

> **What else might I do or be interested in?**
> Before migrating a WebSphere Application Server Version 5.1.x or Version 6.x application server, you can use the backupConfig command or your own preferred backup utility to back up your existing configuration.

Procedure

1. Log in as root user.
2. If you have not already done so, stop the WebSphere Application Server Version 6.1 application server. In a terminal window, navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and enter `./stopServer.sh server1`.
3. Start the migration wizard. Navigate to `/opt/IBM/WebSphere/AppServer1/bin/ProfileManagement`, and then enter `./wct.sh`.
4. Click the **Migration Sources** tab, and then click **New**.
5. On the "Detected versions of WebSphere Application Server" page, from the list of detected versions of WebSphere Application Server that exist on your computer, select your Version 6.1 installation. For example, `IBM WebSphere Application Server Network Deployment Version 6.1.0.0`.

   Click **Next**.
6. On the "Source profile selection" page, select the source profile that you want to migrate, for example `AppSrv01`, then click **Next**.

7. On the "Profile Migration Output" page, specify the location of the migration backup directory in which you want to place a backup copy of the configuration from the previous version, then click **Next**.

> **What else might I do or be interested in?**
> If the directory does not exist, it is created. If the directory exists, it must be empty because otherwise the backup installation might overwrite existing backup files that are still needed.

8. On the "Target profile selection" page, complete the following steps:

   a. Click **Migrate to an existing profile**, and then select the target profile to which you want to migrate the selected Version 6 source profile, for example AppSrv01.

   > **What else might I do or be interested in?**
   > This page also offers you the option to create a new profile. If you did not create a profile when you installed WebSphere Application Server Version 8.5 you must select **Create new profile** here, then enter the parameters for creating the new profile.

   b. Select **Create a backup of the target profile**.

   > **Why am I doing this?**
   > If necessary you can then use the **restoreConfig** command to restore the configuration after migration.

   c. Click **Next**.

9. On the "Application Migration" page, leave the default option **Migrate and install the applications as part of the migration process** selected.

   > **Why am I doing this?**
   > In this scenario, the sample applications used on Version 6 are to be used on Version 8.5 without any change, so are included as part of the migration.

10. In the **Application installation directory** field, leave the default option **Install the applications in the default directory of the target profile** selected, and then click **Next**.

    > **What else might I do or be interested in?**
    > The other options are to keep the same application installation directories as the previous version or to install the applications into the specified application installation directory. If you select the option to install your applications in the specified application installation directory, you also need to specify where the migrated applications should be located.

11. On the "Port value" page, select the option **Use the port values assigned to the previous (source) profile**, then click **Next**.

12. On the "Additional Migration Options" page, complete the following steps:
    a. In the **Scripting compatibility** field, clear **Migrate object types to support
       script compatibility**.

    b. In the **Administrative console "My Tasks"** field, click **Use the default
       workspace user root directory**.

13. Check the information on the "Migration summary" page, then click **Migrate**
    to start the migration.

14. When the wizard indicates that each stage of the process has been successful, click **Next** to move to the next stage.

    Tip: In the output from the post-upgrade processing, you should see a message that the sample application has been migrated to the Version 8.5 profile: `MIGR0499I: Application sampleMDBv6.ear successfully deployed using the wsadmin command.`

    If any problems occur, correct them and then start the migration process again.
15. On the "Migration Output" page, click **Next**.
16. On the "Migration Results" page, click **Finish**.

Results

You have migrated the listener port, connection factory, and queue from WebSphere Application Server Version 6.1 to Version 8.5, and are ready to check that you can view these resources in the Version 8.5 administrative console.

What to do next

To confirm that the migration of the Version 6.1 resources has been successful, follow the instructions in "Confirming migration of WebSphere Application Server resources to Version 8.5."

Confirming migration of WebSphere Application Server resources to Version 8.5

Using the administrative console, check that the listener port, connection factory, and queue have been successfully migrated from WebSphere Application Server Version 6.1 to Version 8.5.

About this task

To use JMS messaging between WebSphere Application Server Version 8.5 and WebSphere MQ Version 6, you must check that following WebSphere Application Server objects have been successfully migrated from Version 6.1.

* Listener port

Why am I doing this?

Listener ports can be used to enable message-driven beans to be used with WebSphere MQ as a JMS provider to WebSphere Application Server.

What else might I do or be interested in?

In WebSphere Application Server Version 8.5, you can continue to configure EJB 3, EJB 2.1, and EJB 2.0 message-driven beans against a listener port, for example for compatibility with existing message-driven bean applications. However, listener ports are stabilized for WebSphere Application Server Version 8.5, and you should plan to migrate all your message-driven beans to use JCA 1.5-compliant resources. Activation specifications are part of the JCA 1.5 standard, which provides a standard way to integrate JMS providers, such as WebSphere MQ Version 7.5, with Java EE application servers such as WebSphere Application Server.

- Connection factory
- Queue (a type of JMS destination)

These objects are described in "Overview: Initial IT configuration" on page 6.

Procedure

1. Log in as root user.

2. If WebSphere Application Server is not already started, open a terminal window and navigate to /opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin, and then enter ./startServer.sh server1. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

3. Start a web browser, in the address field enter *host_name*:9060/ibm/console. The *host_name* is the name of your WebSphere Application Server host. If the browser is on the same computer as WebSphere Application Server, the value of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when you installed the product. If you did not enable administrative security, the user ID is used only to distinguish changes that you make in the administrative console from changes that are made by other users.

4. Check the listener port settings.

   a. In the administrative console, click **Servers** > **[Server Types] WebSphere application servers** > *server_name* > **[Communications] Messaging** > **Message Listener Service** > **[Additional Properties] Listener Ports**.

   b. Click **PiShopReplyMDBListener**. The Listener Port settings page is displayed.

   c. Check that the following values appear in the required fields:

   Name PiShopReplyMDBListener

   Initial state
   Started

   Connection factory JNDI name
   jms/PiShopReplyCF

   Destination JNDI name
   jms/wasQueue

   d. If everything is set up correctly, click **Cancel**, otherwise make any required changes, then click **OK**.

5. Check the connection factory settings.

   a. In the administrative console, click **Resources** > **JMS** > **Connection factories**.

   b. Click **PiShopReplyCF**. The Connection factories settings page is displayed.

   c. On the Connection factories settings page, under General Properties, check that the following values appear in the required fields:

   Name PiShopReplyCF

   JNDI name
   jms/PiShopReplyCF

   Queue manager
   sampleQM

Hostname
> The host name of the first WebSphere MQ host. If you are using
> WebSphere Application Server and WebSphere MQ on the same
> computer, it is `localhost`.

Port   1414 by default. If you specified a port number during the
WebSphere MQ configuration, it is the number you specified.

Server connection channel
> JMS.CHANNEL

Transport type
> CLIENT

   d. If everything is set up correctly, click **Cancel**, otherwise make any required
   changes, then click **OK**.
6. Check the queue settings.
   a. In the administrative console, click **Resources** > **JMS** > **Queues**. The
   Queues settings page is displayed.
   b. Click **wasQueue**.
   c. Check that the following values appear in the required fields.

   Name `wasQueue`

   JNDI name
   > `jms/wasQueue`

   Base queue name
   > `myQueue`

   d. If everything is set up correctly, click **Cancel**, otherwise make any required
   changes, then click **OK**.
7. If you have made any changes, save these changes to the master configuration,
   then stop and then restart the application server.
   a. Click the **Save** link in the Messages section at the top of the pane to save
   your changes to the master configuration.
   b. Stop the application server. In a terminal, navigate to `/opt/IBM/WebSphere/`
   `AppServer1/profiles/AppSrv01/bin`, and enter `./stopServer.sh server1`.
   c. Start the application server. In the same terminal, enter `./startServer.sh`
   `server1`

Results

You have confirmed that the listener port, connection factory, and queue from your
WebSphere Application Server Version 6.1 installation have successfully migrated
to WebSphere Application Server Version 8.5.

What to do next

Follow the instructions in "Phase 3: Verifying the solution" on page 86 to confirm
that WebSphere Application Server and WebSphere MQ can still communicate with
each other following the migration of WebSphere MQ from Version 6.0 to Version
7.5 and WebSphere Application Server from Version 6.1 to Version 8.5.

Phase 3: Verifying the solution

Verify that WebSphere Application Server and WebSphere MQ can still communicate with each other following the migration of WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5.

Procedure

1. Follow the instructions in "Deploying the Version 8 sample MDB application" on page 59 to deploy the sample MDB that you need to use to complete the final phase of testing.
2. Follow the instructions in "Verifying the migrated IT configuration" on page 62 to complete the final stage of testing the migrated WebSphere Application Server and WebSphere MQ configuration.

Results

Following the migration of WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5, WebSphere Application Server can still connect to WebSphere MQ and messages can be transmitted successfully between the two products.

What to do next

After completing phase 3, you are ready to move on to the final stage of the scenario, which involves replacing the migrated listener port with an activation specification. To complete this final stage, follow the instructions in "Migrating WebSphere Application Server listener ports to activation specifications" on page 91.

Deploying the Version 8 sample MDB application

Using the WebSphere Application Server administrative console, install the Version 8 sample message driven bean (MDB) application so that you can use it to verify that WebSphere Application Server is communicating with WebSphere MQ following the migration of WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5.

Before you begin

If you have not already done so, download the Version 8 sample MDB application. Click the following link and save the file to the machine that hosts WebSphere Application Server: sampleMDB.ear">sampleMDB.earsampleMDB.ear (in online information center).

About this task

The Version 8 sample MDB application, `sampleMDB.ear`, should now be able to use the objects that you migrated from WebSphere Application Server Version 6 to Version 8.5. The MDB application uses these objects to send a message to WebSphere MQ, for receipt by the sample JMS application requester client that you used in "Testing the Version 7 sample JMS application" on page 57.

Procedure

1. Log in as root user.

2. If WebSphere Application Server is not already started, open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and then enter `./startServer.sh server1`. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

3. Start a web browser, in the address field enter *host_name*`:9060/ibm/console`. The *host_name* is the name of your WebSphere Application Server host. If the browser is on the same computer as WebSphere Application Server, the value of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when you installed the product. If you did not enable administrative security, the user ID is used only to distinguish changes that you make in the administrative console from changes that are made by other users.

4. Click **Applications** > **New Application** > **[Install a New Application] New Enterprise Application**.

5. On the first Preparing for the application installation page, select the MDB application that you want to upload and install:

   a. Select **Local file system**, then click **Browse**. The File Upload window opens.

   b. In the File Upload window, browse to the folder where you downloaded the sample MDB application, select `sampleMDB.ear`, then click **Open**.

   c. Click **Next**.

   > **What else might I do or be interested in?**
   > Unless you specify otherwise, the name of the installed application is derived from the name of the MDB EAR file, which in the case of this scenario is `sampleMDB.ear`. If you prefer, you can give the application a different name. To do this, click the **Step 1** link to go to the Select installation options page then specify your chosen **Application name**.

6. On the second Preparing for the application installation page, leave the default setting **Fast Path**, then click **Next**.

7. On the Install New Application page, click the **Step 3** link to go to the Bind listeners for message-driven beans page.

   > **What else might I do or be interested in?**
   > To save having to specify the binding whenever you deploy the MDB EAR file, you can bind the MDB to a listener port name by using an assembly tool. For more information, see the topic Preparing for application installation binding settings in the WebSphere Application Server information center.

8. On the Bind listeners for message-driven beans page specify the listener port name to which the sample MDB application must be bound:

   a. Under **Listener Bindings**, select **Listener port**.

   b. Type the **Name** of the listener port: `PiShopReplyMDBListener`

> **Tip:** This is the listener port that you migrated from WebSphere Application Server Version 6.1 as described in "Migrating your WebSphere Application Server configuration to Version 8.5" on page 46.

   c. Click **Next**.

9. On the Map resource references to resources page, map the connection factory resource reference defined in the sample MDB application to the JNDI name of the connection factory.

   a. Under **Target Resource JNDI Name**, click **Browse** to display a list of resources that are available to be used to bind the resource reference of the MDB.

   b. Select `PiShopReplyCF` then click **Apply**.

> **Tip:** For this scenario, the resource that you select is the connection factory PiShopReplyCF that you migrated from WebSphere Application Server Version 6.1 as described in "Migrating your WebSphere Application Server configuration to Version 8.5" on page 46.

   c. Leave all other fields unchanged and click **Next**.

10. Click **Next** again to go to the Summary page.

11. On the Summary page, click **Finish**. The program displays messages to indicate that the application is being installed. If the application does not install successfully, examine the messages to identify why the installation failed. Correct the problems as needed and try installing the application again.

12. When the application installs successfully, click the **Save** link that is displayed at the end of the messages, to save your changes to the master configuration.

13. Restart the application server.

   a. Stop the application server. In a terminal, navigate to `/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin`, and enter `./stopServer.sh server1`.

   b. Start the application server. In the same terminal, enter `./startServer.sh server1`

Results

You have deployed the MDB application, which is now visible in the Enterprise Applications panel. To view this panel, click **Applications** > **Application Types** > **WebSphere enterprise applications**.

What to do next

Follow the instructions in "Verifying the migrated IT configuration" to run the sample applications and verify that messages can be passed between the two products.

Verifying the migrated IT configuration

Run the sample JMS and message drive bean (MDB) applications to verify that your migrated WebSphere Application Server and WebSphere MQ configurations can communicate with each other.

About this task

In "Testing the Version 7 sample JMS application" on page 72, you ran the supplied JMS sample application to verify that the requester and responder clients of the application could communicate through WebSphere MQ. In this task, you run the requester application as before, but this time the reply comes from the

MDB application that you installed in "Deploying the Version 8 sample MDB application" on page 86, instead of from the responder application, verifying that messages are being passed between WebSphere MQ and WebSphere Application Server.

Procedure

1. If sampleQM is not started, log in as the mqm user and then enter `strmqm sampleQM`.

2. Log in as root user.

3. If WebSphere Application Server is not already started, open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and then enter `./startServer.sh server1`. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

4. Start a web browser, in the address field enter *host_name*`:9061/ibm/console`. The *host_name* is the name of your WebSphere Application Server host. If the browser is on the same computer as WebSphere Application Server, the value of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when you installed the product. If you did not enable administrative security, the user ID is used only to distinguish changes that you make in the administrative console from changes that are made by other users.

5. Check the status of the listener port PiShopReplyMDBListener then, if it is not already started, start it:

   a. In the administrative console, click **Servers** > **Server Types** > **WebSphere application servers** > *server_name* > **[Communications] Messaging** > **Message Listener Service** > **[Additional Properties] Listener Ports**.

   b. Check whether the status of the listener port is started. If listener port is running, the status is shown as a green arrow.

   c. If the listener port is not already started, select it then click **Start**. The status of the listener port changes to a green arrow.

6. Check whether the Version 8 sample MDB application, sampleMDB, is already running then, if it is not running, start it:

   a. Click **Applications** > **Application Types** > **WebSphere enterprise applications** to display a list of installed applications.

   b. Check whether the application status of sampleMDB is started. If sampleMDB is running, the application status is shown as a green arrow.

   c. If the sampleMDB is not already running, start it. Select the check box next to sampleMDB, then click **Start**. The application status changes to a green arrow. The MDB application is now waiting to receive a message.

   If you see an error message similar to the following message, there might be a mismatch between your WebSphere Application Server configuration and the information contained in the sample application. Check the WebSphere Application Server error logs in *install_directory*`/profiles/AppSrv01/logs/server1` for more information.

```
Error sampleMDB failed to start. Check the logs for server server1 on node
    mymachineNode01 for details.
Error An error occurred while starting sampleMDB. Check the logs for
    server server1 on node mymachineNode01 for more information.
```

7. Optional: Stop the Version 6 MDB application, sampleMDBv6, if it is running. Select the check box next to sampleMDBv6, then click **Stop**.

> **Why am I doing this?**

> You do not need to stop the Version 6 MDB application, sampleMDBv6, to be able to run the Version 8 MDB. However, stopping the Version 6 MDB better makes it clear that the message handling is being done by the Version 8 MDB.

8. Log in to the WebSphere MQ host as the `mqm` user.

9. Run the requester client.

   a. On the command line, navigate to the directory that contains the `sampleJMSApp.jar` and the `runreq.sh` files.

   b. Enter `./runreq.sh`.

Results

The sample requester application sends the same request message as before, but the response now comes from the MDB application. The following output, showing sent and received messages, is displayed in the requester application Command Prompt window:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Connection created.

> Sending stock request for 'BakedBeans'
> Sent Message ID=ID:414d512073616d706c65514d202020204fb88351021a0020

> Received Message ID=ID:414d512073616d706c65514d202020204fb88351041c0020 for 'From MDB: BakedBeans - 15 tins in stock. Exp

> Closing connection to QueueManager.
> Closed Connection.
-----------------------
In this window, observe the messages sent through WebSphere MQ.
- The request message sent
- The reply message received
---
When ready, press any key to close the window.
```

If you see errors similar to the following error in the WebSphere Application Server logs, check that the name of the queue in WebSphere Application Server matches the administrative name (not the JNDI name) of the queue in WebSphere MQ:

```
WebSphere MQ call failed with compcode '2' ('MQCC_FAILED') reason '2085'
    ('MQRC_UNKNOWN_OBJECT_NAME')

com.ibm.msg.client.jms.DetailedInvalidDestinationException: JMSWMQ2008:
    Failed to open MQ queue 'myQueue'. JMS attempted to perform an MQOPEN,
    but WebSphere MQ reported an error. Use the linked exception to determine
    the cause of this error. Check that the specified queue and queue manager
    are defined correctly.
```

Note: If you update the WebSphere Application Server configuration, you must restart the application server for the changes to take effect.

You have finished migrating WebSphere Application Server from Version 6 to Version 8.5 and then WebSphere MQ from Version 6 to Version 7, and verified that, following the migration, WebSphere Application Server can still connect to WebSphere MQ for the successful transmission of messages.

You have now completed Option 1 of this scenario.

What to do next

At this point, the Version 8.5 sample MDB application is still using the listener port that you migrated from WebSphere Application Server Version 6 to connect to WebSphere MQ. To complete the delivered logical topology described in "Overview: The delivered logical topology" on page 10, you now need to create an activation specification and configure the Version 8 sample MDB application to use this activation specification instead of the listener port to listen for and respond to messages received from WebSphere MQ. To complete this final stage of the scenario, follow the instructions in "Migrating WebSphere Application Server listener ports to activation specifications."

Migrating WebSphere Application Server listener ports to activation specifications

Create an activation specification for the Version 8 sample MDB application to use, instead of the listener port that you migrated from Version 6, to connect to WebSphere MQ. After creating the activation specification, bind the MDB to the activation specification JNDI name, to replace the previous use of the listener port.

Before you begin

Listener ports are based on an optional, and now deprecated, part of the JMS specification and are stabilized for WebSphere Application Server Version 7 and later. Although you can continue to use listener ports in WebSphere Application Server Version 8.5, if you want to use JCA 1.5-compliant resources, you should plan to migrate your message-driven beans to activation specifications. Activation specifications are part of the JCA 1.5 specification, which provides a standard way to integrate JMS providers, such as WebSphere MQ, with Java EE application servers such as WebSphere Application Server.

For more information, see the topics Message-driven beans, activation specifications, and listener ports and Stabilized features in the WebSphere Application Server Version 8.5 information center.

> ┌─ **What else might I do or be interested in?** ─────────────────
> You should not begin migrating your listener ports until you are sure that the application does not have to work on application servers earlier than WebSphere Application Server Version 7. For example, if you have an application server cluster with some members at Version 6.1 and some at Version 7, you should not migrate applications on that cluster to use activation specifications until after you migrate all the application servers in the cluster to Version 8.5.

EJB 2.0 message-driven beans cannot be configured against JCA 1.5-compliant resources. If you were using an MDB that was an EJB 2.0 application, you would need to upgrade it to EJB 3 or EJB 2.1 before completing this task.

About this task

In this task, the migration from a listener port to an activation specification is achieved by creating a new activation specification to replace the listener port that

you created in "Configuring WebSphere Application Server Version 6.1" on page 28 and then migrated in "Migrating your WebSphere Application Server configuration to Version 8.5" on page 46.

An alternative way of migrating from a listener port to an activation specification is to use the wizard available in WebSphere Application Server. For more information about using this wizard, see the topic Migrating a listener port to an activation specification for use with the WebSphere MQ messaging provider in the WebSphere Application Server Version 8.5 information center.

Procedure

1. Log in as root user.

2. If WebSphere Application Server is not already started, open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and then enter `./startServer.sh server1`. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

3. Start a web browser, in the address field enter *host_name*`:9061/ibm/console`. The *host_name* is the name of your WebSphere Application Server host. If the browser is on the same computer as WebSphere Application Server, the value of *host_name* is localhost.

   Note: You must log on even if you did not enable administrative security when you installed the product. If you did not enable administrative security, the user ID is used only to distinguish changes that you make in the administrative console from changes that are made by other users.

4. In the administrative console, click **Resources** > **JMS** > **Activation specifications**.

5. Ensure that the scope is set to **Cell=***cell_name*, then click **New**.

Why am I doing this?
You use the **Scope** property to set the level at which the activation specification is visible. The cell scope is the highest level, giving the activation specification the greatest visibility.

What else might I do or be interested in?
You can also create activation specifications at other levels. For example, if you have multiple servers you might create an activation specification for each server, using the server scope, so that you can specify different settings to be used for each server.

---

6. Select **WebSphere MQ messaging provider** then click **OK**. The "Create WebSphere MQ JMS resource" wizard starts.

7. On the "Configure basic attributes" page, complete the following fields.

   Name   `PiShopReplyAS`

   The name by which the activation specification is to be known for administrative purposes. This name used to display the activation specification in the administrative console.

JNDI name

> jms/PiShopReplyAS
>
> The JNDI name to be used to reference the object in the JNDI namespace.

Click **Next**.

8. On the "Specify MDB destination data" page type `jms/wasQueue` in the **Destination JNDI name** field, then click **Next**. The destination type is `Queue` by default.

    Tip: This name must match what you specified for the **JNDI name** when you configured the queue in "Deploying the sample MDB application to WebSphere Application Server Version 6.1" on page 34.

    > ┌─ **Why am I doing this?** ─────────────────────────────
    > The sample application uses point-to-point messaging, which is achieved by using a queue destination. Topic destinations are used for publish/subscribe messaging. The JNDI name is used to reference the queue in the JNDI namespace. You use the same JNDI name when you create the queue object.

9. On the "Select connection method" page, leave the default option, **Enter all the required information into this wizard** selected, and then click **Next**.

10. On the "Supply queue connection details" page type the name of the WebSphere MQ queue manager, `sampleQM` in the **Queue manager or queue-sharing group name** field, then click **Next**.

11. On the "Enter connection details" page, enter the following information, then click **Next**.

    Transport mode

    > Leave as the default setting, **Bindings, then client**.

    > ┌─ **Why am I doing this?** ─────────────────────────────
    > In this mode, WebSphere Application Server first attempts to connect in bindings mode, where the WebSphere MQ messaging provider within WebSphere Application Server connects directly to the WebSphere MQ queue manager. If this connection attempt fails, the WebSphere MQ messaging provider then attempts to connect by using client mode, which uses a TCP/IP-based network. Bindings mode usually has better performance, but can be used only when WebSphere Application Server and WebSphere MQ are installed on the same machine. The host name, port, and server connection channel information is required for client mode. Bindings mode requires the configuration of native library path information, which you will add in a later step in this task.

    Host name

    > The host name of the first WebSphere MQ host. If you are using WebSphere Application Server and WebSphere MQ on the same computer, enter `localhost`.

    Port Default value is **1414**, if you specified a port number during the WebSphere MQ configuration, enter this value here.

Server connection channel

Enter `JMS.CHANNEL`.

> **Why am I doing this?**
> Defines the name of the channel a client can use to connect to the queue manager, sampleQM.

12. On the "Test connection" page, click **Test connection**.

> **Why am I doing this?**
> The **Test connection** button verifies that the activation specification contains the necessary details to create a connection to WebSphere MQ.

If successful, the wizard displays the message `A connection was successfully made to WebSphere MQ`. If not successful, check that the queue manager name, port, and server channel that you specified are correct, and match the corresponding values in WebSphere MQ. For example, the following error indicates that the channel name was not found on the queue manager that you specified.

```
A connection could not be made to WebSphere MQ for the following reason:
    JMSCMQ0001: WebSphere MQ call failed with compcode '2' ('MQCC_FAILED')
    reason '2540' ('MQRC_UNKNOWN_CHANNEL_NAME').
```

13. Click **Next**, then on the Summary page click **Finish**.

14. Click the **Save** link in the Messages section of the administrative console page to save your changes to the master configuration.

15. Bind the MDB to the activation specification JNDI name to replace the use of the listener port.

    a. Click **Applications** > **Application Types** > **WebSphere enterprise applications** >

    b. Click **sampleMDB**

    c. Click **[Enterprise Java Bean Properties] Message Driven Bean listener bindings** A table of binding properties is displayed for the EJB module `pishopExampleReplyMDB`.

    d. In the Listener Bindings column, select the `Activation Specification` check box, then specify the JNDI name. In the **Target Resource JNDI Name** field, type `jms/PiShopReplyAS`

       This is the JNDI name of the activation specification that the MDB is to use instead of the listener port. The value must match the JNDI name that you specified in step 7.

       You do not need to set other properties, which are provided to allow you to override values configured on the activation specification. For example, in this scenario the MDB uses the JMS destination, `jms/wasQueue`, that you configured on the activation specification in step 7. (The optional **Destination JNDI name** field overrides the destination configured on the activation specification.)

16. If you have made any changes, save these changes to the master configuration, then stop and then restart the application server.

    a. Click the **Save** link in the Messages section at the top of the pane to save your changes to the master configuration.

b.   Stop the application server. In a terminal, navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and enter `./stopServer.sh server1`.

c.   Start the application server. In the same terminal, enter `./startServer.sh server1`

What to do next

Test that the MDB can still listen for messages, based on the configuration provided by the activation specification instead of the listener port, as described in "Verifying the MDB with an activation specification."

Verifying the MDB with an activation specification

Run the sample JMS and message driven bean (MDB) applications to verify that your migrated WebSphere Application Server and WebSphere MQ configurations can communicate with each other using an activation specification instead of the listener port that was migrated.

Before you begin

Before starting this task, you must have configured the activation specification, and then bound it to the MDB, as described in "Migrating WebSphere Application Server listener ports to activation specifications" on page 91.

About this task

In the final stage of the migration options, you ran the supplied JMS sample application to verify that the requester and responder clients of the application could communicate through WebSphere MQ. The reply came from the Version 7 JMS application that you installed with a binding to the listener port that you had migrated from WebSphere Application Server Version 6.

This task is to test that the MDB still works after you have replaced the listener port with an activation specification.

Procedure

1.   If sampleQM is not started, log in as the mqm user and then enter `strmqm sampleQM`.

2.   Log in as root user.

3.   If WebSphere Application Server is not already started, open a terminal window and navigate to `/opt/IBM/WebSphere/AppServer1/profiles/AppSrv01/bin`, and then enter `./startServer.sh server1`. The following message is displayed:

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process ID is 86479
```

4.   Check the status of the listener port PiShopReplyMDBListener then, if it is already started, stop it:

a.   In the administrative console, click **Servers** > **Server Types** > **WebSphere application servers** > *server_name* > **Communications** > **Messaging** > **Message Listener Service** > **[Additional Properties] Listener Ports**.

b.   Check whether the status of the listener port is started. If listener port is running, the status is shown as a green arrow.

c.   If the listener port is already started, select it then click **Stop**. The status of the listener port changes to a red cross.

5. Check whether the Version 8 sample MDB application, sampleMDB, is already running then, if it is not running, start it:

   a. Click **Applications** > **Application Types** > **WebSphere enterprise applications** to display a list of installed applications.

   b. Check whether the application status of sampleMDB is started. If sampleMDB is running, the application status is shown as a green arrow.

   c. If the sampleMDB is not already running, start it. Select the check box next to sampleMDB, then click **Start**. The application status changes to a green arrow. The MDB application is now waiting to receive a message.

      If you see an error message similar to the following message, there might be a mismatch between your WebSphere Application Server configuration and the information contained in the sample application. Check the WebSphere Application Server error logs in *install_directory*\profiles\AppSrv01\ logs\server1 for more information.

```
Error sampleMDB failed to start. Check the logs for server server1 on node
     mymachineNode01 for details.
Error An error occurred while starting sampleMDB. Check the logs for
     server server1 on node mymachineNode01 for more information.
```

6. Optional: Stop the Version 6 MDB application, sampleMDBv6, if it is running. Select the check box next to sampleMDBv6, then click **Stop**.

Why am I doing this?

You do not need to stop the Version 6 MDB application, sampleMDBv6, to be able to run the Version 8 MDB. However, stopping the Version 6 MDB better makes it clear that the message handling is being done by the Version 8 MDB.

---

7. Log off as root and log in as mqm.

8. Run the requester client.

   a. On the command line, navigate to the directory that contains the sampleJMSApp.jar and the runreq.sh files.

   b. Enter ./runreq.sh.

Results

The sample requester application sends the same request message as before, but the response now comes from the MDB application. This time, the configuration necessary for the MDB to receive messages is provided by the activation specification instead of the listener port.

The following output, showing sent and received messages, is displayed in the requester application Command Prompt window:

```
> Connection factory located in JNDI.
> Destination located in JNDI.
> Creating connection to QueueManager.
> Connection created.

> Sending stock request for 'BakedBeans'
> Sent Message ID=ID:414d512073616d706c65514d202020204fb8835102210020

> Received Message ID=ID:414d512073616d706c65514d202020204fb8835104230020 for 'From MDB: BakedBeans - 15 tins in stock. Exp

> Closing connection to QueueManager.
> Closed Connection.
-----------------------
```

```
In this window, observe the messages sent through WebSphere MQ.
- The request message sent
- The reply message received
---
When ready, press any key to close the window.
```

Troubleshooting:

- If you see errors similar to the following error in the WebSphere Application Server logs, check that the name of the queue in WebSphere Application Server matches the administrative name (not the JNDI name) of the queue in WebSphere MQ:

```
WebSphere MQ call failed with compcode '2' ('MQCC_FAILED') reason '2085'
    ('MQRC_UNKNOWN_OBJECT_NAME')

com.ibm.msg.client.jms.DetailedInvalidDestinationException: JMSWMQ2008:
    Failed to open MQ queue 'myQueue'. JMS attempted to perform an MQOPEN,
    but WebSphere MQ reported an error. Use the linked exception to determine
    the cause of this error. Check that the specified queue and queue manager
    are defined correctly.
```

Note: If you update the WebSphere Application Server configuration, you must restart the application server for the changes to take effect.

- For any other errors, see the Troubleshooting and support section of the WebSphere Application Server product documentation.

Conclusion

You have finished migrating WebSphere Application Server from Version 6 to Version 8.5, and WebSphere MQ from Version 6 to Version 7.5, and verified that, following the migration, the MDB on WebSphere Application Server can use an activation specification to listen for and respond to messages received from WebSphere MQ.

Using either migration option 1 or option 2, and then migrating the listener port to an activation specification, you have achieved the logical topology shown in Figure 8 on page 98.
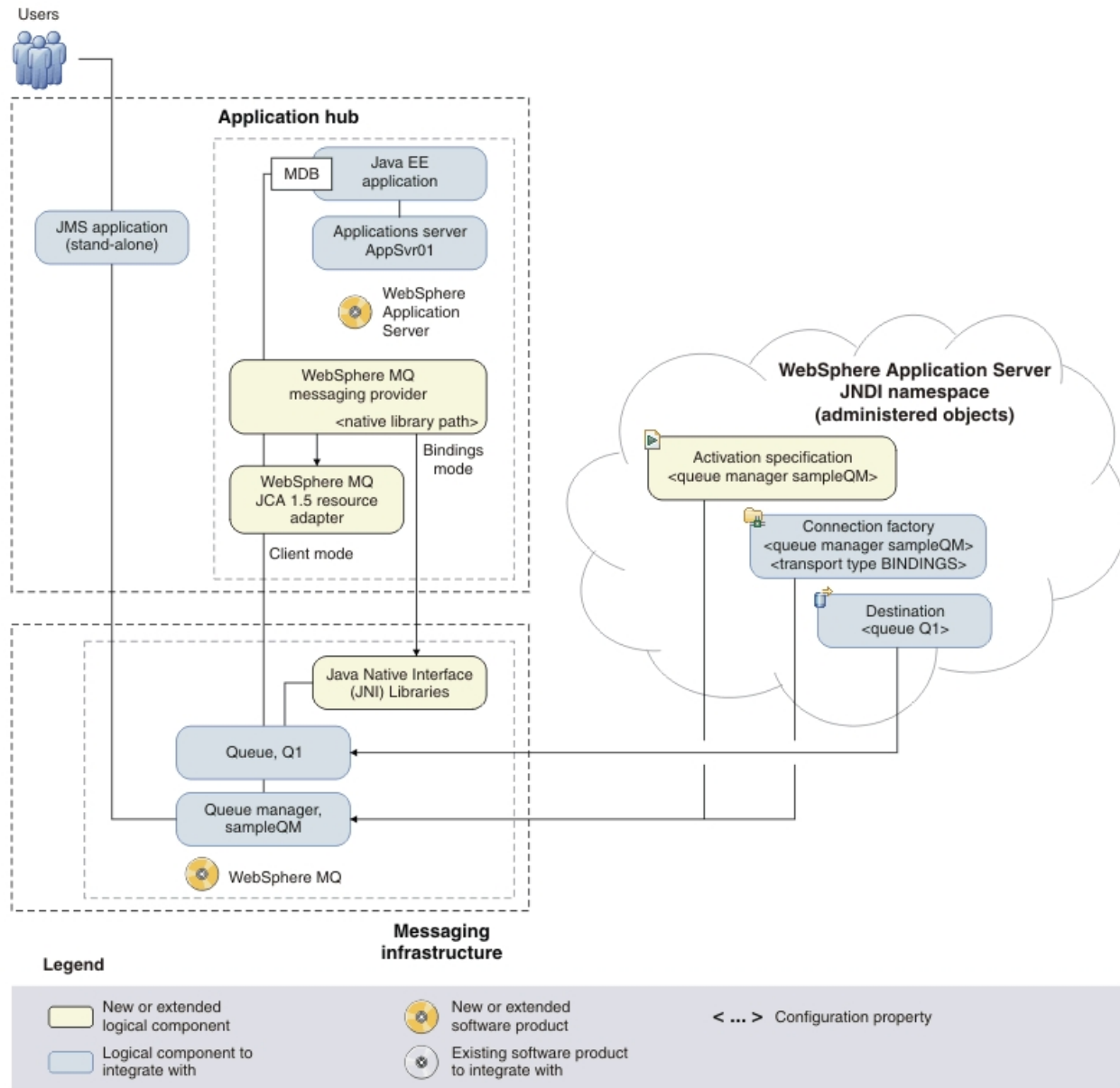
Figure 8. Delivered logical topology. The high-level logical topology diagram for software functions delivered by migrating WebSphere Application Server from Version 6 to Version 8.5 and WebSphere MQ from Version 6 to Version 7.5. This logical topology shows the new functions as new or extended logical components, integrated with the software functions and products from the initial logical topology.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

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

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Software Interoperability Coordinator, Department 49XA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software for use with this program.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Important: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks

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. A current list of IBM trademarks is available on the Web at Copyright and trademark information (www.ibm.com/legal/copytrade.shtml).

Sending your comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or give us any other feedback that you might have.

Use one of the following methods to send us your comments:
- Send an email to ibmkc@us.ibm.com
- Use the form on the web here: www.ibm.com/software/data/rcf/

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

Include the following information:
- Your name and address
- Your email address
- Your telephone or fax number
- The publication title and order number
- The topic and page number related to your comment
- The text of your comment

IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you submit.

Thank you for your participation.

IBM ®

Printed in USA