

IBM Product Connectivity Scenarios and Patterns
Version 1 Release 0

*Scenario: Connecting IBM
MessageSight, WebSphere MQ, and
WebSphere Application Server Liberty
profile getting started*

IBM

Note

Before using this information and the product it supports, read the information in "Notices" on page 33.

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

© **Copyright IBM Corporation 2011.**

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

Contents

Figures	v	Installing WebSphere MQ.	14
Chapter 1. Scenario: Connecting IBM MessageSight, WebSphere MQ, and WebSphere Application Server Liberty profile getting started	1	Configuring the JNDI namespace and administered objects	18
What's new?	2	Configuring the MQTT service with WebSphere MQ	22
Chapter 2. Planning the solution	3	Obtaining the WebSphere MQ Resource adapter	23
Assumptions	3	Installing WebSphere Application Server Liberty profile	24
Business overview	3	Creating a Liberty profile server	26
User roles and interactions.	4	Installing IBM MessageSight for Developers on Oracle VirtualBox 4.3.10	26
Technical solution.	5	Supported browsers	27
Overview: Initial IT configuration	5	Configuring IBM MessageSight.	28
Extending the Initial IT configuration with IBM MessageSight	9	Deploying the sample application to Liberty profile	30
Production physical topology and product mapping	11	Verifying the solution	31
Chapter 3. Implementing the solution	13	Notices	33
Creating a sample initial IT configuration	14	Programming interface information	35
		Trademarks	35
		Sending your comments to IBM	37

Figures

1. User role interactions to deploy the solution	5	4. Production physical topology	12
2. Delivered logical topology	9	5. Delivered logical topology, including	
3. Delivered logical topology	11	MessageSight	14

Chapter 1. Scenario: Connecting IBM MessageSight, WebSphere MQ, and WebSphere Application Server Liberty profile getting started

By connecting to IBM MessageSight, an existing solution involving WebSphere MQ and a Java EE application on WebSphere Application Server Liberty profile can be extended to process a high volume of messages. This scenario leads you through the key tasks required to connect an existing WebSphere MQ and WebSphere Application Server installation to IBM MessageSight and check it works with a sample application.

This scenario was developed using a sample WebSphere MQ installation, and uses sample applications to demonstrate the use of WebSphere Application Server Liberty profile connected to WebSphere MQ and MessageSight. 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 provided with the scenario to verify your progress through each stage.

Notes

This scenario applies to WebSphere® Application Server Version 8 with IBM® MessageSight version 1.0.1.

This scenario was developed and tested with WebSphere Application Server Version 8 and IBM MessageSight version 1.0.1. If you use WebSphere Application Server Version 7, you will notice some differences like using the IBM Installation Manager instead of the installation wizard described in this scenario.

Optional information to help you learn while implementing 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:

- [🔗 Scenarios and Patterns](#)
This scenario in the Scenarios and Patterns information center
- [🔗 Product web page](#)
IBM MessageSight product web page
- [🔗 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

What's new?

This scenario has been updated to add an electronic book (ebook) version of the scenario in EPUB format.

The EPUB file is linked in the “Related resource” list on the scenario welcome page.

You can read EPUB files with many eBook readers; for example, the iBooks app for Apple products, a variety of ebook reader apps for Android devices, and dedicated eBook reader devices like the Barnes & Noble Nook eReader range and the Sony Reader. You can also read ePub files in your web browser with EPUBReader add-on for Firefox and MagicScroll for Chrome. If you want to read an EPUB file on the Amazon Kindle, you can convert EPUB files to kindle format with a converter like Calibre.

Chapter 2. Planning the solution

You can connect IBM MessageSight to WebSphere MQ and WebSphere Application Server Liberty profile in a number of different ways. 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 your system, such as the version of the products that you are using.

This scenario assumes the following points:

- You are using the Windows operating system.
- This scenario was developed by using WebSphere Application Server Liberty profile V8.5. Instructions are provided for obtaining and installing fix packs for WebSphere Application Server Liberty profile.

Note: This scenario applies to the use of WebSphere Application Server Liberty profile Version 8 with IBM MessageSight version 1.1.

- You do not consider full hardening of security for MessageSight. This scenario does not describe security configuration. However, if you do have security configured for MessageSight, you should still be able to complete the scenario.

What else might I do or be interested in?

IBM MessageSight is scalable to deliver large amounts of data to analytics engines and other types of big data application. The HA nature of IBM MessageSight is its ability to withstand software or hardware outages so that it is available as much of the time as possible. See IBM MessageSight Information Center for more information.

Business overview

A company has an existing messaging infrastructure using WebSphere Application Server Liberty profile and WebSphere MQ and now wants to add IBM MessageSight to deal with a large amount of data from outside of the existing system.

A company has an existing messaging infrastructure provided by WebSphere MQ for some existing applications. The company wants to extend this to include communication with devices over the internet using machine to machine (M2M) messaging. The sample in this scenario is a car manufacturer who wants to analyze data from a large number of cars in real time.

In general, the following must be taken into account:

- Volume and cost of data that is transmitted, especially in machine to machine (M2M) messaging with limited data plans
- Power consumption of battery-powered devices

- Responsiveness (near-real time delivery of information)
- Reliable delivery over fragile connections
- Security and privacy
- Scalability

Limited data plans and power consumption are not as important for the car manufacturer since the devices will be attached to specific cars, but might be important considerations for other similar scenarios. The MQ Telemetry Transport (MQTT) protocol is designed to meet these challenges.

Additionally, the company is concerned about the security of implementing this with their existing WebSphere MQ messaging infrastructure. IBM MessageSight is more suitable for this situation since it is not affected by Operating System vulnerabilities and is capable of handling a much higher load. This means the system is more secure and can handle the number of messages being sent by all of its cars.

User roles and interactions

Roles used throughout these scenarios, with interactions between roles for scenario tasks.

Although roles to develop the solution are listed, this scenario focuses on deploying a solution that connects IBM MessageSight to WebSphere MQ and WebSphere Application Server Liberty profile.

Develop the solution

Develop the software aspects of the solution.

Software Architect

The Software Architect is responsible for dividing the required function between the components that make up a software solution. This person works with the specifications and standards used by existing IT systems, and determines where enhancements or new components must be written by the Developer.

Developer

The Developer is responsible for creating and testing the software components and linking them together. In some cases the Developer might need information from the Administrator, for example if the Developer must use an existing queue name in their software code. After the Developer finishes creating the software components, the Developer gives those components to the Administrator for deployment.

Deploy the solution

Deploy the solution for production use, by installing, configuring, and testing the components that provide the solution.

Administrator

The Administrator installs and configures the components that support the solution, in this case MessageSight, WebSphere MQ, and Liberty profile. 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 MessageSight

administrator and WebSphere MQ administrator must share information for the MQ Connectivity feature in MessageSight. The Administrator also deploys the software components given to them by the Developer.

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?

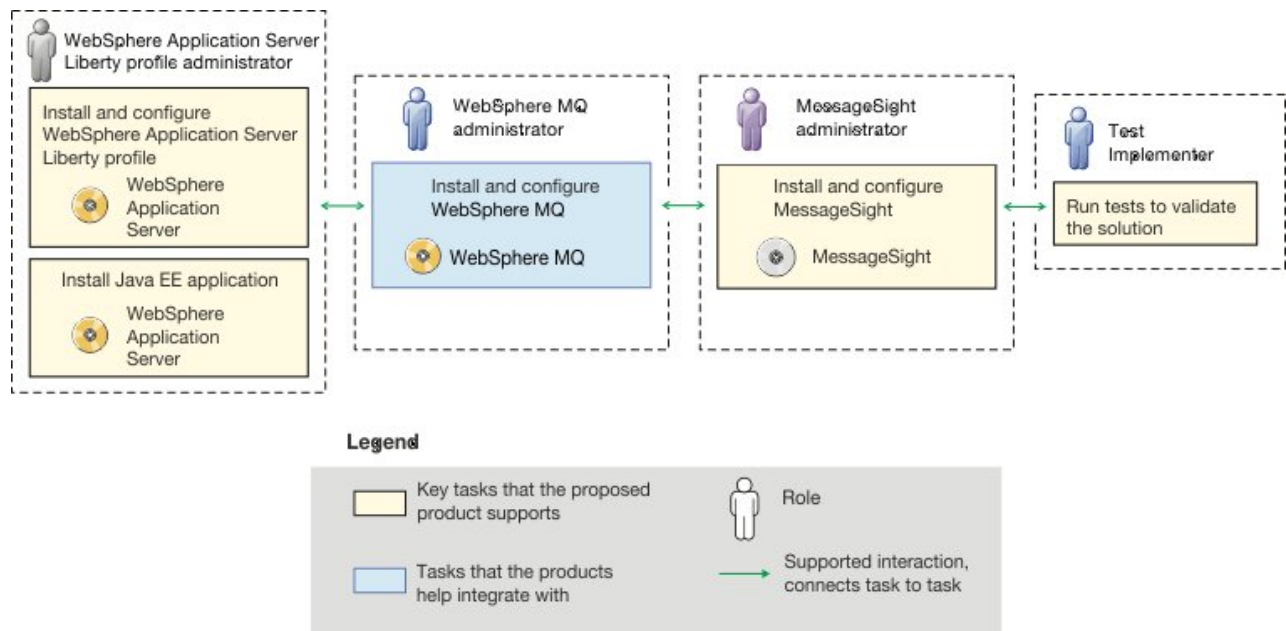


Figure 1. User role interactions to deploy the solution

Technical solution

This scenario describes one way of connecting IBM MessageSight to WebSphere MQ and WebSphere Application Server Liberty profile.

Overview: Initial IT configuration

Company A has an existing IT configuration that uses WebSphere MQ for its messaging infrastructure. This scenario describes adding IBM MessageSight to that initial messaging infrastructure.

The initial IT configuration includes several components that an administrator configures or uses, as shown in Figure 2 on page 9:

JMS application

A stand-alone application that business users interact with, for example to register an order. The application uses the Java™ Message Service (JMS) for asynchronous messaging.

Note: The JMS application will not be provided for verification.

Why am I doing this?

- JMS is the Java EE messaging standard that is widely supported. JMS-based applications are therefore portable across many messaging products.
- JMS provides a level of abstraction from the details of the messaging layer, simplifying the application development process.
- JMS provides asynchronous communication, enabling applications to run without having to wait for a reply, unlike tightly coupled systems such as remote procedure call (RPC).
- Applications that use JMS do not directly specify details to access resources. Instead, they look up and use administered JMS objects such as a connection factory and a destination.

What else might I do or be interested in?

For some situations, other messaging standards might be more suitable than JMS. For example IBM Message Service Clients for C, C++ and .NET, also known as XMS, are APIs that provide similar benefits to JMS for non-Java applications. XMS is therefore more suitable if you are using the .NET platform, or you want to integrate existing C++ applications with newer Java EE applications.

The application uses point-to-point messaging to send messages to a queue in the infrastructure and processes reply messages, to provide the business user with an appropriate response.

Why am I doing this?

In this messaging model, an application sends a message to a queue, and another application receives the message from the queue and acknowledges receipt of the message. This model is the simplest form of messaging because it involves only two endpoints. This model is also the most appropriate for the scenario sample application: a single client requests information from a single server.

What else might I do or be interested in?

In the alternative messaging model, publish/subscribe, a publisher publishes a message to a message topic. Subscribers subscribe to the topic to receive messages. The publisher and subscriber do not have any information about each other, and the message is received by zero or more recipients.

Q1 [Message queue]

The WebSphere MQ queue that the JMS application sends messages to.

Queue manager, sampleQM

The WebSphere MQ queue manager that provides the initial messaging infrastructure. It hosts the queue that the JMS application works with.

Java EE application

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

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 Liberty profile 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 Liberty profile that provides the execution environment for Java EE application programs.

WebSphere Application Server Liberty profile JNDI namespace

WebSphere Application Server Liberty profile includes a name server which provides access to the following JMS administered objects through the Java Naming and Directory Interface (JNDI).

Activation specification

A JMS activation specification can be associated with one or more 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 Liberty profile.

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 Liberty profile Version 7 and later. If you are using WebSphere Application Server Liberty profile for the first time avoid using listener ports.

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.

Destination, myQueue

A JMS destination can be a topic or a queue. In this scenario, the destination is a queue, and identifies the WebSphere MQ queue that applications send messages to, or from which an application receives messages, or both. An application looks up the destination in the JNDI namespace to create a connection to the WebSphere MQ queue.

In this scenario, when creating JMS objects in WebSphere Application Server Liberty profile, 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 Liberty profile 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.

Tip: This scenario was developed by using a sample version of the initial IT configuration described. If you want to try out the scenario, you can either follow the instructions to configure WebSphere Application Server Liberty profile to communicate with your own MessageSight system, or set up a copy of the sample IT configuration.

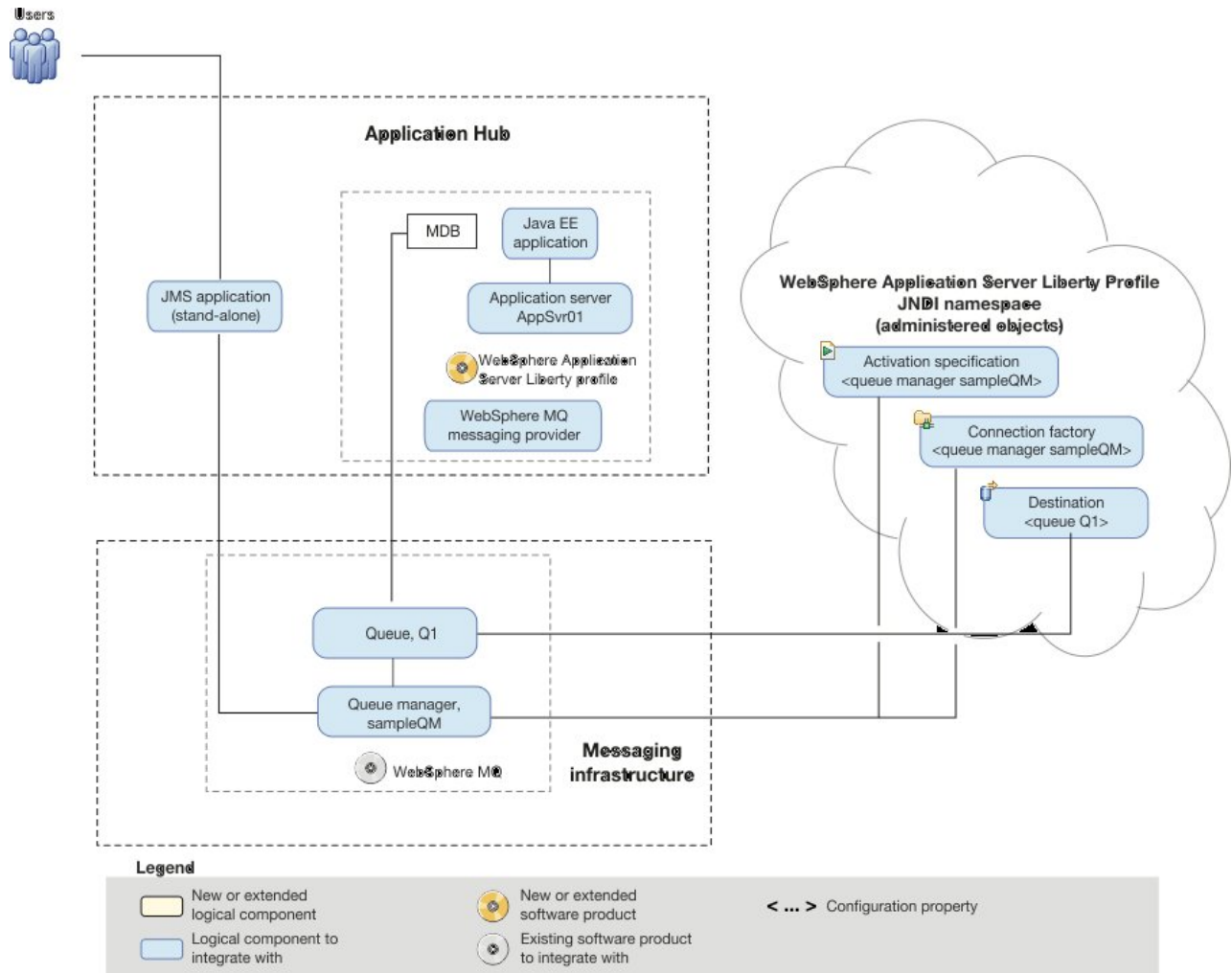


Figure 2. Delivered logical topology. The high-level logical topology diagram for the initial WebSphere MQ messaging infrastructure.

Extending the Initial IT configuration with IBM MessageSight

The company adds a Java EE application on WebSphere Application Server Liberty profile to consume and work with messages from an existing messaging infrastructure provided by WebSphere MQ.

The delivered IT configuration includes all the components in “Overview: Initial IT configuration” on page 5 and IBM MessageSight as a *messaging provider* for WebSphere Application Server Liberty profile. See Figure 3 on page 11.

IBM MessageSight as a *messaging provider* for WebSphere Application Server Liberty profile

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

Why am I doing this?

WebSphere Application Server Liberty profile applications can interact with MessageSight destinations to send and receive messages in the same way as any JMS application in the MessageSight environment.

Java EE application

The application consumes and works with messages from MessageSight. This application runs on an *application server* of the WebSphere Application Server Liberty profile product that has been installed and connected with MessageSight.

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 Liberty profile 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.

You can download the sample applications:

- SampleCarApp.ear
- AutoRemote.war
- SampleCarApp.ear
- AutoRemote.war

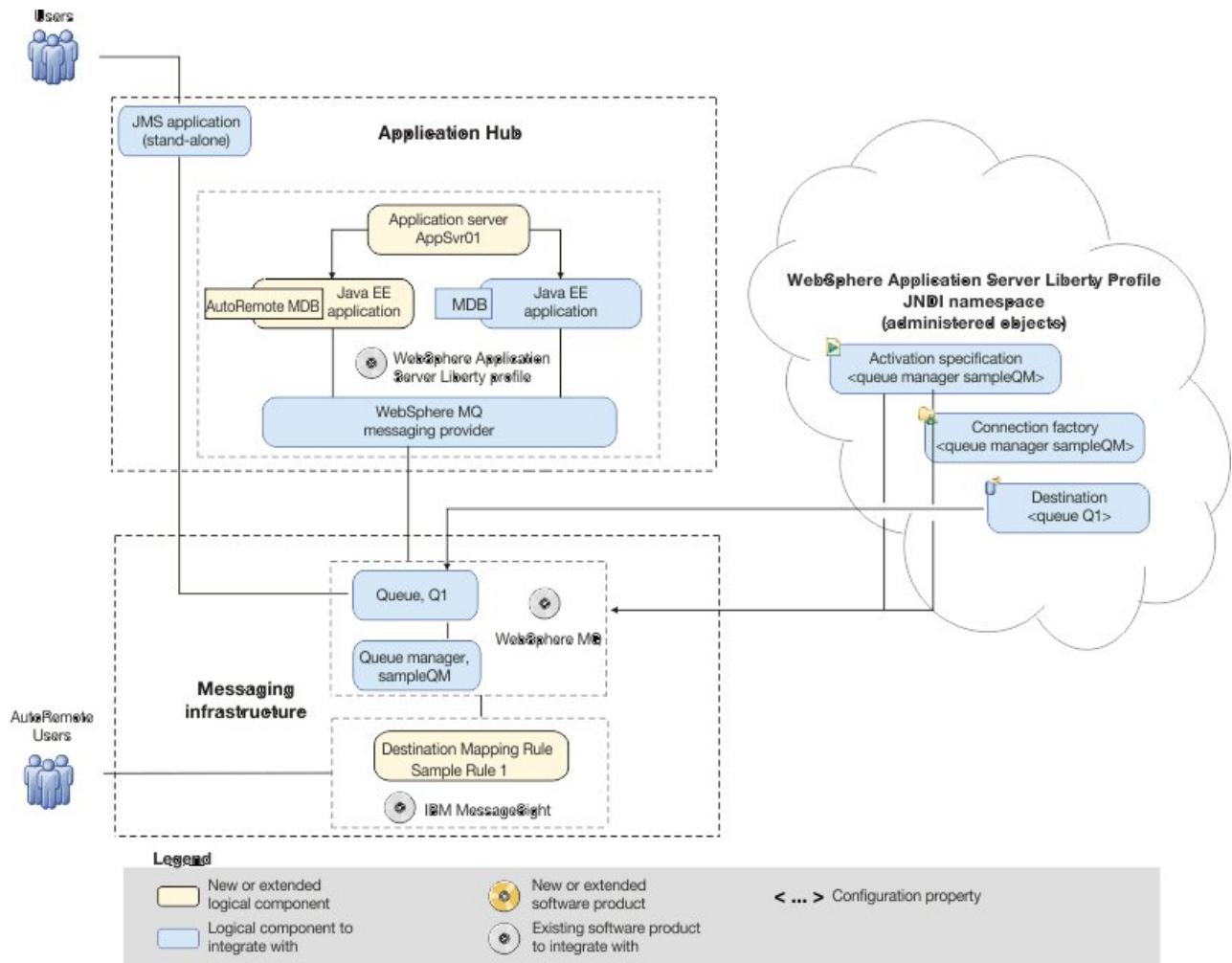


Figure 3. Delivered logical topology. The high-level logical topology diagram for new software functions delivered by integrating IBM MessageSight into the initial WebSphere MQ messaging infrastructure. This delivered logical topology shows the new functions as new or extended logical components, integrated with the software functions and products from the initial logical topology.

Production physical topology and product mapping

The production physical topology specifies the recommended operational environment (machines, operating systems, and software products) for running the applications and services of the solution.

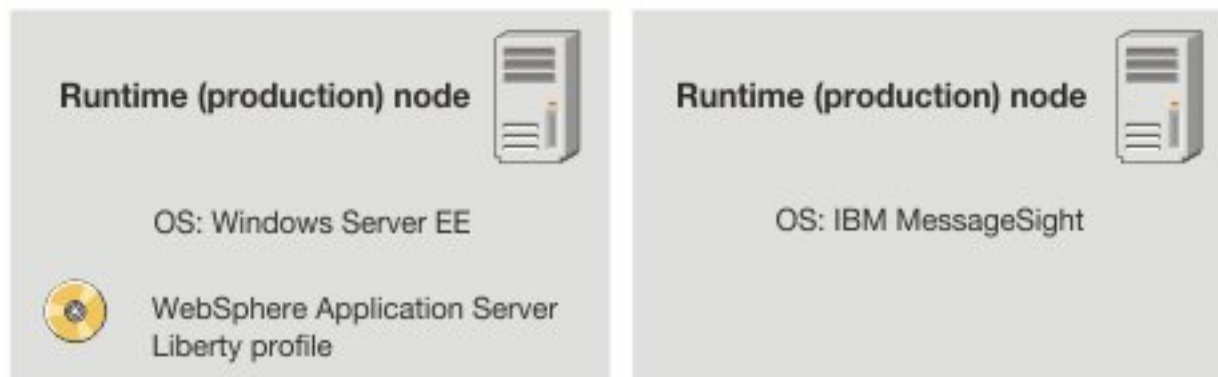


Figure 4. Production physical topology

Node	Operating system and hardware	Software
Host 1 (runtime node)	<ul style="list-style-type: none"> Windows Server 2003 or 2008, Standard or Enterprise Edition. You can use Windows 7, or for 32-bit environments, Windows Vista or Windows XP Professional, but these operating systems are not supported for production use of WebSphere Application Server. Intel EM64T, AMD Opteron, or for 32-bit environments, Intel Pentium processor at 2.33 GHz or faster 1 GB physical memory 3 GB free disk space 	<ul style="list-style-type: none"> WebSphere Application Server Liberty profile IBM MessageSight Prerequisite software: <ul style="list-style-type: none"> A web browser for use by the Launchpad, Windows Explorer, and WebSphere Application Server Liberty profile administrative console: Microsoft Internet Explorer Version 6.0, or Mozilla-based browsers Version 1.7 or higher

Chapter 3. Implementing the solution

Implementing the solution in this scenario involves connecting IBM MessageSight to WebSphere Application Server Liberty profile and WebSphere MQ, which provided the initial messaging infrastructure for a company.

Before you begin

The starting point for this scenario is an existing, verified, Liberty profile and WebSphere MQ installation as the initial messaging infrastructure.

This scenario was developed using a sample installation, and uses sample applications to demonstrate connecting to MessageSight. 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 provided with the scenario to verify your progress through each stage.

About this task

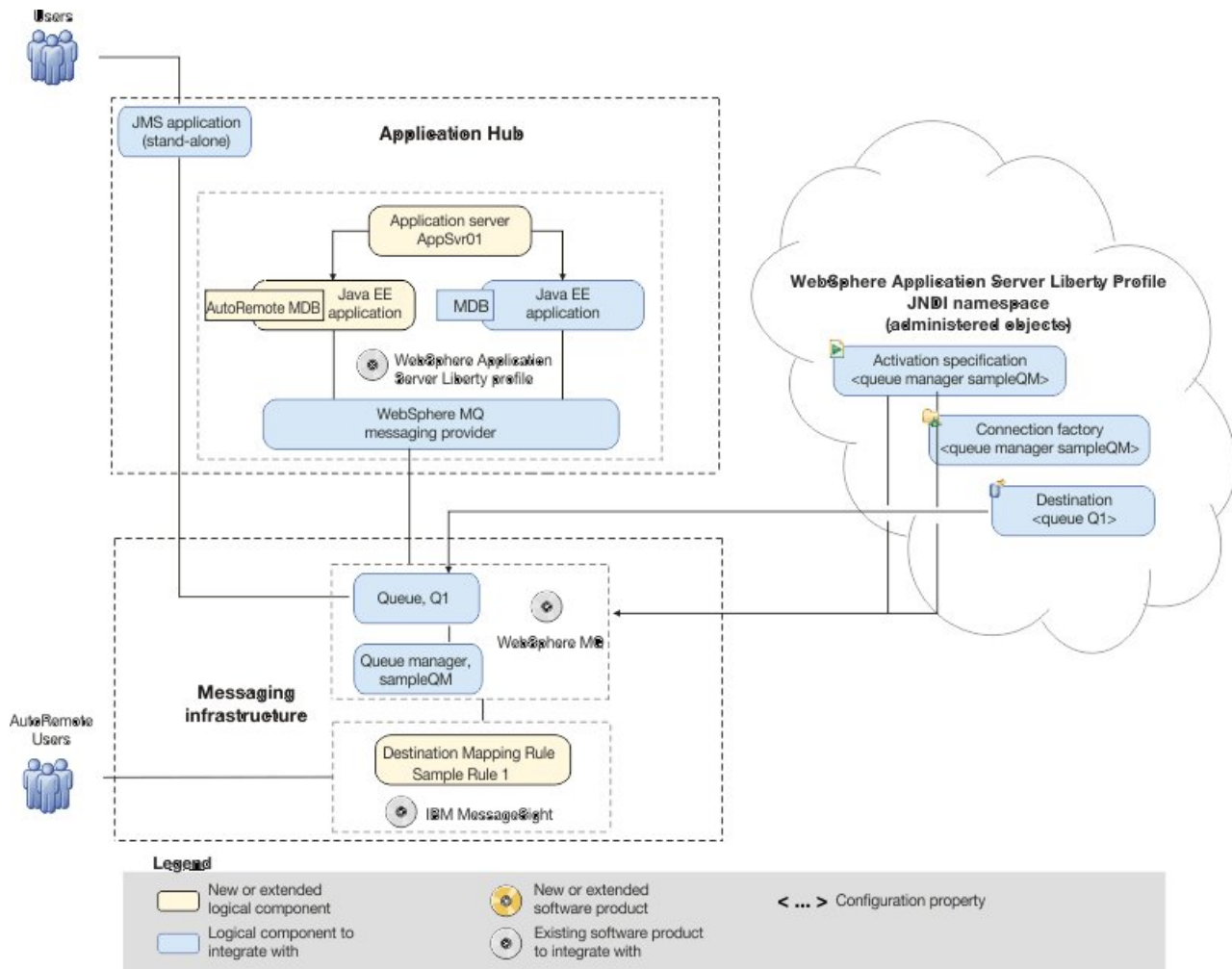


Figure 5. Delivered logical topology, including MessageSight. The high-level logical topology diagram for new software functions delivered by integrating MessageSight into the initial messaging infrastructure. This delivered logical topology shows the new functions as new or extended logical components, integrated with the software functions and products from the initial logical topology.

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

Creating a sample initial IT configuration

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

Installing WebSphere MQ

To complete this scenario you must have a functioning WebSphere MQ messaging network. This task provides basic instructions to install WebSphere MQ if you do not already have it installed on your system.

Before you begin

- You must have local administrator authority when you are installing. Define this authority through the Windows facilities.
- Ensure that the machine name does not contain any spaces.
- Ensure that you have sufficient disk space, up to 600 MB, to fully install WebSphere MQ for Windows.
- Determine whether any WebSphere MQ users are defined on Windows 2000 or later domain controllers.

Why am I doing this?

WebSphere MQ checks that only authorized users can access queue managers or queues. When a user attempts such access, WebSphere MQ uses its own local account to query information about the user. However, domain controllers running on Windows 2000 or later can be configured such that WebSphere MQ cannot use local accounts to make these queries. In this situation, you must provide WebSphere MQ with a special account to use. This account is necessary when both of the following conditions apply:

- Any domain controller on your network is running on Windows 2000 or later
- Local user accounts are not authorized to query the group membership of the domain user accounts

If you require a special account, refer your domain administrator to the instructions in Creating and setting up domain accounts for WebSphere MQ in the WebSphere MQ information center.

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

About this task

This task describes the basic steps to install WebSphere MQ on the Windows operating system.

The installation programs contain links to further information if you require it during the installation process. The installation process has four parts:

1. Use the Launchpad to check and install software requirements, specify network information, and start the WebSphere MQ installation wizard.
2. Use the WebSphere MQ installation wizard to install the software, and start the Prepare WebSphere MQ wizard.
3. Use the Prepare MQ wizard to start the WebSphere MQ service and start the Default Configuration wizard.
4. Optionally, use the Default Configuration wizard to create objects for verifying the installation using the Postcard application supplied with WebSphere MQ.

Procedure

1. Insert the WebSphere MQ for Windows Server DVD into the DVD drive. If autorun is enabled, the installation process starts. Otherwise, double-click the Setup icon in the root folder of the DVD to start the installation process.
2. Use the buttons on the left of the Launchpad to review and, if necessary, modify the software requirements and network configuration.

WebSphere Eclipse Platform 3.3 is required for using the WebSphere MQ Explorer and Managed File Transfer Application components. You can use the launchpad to install this software if required: after you choose whether to install from a DVD or your file system, the Launchpad opens a dialog box with the Eclipse Setup.exe program selected. Click **Open** to run the WebSphere Studio IES 3.3 installation wizard. After installing, click **Refresh** in the Launchpad to update the status of the software requirements.

3. On the WebSphere MQ Installation page of the Launchpad, Select installation language and click **Launch IBM WebSphere MQ Installer** to start the IBM WebSphere MQ - InstallShield Wizard.
4. On the setup wizard, select the **Custom** setup type and click **Next**.
5. On the Installation Details page, specify the **Installation Name** and the **Installation Description** field and click **Next**.
6. Optional: You can click the **Change** button to change the Installation folder for program files. Click **Next**.
7. Select folder for log files and click **Next**.
8. On the Features page, click the icon and select **Install this feature** for the following features:
 - Server
 - Telemetry Service
 - MQ Explorer
 - Telemetry Clients
 - Development Toolkit
 - Java and .NET Messaging and Web Services

Click **Next** to go to the Ready to Install WebSphere MQ page.

9. On the Ready to Install WebSphere MQ page, click **Install** and go to the Installing WebSphere MQ page. It takes several minutes to install the WebSphere MQ. At the end of the process, the WebSphere MQ Setup window displays the message Installation Wizard Completed Successfully. Click **Finish**. The Prepare WebSphere MQ wizard starts automatically.
10. Follow the instructions in the Prepare WebSphere MQ wizard.

Why am I doing this?

The Prepare WebSphere MQ wizard helps you to configure WebSphere MQ files and a user account for your network. You must run the wizard to configure the WebSphere MQ Service before you can start any queue managers.

- a. On the Welcome to the Prepare WebSphere MQ wizard page of the wizard, click **Next**. The wizard performs some configuration and displays status messages, then displays the WebSphere MQ Network Configuration page. The contents of this page vary according to your use of domain controllers.
 - b. Follow the instructions for WebSphere MQ Network Configuration page and then click **Next** to continue through the wizard until the wizard displays the Completing the Prepare WebSphere MQ Wizard page.
11. Optional: If you want to use the WebSphere MQ Postcard application to verify your installation, click the **Set up the Default Configuration** link to display the WebSphere MQ Default Configuration window.

Why am I doing this?

The default configuration creates objects, such as a queue manager called *QM_machine_name*, that you can use to verify the success of your installation by using the Postcard application that is supplied as part of WebSphere MQ. If you do not create the default configuration, you can still use the Postcard application later on, after you have created your own objects in WebSphere MQ.

Note: You cannot create the default configuration if you have already created other queue managers; you must first delete the other queue managers then run the Default Configuration wizard.

- a. The WebSphere MQ Default Configuration window shows an initial summary of the configuration you are about to create. This window remains on screen while you proceed through the pages of the Default Configuration wizard. Click the **Set up Default Configuration...** button to start the Default Configuration wizard.
- b. Click **Next** to move through the information pages.
- c. On the Select Options page, clear both the **Allow remote administration of the queue manager** and **Join the queue manager to the default cluster** check boxes, then click **Next**.

Why am I doing this?

This scenario assumes that you have one machine on which you are installing WebSphere Application Server and WebSphere MQ. You therefore do not need to administer the queue from another machine. Clustering is not described in this scenario.

- d. On the summary page, click **Finish**. A dialog box is displayed while the default configuration is created.

What else might I do or be interested in?

You can use this window after installation to change the default configuration. To display this window after installation, use the Windows **Start** menu to start the Prepare WebSphere MQ wizard, then follow the steps.

The existing WebSphere MQ Default Configuration window is updated with the results of the configuration.

- e. Click **Close** to close the Default Configuration wizard.
12. Leave the **Launch WebSphere MQ Explorer** check box selected, choose whether to start Notepad to view the release notes, then click **Finish**. The WebSphere MQ Explorer starts.
13. If you created a default configuration, verify your installation by using the Postcard application that is supplied with WebSphere MQ.
 - a. If the Welcome page is not already displayed, click **Help > Welcome** to display it.
 - b. Click the **Get Started** icon.

- c. Click the **Launch the Postcard Application** icon to display the Postcard - Sign On dialog box.
- d. Click **Help** in this and other Postcard application dialog boxes to view instructions about running the Postcard application.

If you did not create a default configuration, you can still follow these steps to verify your installation, but you must first configure a queue manager as described in the next task of this scenario.

Note: Running the Postcard application on a non-default configuration automatically creates a queue called postcard on your queue manager. You can delete this queue after using the postcard application.

Results

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

If there were problems during installation, use the .log files in the C:\Documents and Settings\userID\Local Settings\Temp directory, and the amqmsccw.txt and amqmjpsc.txt files in the installation directory, to investigate. The default installation directory is C:\Program Files\IBM\WebSphere.

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

Define to WebSphere MQ an initial context for the JNDI namespace, then define in the namespace the administered objects that the sample application can use.

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:

- 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 MQ destinations with which to send or receive messages.

Procedure

1. Start WebSphere MQ Explorer, if it is not already started, by clicking **Start > All Programs > IBM WebSphere MQ > WebSphere MQ Explorer**. If WebSphere MQ Explorer is running and displaying the Welcome page, close the Welcome page to start administering WebSphere MQ objects.
2. Create a queue manager to use for the sample application.
 - a. Right-click **Queue Managers** and select **New > Queue Manager**. The Create Queue Manager wizard starts.
 - b. In the **Queue manager name** field type `sampleQM`.

What else might I do or be interested in?

You can choose a different name for the queue manager, but you must remember to use it in later configuration steps, in place of `sampleQM`.

Note: The name must have no more 48 characters, from the following set:

- Uppercase or lowercase A-Z
- Numerics 0-9
- Period (.)
- Forward slash (/)
- Underscore (_)
- Percent sign (%)

Names are case-sensitive. Objects of the same type must have different names. For example two queues cannot have the same name, but a queue manager and a queue can.

- c. In the **Dead-letter queue** field type `SYSTEM.DEAD.LETTER.QUEUE`. This field is the name of the dead-letter queue that is automatically created when you create the queue manager.

Why am I doing this?

A dead-letter queue stores messages that cannot be delivered to their correct destination, for example because the queue is full. All queue managers should have an associated dead-letter queue.

- d. Leave the other fields as defaults and click **Finish**. If the **Finish** button is disabled, click **Next**. The **Finish** button is disabled if the port number conflicts with an existing queue manager, for example, the queue manager that is created as part of the default configuration. You must continue through the wizard to change the default port number.
- e. If you clicked **Next**, continue to accept the defaults and click **Next** on each page until you get to the final page of the wizard, when the **Finish** button becomes available. Change the specified port number, for example to 1415, and click **Finish**.

WebSphere MQ displays a dialog window while the queue manager is created and started.

3. Add an initial context for the *JNDI namespace* then connect WebSphere MQ Explorer to that context.

Why am I doing this?

Before you can use WebSphere MQ Explorer to create and configure JMS administered objects, you must add an initial context to define the root of the JNDI namespace in which the administered objects are stored. Whenever you want to use WebSphere MQ Explorer to create or manage administered objects in the JNDI namespace, you must connect the MQ Explorer to the initial context of the JNDI namespace.

- a. In the MQ Explorer - Navigator pane, right-click **JMS Administered Objects**, then select **Add Initial Context**. This action displays the "Connection details" page.
- b. Under "Where is the JNDI namespace located?", select the **File system** check box.
- c. In the **Bindings directory** field, type C:\JNDI-Directory.

Why am I doing this?

This value matches the JNDI namespace location specified in the sample JMS application. If you want to specify a different JNDI directory, you must modify the application to match.

If the directory does not exist on your system the window displays the message Specified location does not exist or is not readable. Click **Browse** to open a file system window, navigate to Local Disk (C:), then click **Make New Folder** to create the JNDI-Directory folder. Click **OK**. Click **Next**.

- d. On the User preferences page, leave the default settings.

Why am I doing this?

- **Context nickname:** The location of the JNDI namespace will be used as the nickname to display the initial context in WebSphere MQ Explorer.
- **Connect immediately on finish:** This option connects WebSphere MQ Explorer to the JNDI namespace when you finish creating the initial context, so that you can create administered objects immediately.
- **Automatically reconnect to context on startup:** This option is not selected, because usually you do not need WebSphere MQ Explorer to automatically reconnect to the initial context every time that you close and reopen WebSphere MQ Explorer.

What else might I do or be interested in?

If you routinely use WebSphere MQ Explorer to create or manage administered objects in the JNDI namespace, you can select the **Automatically reconnect to context on startup** check box to cause WebSphere MQ Explorer to automatically reconnect to the initial context whenever WebSphere MQ Explorer is started. This option saves you having to manually connect WebSphere MQ Explorer to the initial context.

Click **Finish** to create and display the initial context.

4. Create a connection factory administered object.

Why am I doing this?

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

- a. In the MQ Explorer - Navigator pane, expand **JMS Administered Objects**, then expand the initial context, labeled **file:/C:/JNDI-Directory/**.
- b. Right-click **Connection Factories**, then select **New > Connection Factory**. This action displays the New Connection Factory wizard.
- c. In the Name field, type **myCF**

Why am I doing this?

The sample JMS application contains code that looks up a connection factory with the name **myCF**. If you must use a different name, you must modify the application to match. WebSphere MQ is used for the messaging provider, because the sample application uses point-to-point messaging

Click **Next**.

- d. Leave the type of connection factory as **Connection Factory**, because this option is the most flexible for general JMS use.

Why am I doing this?

A domain-independent connection factory enables JMS applications to use both point-to-point messaging and publish/subscribe messaging, especially if you want the JMS application to perform both types of messaging under the same transaction.

What else might I do or be interested in?

If a JMS application is intended to use only point-to-point messaging or only publish/subscribe messaging, you can select the specific messaging domain when you create the connection factory and a domain-specific (queue or topic) connection factory is created.

- e. Leave the support for XA transactions as cleared.

Why am I doing this?

The sample application does not use XA-compliant transactions.

What else might I do or be interested in?

WebSphere MQ JMS supports XA-compliant transactions in bindings mode. If you want the sample application to use XA-compliant transactions, you must modify the sample application.

Click **Next**.

- f. Leave the transport as **Bindings**.

Why am I doing this?

The sample JMS application that uses the connection factory runs on the same computer as the queue manager, so can use Bindings mode transport. This option means that the JMS application connects directly to the queue manager, and offers a performance advantage over the alternative client mode.

Click **Next**, then **Next** again.

- g. On the Change properties page, select **Connection** from the menu on the left, then in the Connection pane select sampleQM as the **Base queue manager**.

Why am I doing this?

The base queue manager is the queue manager that the application will connect to. Leave this value blank if you want the application to be able to connect to more than one queue manager.

- h. Click **Finish**. WebSphere MQ displays a dialog window to show that the object was created successfully. Click **OK** to close the dialog window.

Results

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

Configuring the MQTT service with WebSphere MQ

The MQTT service for WebSphere MQ can be installed and ready to use with the following steps.

Before you begin

- Set up and configure the sample IT configuration from Creating a sample initial IT configuration on your system.
- You must have local administrator authority when you are installing. Define this authority through the Windows facilities.

About this task

This task modifies an existing WebSphere MQ installation and creates an MQTT service to listen on port 1883. The configuration gives everyone permission to publish and subscribe to any topic.

Procedure

- Optional: If you do not install **Telemetry Service** and **Telemetry Clients** when you install WebSphere MQ in *Installing WebSphere MQ*, you can modify the existing installation to include the MQTT service.
 - Run the same WebSphere MQ installer as in the initial sample IT configuration. This is the IBM WebSphere MQ InstallShield Wizard. For more information, see steps 1-3 in *Installing WebSphere MQ*.
 - In the installer, select **Maintain or upgrade an existing instance**.
 - Select your existing installation and click **Next**.
 - Leave **Modify** selected and click **Next**.
 - Click the icons for **Telemetry Service** and **Telemetry Clients**, and select **Install this feature** to install each feature.
 - Click **Next** and then click **Modify**. The installer will stop any WebSphere MQ activity if it is running.
- Open WebSphere MQ Explorer.
- Run the Telemetry “Define sample configuration” wizard.
 - Optional: If the “MQ Explorer - Content” window is closed, click **Window > Reset Perspective**.
 - Expand sampleQM and select the **Telemetry** folder.
 - Click **Define sample configuration** to start the wizard.
 - Click **Finish** to create the telemetry service and run the MQTT Client Utility.
- Test the connection in the MQTT Client Utility window.
 - In the Connection section, click **Connect**.
 - In the Subscription section, click **Subscribe**.
 - In the Publication section, click **Publish**. In the Client History section, you can see that a message has been published and received.

Results

You have installed the MQTT service.

Obtaining the WebSphere MQ Resource adapter

To connect to WebSphere MQ from the Liberty profile, the WebSphere MQ Resource adapter must be used. The Liberty profile does not contain the WebSphere MQ Resource adapter. You must download it separately.

Procedure

- Download the WebSphere MQ Resource adapter from Fix Central. The file download has a file name in the format *V.R.M.F-WS-MQ-Java-Instal1RA.jar*
- Install the WebSphere MQ Resource adapter.
 - In a command prompt, navigate to the folder where WebSphere MQ Resource adapter is located, and then enter the command: `java -jar <V.R.M.F>-WS-MQ-Java-Instal1RA.jar` The following message is displayed:

LICENSE INFORMATION

The Programs listed below are licensed under the following terms and conditions in addition to those of the IBM International License Agreement for Evaluation of Programs (IBM form number Z125-5543-05).

Program Name: IBM WebSphere MQ V7.5 - 06/2013
Before you can use, extract, or install IBM WebSphere MQ V7.5 - 06/2013, you must accept the terms of IMPORTANT: READ CAREFULLY and additional license information. Please read the following license agreements carefully.

The license agreement is separately viewable using the --viewLicenseAgreement option.

Press Enter to display the license terms now, or 'x' to skip.

b. Press Enter to display the license terms. This message is displayed:

By choosing the "I Agree" option below, you agree to the terms of the license agreement and non-IBM terms, if applicable. If you do not agree, select "I do not Agree".

Select [1] I Agree, or [2] I do not Agree:

c. Press 1 to agree to the license agreement. The following message is displayed:

Enter directory for product files or leave blank to accept the default value.
The default target directory is H:\Liberty\WMQ

Target directory for product files?

d. Enter the directory to install the files to, or press Enter without entering any value to install the default location as displayed. After the files are installed to the requested location, a confirmation message is displayed:

Extracting files to H:\Liberty\WMQ\wmq
Successfully extracted all product files.

Results

Within the selected directory, a new directory that is called wmq is created. Within this directory, the following files are installed:

```
jboss-jmsra-ds.xml
WAS_CE_jmsra_deployment_plan.xml
WAS_CE_jmsra_ivt_deployment_plan.xml
wmq.jmsra.ivt.ear
wmq.jmsra.rar
```

What to do next

Follow the instructions in "Installing WebSphere Application Server Liberty profile" to install and configure WebSphere Application Server Liberty profile.

Installing WebSphere Application Server Liberty profile

Install WebSphere Application Server Liberty profile by extracting the distribution image. At the end of this task, you are ready to configure WebSphere Application Server Liberty profile to communicate with WebSphere MQ.

Before you begin

Ensure that your system meets the operating system and Java requirements for using the Liberty profile. See <http://www.ibm.com/support/docview.wss?rs=180&uid=swg27028175>.

About this task

Install WebSphere Application Server by extracting the archive file. The distribution image is packaged as a self-extracting archive file. Running the self-extracting archive file installs the Liberty profile so that you are then ready to create a server.

Procedure

1. Download and install the Liberty profile in a non-Eclipse environment:
For the no-charge developer edition (with no IBM support), you can download the archive file from the WASdev community download page. Download the archive file of the WebSphere Application Server Liberty profile run time. The archive file name is in the form of `wlp-edition-runtime-version.jar`.
 - a. Open a command prompt and navigate to the archive file of the Liberty profile.
 - b. Type `java -jar filename.jar` on the command prompt and press **Enter**, where the *filename* is the name of the archive file that you downloaded from the WASdev community downloads page.
 - c. Press **Enter** to display the license terms now, or 'x' to skip.
 - d. Press **Enter** to display more license information now, or 'x' to skip.
 - e. Select 1 to agree the terms of the license agreement and non-IBM terms.
 - f. Type the target directory for product files. Press **Enter** to use the current folder.
 - g. If the following message is displayed, the Liberty file is installed.

```
Extracting files to C:\Program Files\IBM\WebSphere\wlp
Successfully extracted all product files.
```

What else might I do or be interested in?

- Other Liberty profile editions in a non-Eclipse environment:
 - For all editions other than no-charge developer edition, you can use the archive file that is included with each edition of WebSphere Application Server version 8.5.5. For detailed installation instructions, see Installing Liberty in IBM Knowledge Center.
 - You can also download an edition-specific Liberty profile archive file, including the developer edition with IBM support, from Passport Advantage online.
- Liberty profile edition in an Eclipse environment: You can also download the Liberty profile in an Eclipse environment. To learn more about installing the WebSphere Application Server Developer Tools for Eclipse and using the tools to install the Liberty profile see, Download Liberty profile in Eclipse.

2. Optional: Set the **JAVA_HOME** property for your environment. You can use the following command to set the **JAVA_HOME** property, and to add the Java /bin directory to the path:

```
set JAVA_HOME=C:\Program Files\Java\jdk-11.0.2
set PATH=%JAVA_HOME%\bin;%PATH%
```

The Liberty profile runtime environment searches for the **java** command in this order; **JAVA_HOME** property, **JRE_HOME** property, and system **PATH** property.

What to do next

Follow the instructions in “Creating a Liberty profile server” to create a Liberty profile server to communicate with WebSphere MQ.

Creating a Liberty profile server

Create a Liberty profile server in the command prompt.

Procedure

1. Open a command prompt and navigate to the `wlp\bin` in the directory where the distribution image was extracted.
2. Create a server that is called `SCENARIO0`. In the command prompt, type `server create SCENARIO0`.

Results

The server is created.

Installing IBM MessageSight for Developers on Oracle VirtualBox 4.3.10

IBM MessageSight for Developers is an open virtual appliance (OVA) package that you can import into your virtualization environment. You can install IBM MessageSight for Developers on Oracle VirtualBox 4.3.10 according to the following procedure.

Before you begin

Minimum system requirements

The host system must have these minimum system requirements:

- A 2-core 64-bit processor with VT-x/AMD-V support
- 16 GB of available space on a hard disk
- 4 GB RAM, with a minimum of 2 GB available for the virtual appliance

By default, the OVA is configured to run with 4 GB RAM. You can change this value to a lower setting.

Supported browsers

To use IBM MessageSight for Developers, you need access to one of the supported browsers. For more information about the supported browsers, see “Supported browsers” on page 27.

Procedure

1. Click **Oracle VirtualBox** on your computer and open it.
2. In Oracle VirtualBox, import the IBM MessageSight for Developers virtual image OVA package.

- a. Click **File > Import Appliance**.
 - b. Click the Import Virtual Application icon and navigate to the IBM MessageSight OVA file on your computer.
 - c. Click on the file name, and then click **Open**.
 - d. Click **Next** and then **Import**. The IBMMessageSightV1.1.ova file is now importing.
3. After the IBMMessageSightV1.1.ova file is imported, click **Start**. The IBM MessageSight for Developers image is now loading.
 4. After the IBM MessageSight for Developers image is loaded, enter the default user ID: admin and the default password: admin at the login prompt.
 5. For Which interface would you like to configure (default is eth0), enter eth0, and then press Enter.
 6. For Use DHCP for the Ethernet port (yes/no: default no), enter Yes , and then press Enter.
 7. The URL to connect to IBM MessageSight UI is now displayed on the screen. Open a supported browser, enter the IP address that is displayed on the screen to connect to the IBM MessageSight Web UI.
 8. Click **Add Exception** and then **Confirm Security Exception** to add the IBM MessageSight Web UI as an exception.
 9. In the MessageSight Login page, enter the default user ID: admin and the default password admin, and click **Log in**. The License Agreement page is displayed.
 10. Click **I Agree** and the License Accepted window is displayed. Click **Close**.
 11. Follow the steps on the First Steps tab to get started with IBM MessageSight. When you finish, click **Save and Close**.

Results

You have now installed IBM MessageSight for Developers on Oracle VirtualBox 4.3.10, and it is ready to be configured.

Supported browsers

You need one of the following supported web browsers. The browser must be installed on a system that is connected to the same network as the management port of the appliance:

Table 1. Supported web browsers by platform

	Linux RHEL 6	Windows 7		
	Linux SLES 11	Windows 8	Mac OS X	iOS 6
Firefox 10 ESR	Yes	Yes	Yes	No
Firefox 17 ESR	Yes	Yes	Yes	No
Chrome 18	Yes	Yes	No	No
Chrome 21	Yes	Yes	Yes	Yes
Internet Explorer 9	No	Yes	No	No
Internet Explorer 10	No	Yes	No	No
Safari 6	No	No	Yes	Yes

Note: If you are using Internet Explorer, you must ensure that you are running in Internet Explorer 9 standards mode. To change the mode, press F12.

Configuring IBM MessageSight

To configure MessageSight, you must configure the WebSphere MQ server connection channel, then configure MQ Connectivity in MessageSight, and then add a Destination Mapping rule.

Procedure

1. Configure the WebSphere MQ server connection channel.
 - a. Create a Windows user account called MSG. The user cannot be an administrator.
 - b. Use the DEFINE CHANNEL MQSC command to create a server-connection channel that is called SYSTEM.MSG.SVRCONN for sampleQM. Click **Start > Run**, type cmd to open a command prompt. On the command prompt, type the following command:

```
runmqsc sampleQM
DEFINE CHANNEL(SYSTEM.MSG.SVRCONN) CHLTYPE(SVRCONN) TRPTYPE(TCP) MCAUSER(MSG)
```

- c. Grant general WebSphere MQ access to the user account MSG:

```
SET AUTHREC OBJTYPE(QMGR) PRINCIPAL('MSG') AUTHADD(CONNECT, INQ, DSP)
```

- d. Grant authority for the user account MSG to issue requests to WebSphere MQ:

```
SET AUTHREC PROFILE('SYSTEM.DEFAULT.MODEL.QUEUE') OBJTYPE(QUEUE) PRINCIPAL('MSG') AUTHADD(DSP, GET)
SET AUTHREC PROFILE('SYSTEM.ADMIN.COMMAND.QUEUE') OBJTYPE(QUEUE) PRINCIPAL('MSG') AUTHADD(DSP, PUT)
```


- e. Grant authority for the user account MSG to create a WebSphere MQ queue for IBM MessageSight synchronization records:

```
SET AUTHREC PROFILE('SYSTEM.IMA.*') OBJTYPE(QUEUE) PRINCIPAL('MSG') AUTHADD(CRT, PUT, GET, BROWSE)
SET AUTHREC PROFILE('SYSTEM.DEFAULT.LOCAL.QUEUE') OBJTYPE(QUEUE) PRINCIPAL('MSG') AUTHADD(DSP)
```

- f. Grant publish and subscribe authority to AutoRemote:

```
DEFINE TOPIC(MSGTOPIC.sampleQM) TOPICSTR(AutoRemote)
SET AUTHREC PROFILE(MSGTOPIC.sampleQM) OBJTYPE(TOPIC) PRINCIPAL('MSG') AUTHADD(PUB, DSP)
```

2. Configure MQ Connectivity in MessageSight.

- a. In MessageSight, click **Messaging > MQ Connectivity**.
- b. Under Queue Manager Connection Properties, click the  icon.
- c. Enter the following details:

Name myQMC

Queue Manager
sampleQM

Connection Name

The IP address and port number where your WebSphere MQ is installed, for example, 224.0.138.177(1414). If there are more than one connection names, use comma to separate them.

Channel Name
SYSTEM.MSG.SVRCONN

SSL CipherSpec
Ignore this field.

- d. 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 Test connection succeeded.
- If test connection failed with error message CWLNA0309, click the message to see the detail. If the detail is Channel error while connecting to queue manager., run the queue manager with a different CCSID and then restart WebSphere MQ. For example, you can open a command prompt and type **ALTER QMGR CCSID(819)** in the command prompt, and then restart WebSphere MQ and click **Test Connection** again.
- If test connection failed with error message CWLNA0315, click the message to see the detail. If the detail is The operation is not authorized at the queue manager., disable the **Channel authentication records** of sampleQM in WebSphere MQ. To disable the **Channel authentication records** in sampleQM,
 - 1) Open WebSphere MQ Explorer and expand **IBM WebSphere MQ > Queue Managers**.
 - 2) Right click **sampleQM**.
 - 3) Click **Properties > Communication** and select **Disabled** in **Channel authentication records** list.

After the **Channel authentication records** is disabled, restart WebSphere MQ and click **Test Connection** again.

- e. Click **Save**.
3. Add a Destination Mapping Rule.

- a. Under Destination Mapping Rules, click the  icon.
- b. Enter the following details:

Name Sample Rule 1

Rule Type

Select the following type from the dropdown list: MessageSight topic subtree to MQ topic subtree

Source

AutoRemote

Destination

AutoRemote

Max Messages

Specify the maximum number of messages that can be buffered on the destination. The default is 5000.

Retained Messages

All

- c. Under the Associated queue manager connections section, select **myQMC** and click **Save**.

Results

You have now configured MessageSight to work with the sample application.

Deploying the sample application to Liberty profile

After you deploy the sample message driven bean (MDB) application, you can use it to verify that WebSphere Application Server Liberty profile is communicating with IBM MessageSight.

Before you begin

- Download the sample applications:
 - SampleCarApp.ear
 - AutoRemote.war
 - SampleCarApp.ear
 - AutoRemote.war
- You must have the WebSphere MQ Resource adapter. If not, see “Obtaining the WebSphere MQ Resource adapter” on page 23.

About this task

To configure the server, edit the `wlp\usr\servers\SCENARIO\server.xml` file in a text editor.

Procedure

1. Add the features to the feature manager in the `server.xml` file. In the `server.xml` file, add the following features into the feature manager section:

```
<featureManager>
  <feature>jsp-2.2</feature>
  <feature>wmqJmsClient-1.1</feature>
  <feature>jndi-1.0</feature>
  <feature>osgiconsole-1.0</feature>
  <feature>servlet-3.0</feature>
  <feature>jmsMdb-3.1</feature>
</featureManager>
```

Why am I doing this?

Adding the **wmqJmsClient-1.1** feature enables the Liberty server to load the necessary WebSphere MQ bundles that enable you to define the WebSphere MQ JMS resources. For example, the connection factory and activation specification properties provide client libraries to connect to the WebSphere MQ network.

2. Specify the location of the WebSphere MQ Resource adapter:

```
<variable name="wmqJmsClient.rar.location" value="/path to wmq rar/wmq.jmsra.rar"/>
```

Where the value attribute specifies the absolute path to the WebSphere MQ Resource adapter file that was downloaded, `wmq.jmsra.rar`.

3. Add a connection factory:

```
<jmsQueueConnectionFactory jndiName="jms/replyCF" connectionManagerRef="ConMgr6">
  <properties.wmqJms
    hostname="localhost"
    port="1414"
```

```

        channel="SYSTEM.DEF.SVRCONN"
    />
</jmsQueueConnectionFactory>
<connectionManager id="ConMgr6" maxPoolSize="2"/>

```

Note: If you are using another port rather than 1414 for your queue manager sampleQM, use the port number that is used. This port number is displayed in WebSphere MQ Explorer under **Queue managers > sampleQM > Advanced > Listeners**.

Why am I doing this?

- The value of **jndiName** is 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.
- The **channel** is the name of the WebSphere MQ channel to use for communication. If you do not have a specific channel that you want to use, specify the default channel that is created when you created the queue manager, SYSTEM.DEF.SVRCONN.

4. Add a topic:

```

<jmsTopic id="myTopic">
  <properties.wmqJms baseTopicName="#" />
</jmsTopic>

```

5. Add an activation specification:

```

<jmsActivationSpec id="SampleCarApp/SampleCarAppEJB/MyMDB">
  <properties.wmqJms destinationRef="myTopic" destinationType="javax.jms.Topic" port="1414" />
</jmsActivationSpec>

```

6. Add the application:

```

<application type="ear" id="SampleCarApp" name="SampleCarApp" location="SampleCarApp.ear"/>

```

7. Copy the SampleCarApp.ear file to the \apps folder. The default file path is wlp\usr\servers\servername\apps.

8. Add the web application:

```

<webApplication id="AutoRemote" location="AutoRemote.war" name="AutoRemote"/>

```

9. Copy the AutoRemote.war file to the \apps folder.

10. Save the server.xml file.

11. Use the **server start SCENARIO** command to start the SCENARIO server.

Results

You have deployed the MDB application and web application.

Verifying the solution

Run the sample JMS and message drive bean (MDB) applications to verify that they can communicate with each other through MessageSight and WebSphere MQ.

About this task

You have now connected MessageSight to WebSphere MQ and WebSphere Application Server Liberty profile, and installed a web application and Java EE application on Liberty profile. To verify that this has been done correctly, use the web application to send messages with some example cars.

Procedure

1. Ensure that MessageSight, WebSphere MQ, and WebSphere Application Server Liberty profile are started and configured according to the sample IT configuration.
2. Open your browser. The application uses the WebSocket protocol, which is supported by Mozilla Firefox 6.0 and later, Google Chrome 14.0 and later, and Safari 6.0 and later.
3. Load the web application. If you are on the same machine where Liberty profile is running, then this is `http://localhost:9080/AutoRemote` by default.
4. In each of the windows, enter the server address and default port (16102) of your MessageSight server.
5. There are three Remote/Car pairs. In order for the remote to connect to a car, the driver name should be the same for that pair. You may have a different driver name for each Remote/Car pairing. Click **Connect** to connect the car and remote pair.
6. Change the state of the cars through the remote and observe how the store page updates the statistics. The remotes and cars are communicating through only MessageSight.
7. Use the remote to change the state of the cars.

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.



Printed in USA