

7.5

IBM WebSphere MQ Hypervisor editions



Note

Before using this information and the product it supports, read the information in [“Notices” on page 109](#).

This edition applies to version 7 release 5 of IBM® WebSphere® MQ and to all subsequent releases and modifications until otherwise indicated in new editions.

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 International Business Machines Corporation 2007, 2024.**

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

Contents

WebSphere MQ Hypervisor editions.....	5
Overview of IBM WebSphere MQ Hypervisor editions.....	7
Parts and patterns.....	8
Planning.....	10
Installing.....	12
Adding an IBM WebSphere MQ virtual image to an appliance.....	14
Adding an IBM WebSphere MQ script package to an appliance.....	16
Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance.....	17
Loading the IBM WebSphere MQ virtual image for VMware ESX hypervisor.....	20
Installing IBM WebSphere MQ Managed File Transfer.....	21
Installing IBM WebSphere MQ Advanced Message Security (AMS).....	23
Applying service to a virtual system instance that contains IBM WebSphere MQ basic parts.....	24
Security.....	25
Example: Deploy a local queue with access permissions for an authorized user.....	27
Configuration.....	28
Creating a pattern.....	28
Copying a pattern.....	29
Adding the IBM WebSphere MQ basic part to a pattern.....	30
Adding MQSC commands to a pattern.....	31
Adding a queue manager to a cluster.....	33
Removing a queue manager from a cluster.....	35
Deploying a pattern that contains an IBM WebSphere MQ basic part.....	37
Deploying an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor.....	38
Verifying a deployed IBM WebSphere MQ basic part from an SSH terminal emulator.....	39
Verifying a deployed IBM WebSphere MQ basic part from a VNC session.....	41
Verifying deployed IBM WebSphere MQ parts from a remote IBM WebSphere MQ MQI client.....	44
Administration.....	46
Running IBM WebSphere MQ Explorer.....	46
Collecting IBM WebSphere MQ error logs from a virtual system instance.....	49
Reference.....	50
WebSphere MQ part.....	50
WebSphere MQ virtual system pattern.....	59
WebSphere MQ Hypervisor Edition script packages.....	60
WebSphere MQ Hypervisor Edition command scripts.....	61
Virtual machine image to deploy to AIX from an appliance.....	64
Virtual machine image to deploy to Linux from an appliance.....	65
Virtual machine image to deploy to VMware ESX running Linux.....	66
Glossary.....	67
A.....	67
B.....	70
C.....	71
D.....	75
E.....	77
F.....	79
G.....	80
H.....	80
I.....	81
J.....	83
K.....	83
L.....	84
M.....	85

N.....	89
O.....	90
P.....	91
Q.....	94
R.....	95
S.....	98
T.....	103
U.....	105
V.....	106
W.....	106
X.....	107
Accessibility features for IBM WebSphere MQ.....	107
Accessibility on Windows.....	108
Notices.....	109
Programming interface information.....	110
Trademarks.....	110

Overview of IBM WebSphere MQ Hypervisor editions

The IBM WebSphere MQ Hypervisor Edition for Red Hat® Enterprise Linux® and IBM WebSphere MQ Hypervisor Edition for AIX® are self-contained virtual machine images. The images contain the operating system and IBM WebSphere MQ. You can deploy the virtual machine images into a cloud with IBM Workload Deployer or IBM PureApplication® System. You can also load IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux directly into VMware ESX hypervisor.

Virtual systems and clouds

A cloud contains *virtual systems* that are delivered as services. You create a cloud by deploying virtual systems. Virtual systems are composed of *hypervisors*, and *virtual machine images* that contain an operating system and any number of applications; see [Figure 1 on page 5](#).

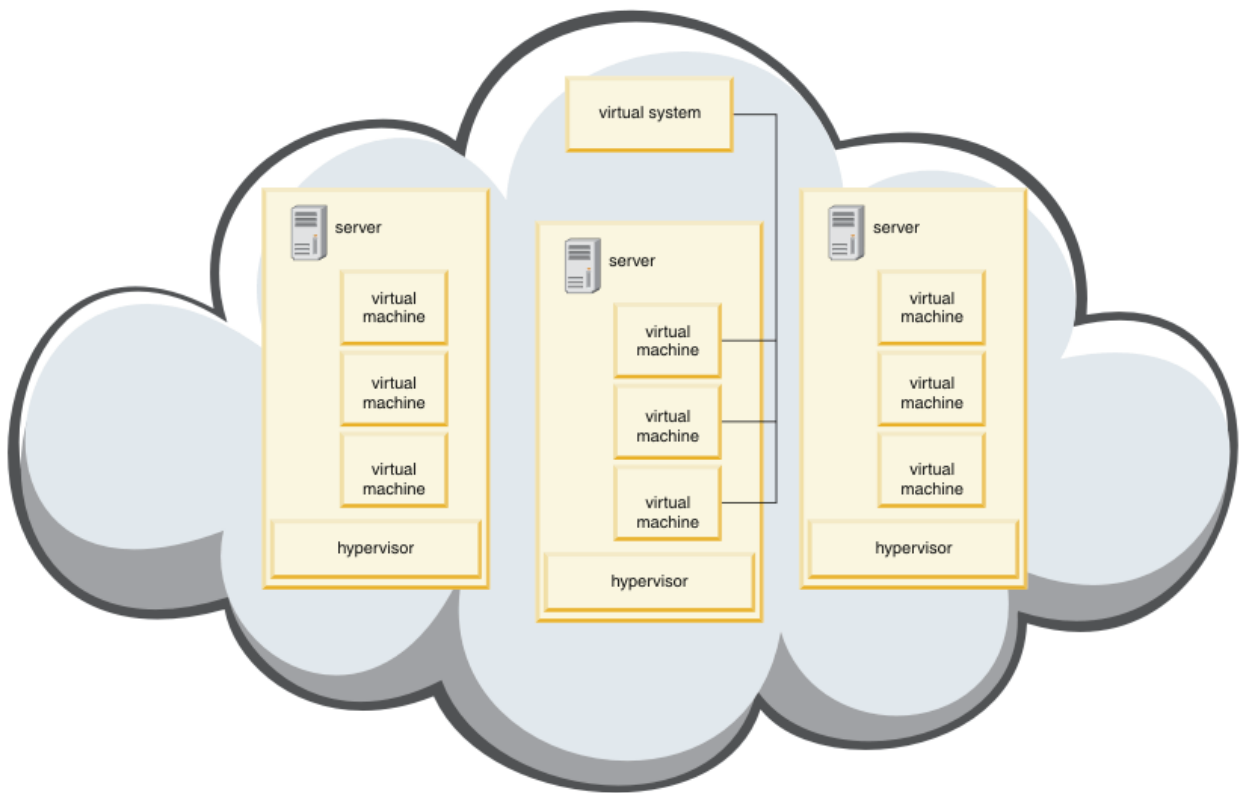


Figure 1. A cloud that contains a virtual system

IBM WebSphere MQ Hypervisor Edition contains virtual machine images that you configure to create applications and virtual systems, which you deploy to a cloud.

Hypervisors

Virtual machine images are run by a hypervisor, which virtualizes the hardware and operating system on which it runs. IBM WebSphere MQ Hypervisor Edition for AIX is run by PowerVM® hypervisor, and IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux is run by VMware ESX hypervisor.

A hypervisor divides the physical resources of a server among multiple virtual machines. It manages the state of the virtual machines on a physical machine, dividing the processors, memory, and other resources between the virtual machines. Each virtual machine isolates a running virtual machine image. The image runs securely on the same server as other images, increasing the utilization of the server.

Appliances

IBM Workload Deployer and IBM PureApplication System are known as *appliances*. They manage virtual systems and clouds.

With an appliance, you install, configure, and deploy virtual machine images and hypervisors that run on one or more servers. Interact with the appliance with a browser, the command interface, or the Representational State Transfer programming interface (REST API) supplied by the appliance.

In an appliance, each virtual machine image is known as a *part*. You configure and assemble parts into *topology patterns*. You deploy the patterns as *virtual system instances* to a *cloud group*, or to a collection of hypervisors. The appliance maintains a catalog of the topology patterns, virtual machine instances, cloud groups, scripts, virtual machine images, hypervisors, and other resources.

IBM WebSphere MQ Hypervisor editions

IBM WebSphere MQ Hypervisor editions contain virtual images, and other resources, which you install onto IBM Workload Deployer and IBM PureApplication System. You can also deploy the virtual image for IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux directly into VMware ESX hypervisor.

The virtual images contain the AIX or Linux operating system and a IBM WebSphere MQ installation.

The other resources include a IBM WebSphere MQ basic part, script packages, and a Python script. The Python script loads the IBM WebSphere MQ virtual image and script packages onto an appliance, and creates a default IBM WebSphere MQ virtual system pattern.

Add the IBM WebSphere MQ basic part to the patterns you create, which you deploy to a cloud to create virtual systems. The part creates a queue manager, and other IBM WebSphere MQ objects.

IBM WebSphere MQ Hypervisor editions contains script packages. With the cluster script packages, you can configure a pattern to add or remove a cluster of queue managers. The other script package runs the MQSC command tool. Use this script package to customize patterns containing the IBM WebSphere MQ basic part.

Related concepts

[“Configuration” on page 28](#)

Tasks to help you configure the deployment of IBM WebSphere MQ Hypervisor editions to an appliance or to VMware ESX hypervisor.

Related tasks

[“Planning” on page 10](#)

You can create, deploy, and manage IBM WebSphere MQ Hypervisor editions on an appliance, and you can also deploy IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to VMware ESX hypervisor.

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

Related reference

[“Reference” on page 50](#)

IBM WebSphere MQ Hypervisor Edition reference information

Related information

[Virtualization with IBM Workload Deployer](#)

[IBM Workload Deployer: Pattern-based Application and Middleware Deployments in a Private Cloud](#)

[Overview of IBM PureSystems](#)

[Preparing for IBM PureApplication System: a five part series](#)

Overview of IBM WebSphere MQ Hypervisor editions

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux and IBM WebSphere MQ Hypervisor Edition for AIX are self-contained virtual machine images. The images contain the operating system and IBM WebSphere MQ. You can deploy the virtual machine images into a cloud with IBM Workload Deployer or IBM PureApplication System. You can also load IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux directly into VMware ESX hypervisor.

Virtual systems and clouds

A cloud contains *virtual systems* that are delivered as services. You create a cloud by deploying virtual systems. Virtual systems are composed of *hypervisors*, and *virtual machine images* that contain an operating system and any number of applications; see [Figure 2 on page 7](#).

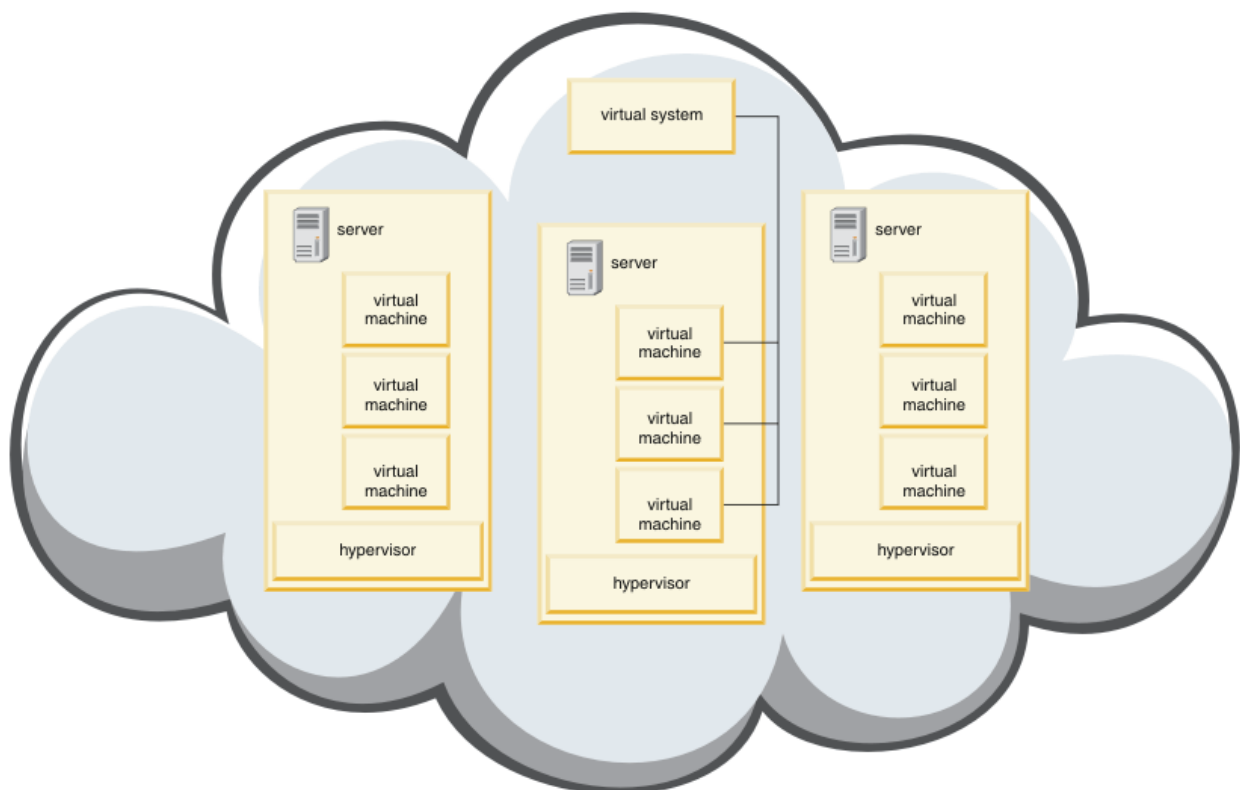


Figure 2. A cloud that contains a virtual system

IBM WebSphere MQ Hypervisor Edition contains virtual machine images that you configure to create applications and virtual systems, which you deploy to a cloud.

Hypervisors

Virtual machine images are run by a hypervisor, which virtualizes the hardware and operating system on which it runs. IBM WebSphere MQ Hypervisor Edition for AIX is run by PowerVM hypervisor, and IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux is run by VMware ESX hypervisor.

A hypervisor divides the physical resources of a server among multiple virtual machines. It manages the state of the virtual machines on a physical machine, dividing the processors, memory, and other resources between the virtual machines. Each virtual machine isolates a running virtual machine image. The image runs securely on the same server as other images, increasing the utilization of the server.

Appliances

IBM Workload Deployer and IBM PureApplication System are known as *appliances*. They manage virtual systems and clouds.

With an appliance, you install, configure, and deploy virtual machine images and hypervisors that run on one or more servers. Interact with the appliance with a browser, the command interface, or the Representational State Transfer programming interface (REST API) supplied by the appliance.

In an appliance, each virtual machine image is known as a *part*. You configure and assemble parts into *topology patterns*. You deploy the patterns as *virtual system instances* to a *cloud group*, or to a collection of hypervisors. The appliance maintains a catalog of the topology patterns, virtual machine instances, cloud groups, scripts, virtual machine images, hypervisors, and other resources.

IBM WebSphere MQ Hypervisor editions

IBM WebSphere MQ Hypervisor editions contain virtual images, and other resources, which you install onto IBM Workload Deployer and IBM PureApplication System. You can also deploy the virtual image for IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux directly into VMware ESX hypervisor.

The virtual images contain the AIX or Linux operating system and a IBM WebSphere MQ installation.

The other resources include a IBM WebSphere MQ basic part, script packages, and a Python script. The Python script loads the IBM WebSphere MQ virtual image and script packages onto an appliance, and creates a default IBM WebSphere MQ virtual system pattern.

Add the IBM WebSphere MQ basic part to the patterns you create, which you deploy to a cloud to create virtual systems. The part creates a queue manager, and other IBM WebSphere MQ objects.

IBM WebSphere MQ Hypervisor editions contains script packages. With the cluster script packages, you can configure a pattern to add or remove a cluster of queue managers. The other script package runs the MQSC command tool. Use this script package to customize patterns containing the IBM WebSphere MQ basic part.

Related concepts

[“Configuration” on page 28](#)

Tasks to help you configure the deployment of IBM WebSphere MQ Hypervisor editions to an appliance or to VMware ESX hypervisor.

Related tasks

[“Planning” on page 10](#)

You can create, deploy, and manage IBM WebSphere MQ Hypervisor editions on an appliance, and you can also deploy IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to VMware ESX hypervisor.

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

Related reference

[“Reference” on page 50](#)

IBM WebSphere MQ Hypervisor Edition reference information

Related information

[Virtualization with IBM Workload Deployer](#)

[IBM Workload Deployer: Pattern-based Application and Middleware Deployments in a Private Cloud](#)

[Overview of IBM PureSystems](#)

[Preparing for IBM PureApplication System: a five part series](#)

Parts and patterns

IBM WebSphere MQ Hypervisor editions contain a IBM WebSphere MQ basic part which you include in virtual system patterns. Virtual system patterns are made up of parts and parts have properties. Each part

represents a single virtual machine. A *pattern* provides a topology definition for repeatable deployment that can be shared. Patterns describe the function provided by each virtual machine in a virtual system. Each function is identified as a part in the pattern

Parts

A part describes the components that are configured on a virtual machine. Use parts to create patterns. Each part has a set of properties (parameters) that are used during deployment to help define the overall configuration of the virtual system.

Parts can also contain add-ons and scripts that might have parameters. You can customize a part by modifying its parameters, by adding a script package, or both. You must combine one or more parts into a pattern to deploy a part to a virtual system.

The basic and advanced IBM WebSphere MQ parts provided in the IBM WebSphere MQ Hypervisor editions for Version 7.0.1 are combined into a single IBM WebSphere MQ basic part in Version 7.5.

Patterns

Patterns take on the characteristics of their associated parts. For example, when a part is put into a pattern, which is then deployed, the result is a virtual machine that has a running IBM WebSphere MQ instance. If you create a pattern with two IBM WebSphere MQ basic parts, the virtual system has two virtual machines.

You can use default patterns, create new patterns, lock patterns, and edit patterns that are not locked. You cannot unlock and modify a locked pattern. You can copy it to create a pattern that you can modify.

Default patterns

The Python installation script creates a default IBM WebSphere MQ virtual system pattern, which you cannot modify. You can customize the pattern when you deploy it. You can also clone the pattern, and modify the cloned pattern.

Custom patterns

You can build custom patterns out of the IBM WebSphere MQ basic part that is supplied with the IBM WebSphere MQ Hypervisor editions. You must understand IBM WebSphere MQ and the IBM WebSphere MQ basic part to create custom patterns.

Creating and deploying patterns

Create and deploy patterns with the IBM Workload Deployer or IBM PureApplication System appliances.

Related concepts

[“Configuration” on page 28](#)

Tasks to help you configure the deployment of IBM WebSphere MQ Hypervisor editions to an appliance or to VMware ESX hypervisor.

Related tasks

[“Creating a pattern” on page 28](#)

Create a virtual system pattern.

[“Adding the IBM WebSphere MQ basic part to a pattern” on page 30](#)

Add the IBM WebSphere MQ basic part to an IBM WebSphere MQ pattern, and edit its properties to start configuring the pattern.

[“Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance” on page 17](#)

Install the IBM WebSphere MQ virtual image and script package to an appliance from a Windows or Linux workstation.

Related reference

[Parts, add-ons, and scripts](#)

[“IBM WebSphere MQ basic part” on page 50](#)

[“IBM WebSphere MQ virtual system pattern” on page 59](#)

Planning

You can create, deploy, and manage IBM WebSphere MQ Hypervisor editions on an appliance, and you can also deploy IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to VMware ESX hypervisor.

Learn about IBM WebSphere MQ and the IBM Workload Deployer or IBM PureApplication System you plan to run. The product documentation is a brief guide to adding IBM WebSphere MQ Hypervisor editions to appliances and running IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux with VMware ESX hypervisor without an appliance. To plan and deploy a solution you must supplement this guide with information about IBM WebSphere MQ and the appliance and hypervisors you plan to run.

You can run IBM WebSphere MQ Hypervisor editions in three environments. The environments have different characteristics. Choose the one that meets your requirements best.

You have three choices:

- Run IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux with VMware ESX hypervisor

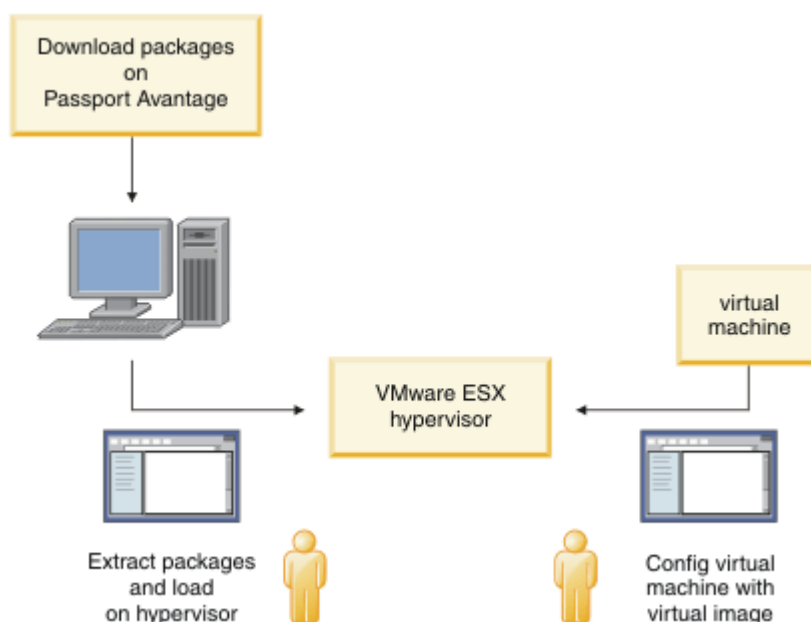


Figure 3. Deploy with VMware ESX hypervisor

In this environment, you configure the VMware ESX hypervisor virtual machine to allocate server resources to run the virtual image.

To configure and manage IBM WebSphere MQ, start the virtual machine. You can save multiple virtual machine images with different configurations, and copy multiple instances of the virtual machines onto one or more servers.

Follow the steps in [“Installing”](#) on page 12, [“Loading the IBM WebSphere MQ virtual image for VMware ESX hypervisor”](#) on page 20, and [“Deploying an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor”](#) on page 38 to download, install, and configure IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux.

- Deploy IBM WebSphere MQ Hypervisor editions with IBM Workload Deployer

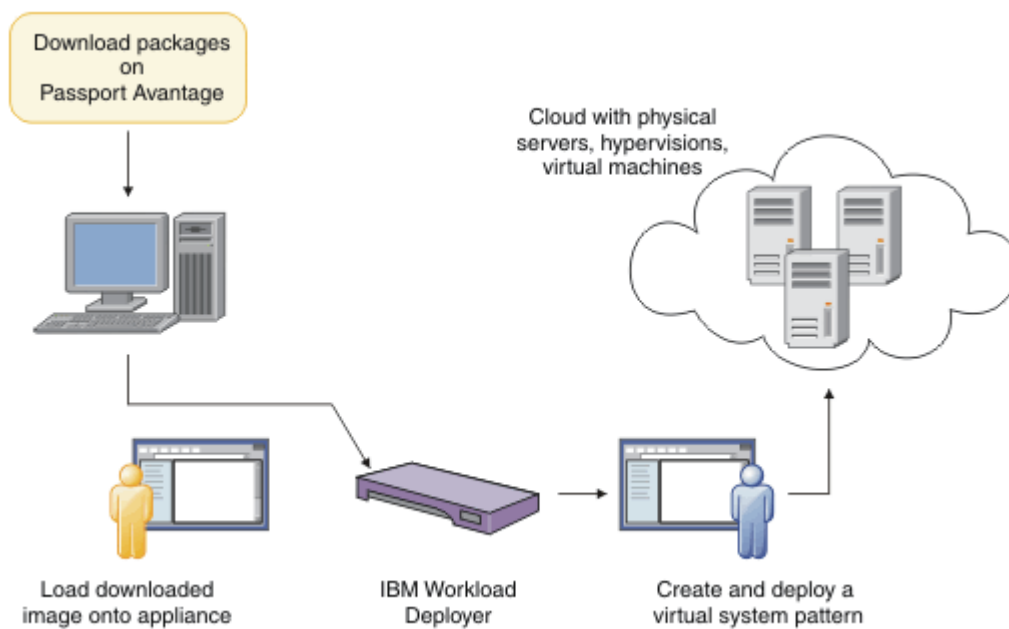


Figure 4. Deploy IBM WebSphere MQ Hypervisor editions with IBM Workload Deployer

Deploy IBM WebSphere MQ Hypervisor editions from IBM Workload Deployer to a cloud.

You can deploy virtual systems with the default IBM WebSphere MQ virtual system pattern, or create virtual system patterns with the IBM WebSphere MQ basic part; see [“Deploying a pattern that contains an IBM WebSphere MQ basic part”](#) on page 37.

Download and install IBM WebSphere MQ Hypervisor editions from Passport Advantage into the IBM Workload Deployer virtual image catalog; see [“Adding an IBM WebSphere MQ virtual image to an appliance”](#) on page 14.

You must provision real physical servers and hypervisors to deploy the virtual systems from IBM Workload Deployer to a cloud. IBM Workload Deployer manages clouds, it is not a run time server.

- Run IBM WebSphere MQ Hypervisor editions in IBM PureApplication System

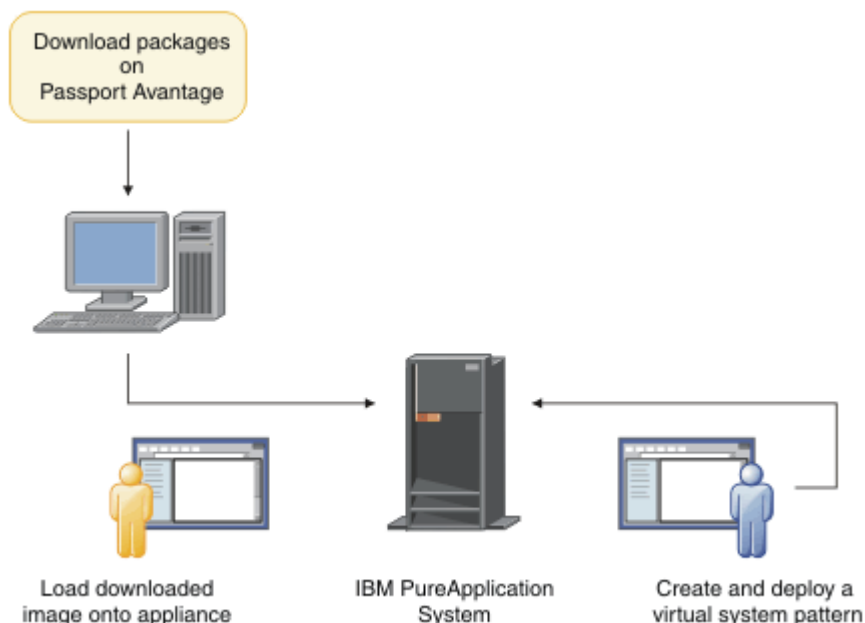


Figure 5. Run IBM WebSphere MQ Hypervisor editions with IBM PureApplication System

Download and install IBM WebSphere MQ Hypervisor editions into the IBM PureApplication System virtual image catalog; see [“Installing” on page 12](#).

With IBM PureApplication System you can add the IBM WebSphere MQ basic part to create virtual system patterns. Deploy the virtual system patterns to IBM PureApplication System. IBM PureApplication System includes the servers and hypervisors to run instances of virtual system patterns. It is both a management and run time server for clouds.

Related concepts

[“Overview of IBM WebSphere MQ Hypervisor editions” on page 5](#)

The IBM WebSphere MQ Hypervisor Edition for Red Hat® Enterprise Linux® and IBM WebSphere MQ Hypervisor Edition for AIX® are self-contained virtual machine images. The images contain the operating system and IBM WebSphere MQ. You can deploy the virtual machine images into a cloud with IBM Workload Deployer or IBM PureApplication® System. You can also load IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux directly into VMware ESX hypervisor.

[“Configuration” on page 28](#)

Tasks to help you configure the deployment of IBM WebSphere MQ Hypervisor editions to an appliance or to VMware ESX hypervisor.

[Deployment methods](#)

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

Related reference

[“Reference” on page 50](#)

IBM WebSphere MQ Hypervisor Edition reference information

Installing

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

If you are installing IBM WebSphere MQ Hypervisor editions onto IBM Workload Deployer or IBM PureApplication System, check your access and permissions to use the device.

- Check that you can establish an active connection to the appliance.
- Check that your user ID on the appliance has one of the following permissions:
 - Create new catalog content.
 - Cloud administration.

From the appliance user interface, click **System > Users** and in the list of users, click your user name.

The IBM WebSphere MQ Hypervisor Edition eAssemblies on Passport Advantage contain compressed files in .tgz format. One contains the virtual image for VMware ESX hypervisor, and the other the virtual image and script packages for IBM Workload Deployer or IBM PureApplication System. Load the same virtual image and script packages on either appliance.

The preliminary installation steps are to download and uncompress the installation files and transfer the files to the target server or appliance. Then manually load the IBM WebSphere MQ virtual image for IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux into VMware ESX hypervisor, or add the script bundles and the virtual images to the catalog on either IBM Workload Deployer or IBM PureApplication System appliances. The instructions are the same for both appliances.

1. Find the latest versions of IBM WebSphere MQ Hypervisor Edition for AIX and IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux on Passport Advantage; see [Passport Advantage®](#) and [Passport Advantage Express®](#) web site.

The full product name of IBM WebSphere MQ Hypervisor Edition for AIX is IBM WebSphere MQ Hypervisor Edition V7.5 for AIX.

2. Optional: Check whether the version on Passport Advantage is more recent than the version on IBM Workload Deployer.
 - a) From the IBM Workload Deployer browser, click **Catalog > Virtual images**
 - b) In the list of virtual images, find the latest release of IBM WebSphere MQ Version 7.5 or IBM WebSphere MQ Version 7.0.1.
 - c) Compare the release number with the version on Passport Advantage.

Continue with these steps if the version on Passport Advantage is more recent.

3. Download the compressed IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux eAssembly files that you require from Passport Advantage onto a UNIX or Windows server.
 - Choose whether to download the IBM WebSphere MQ virtual image for VMware ESX hypervisor or the virtual images and script bundles for the IBM Workload Deployer and IBM PureApplication System appliances.
 - Download the files onto a UNIX or Windows workstation on which you can uncompress GNU compressed file format files.
 - You must be able to transfer the uncompressed files from the workstation to the target server or appliance.
4. Extract the downloaded file into an empty directory.
 - The compressed files for the appliances are in GNU compressed file format. The extracted files are approximately 20% larger than each downloaded compressed file.
 - On UNIX, extract the image from the package by running the **tar** command with the **-z** (.gzip) option. **tar** must be at version 1.20 or higher.
 - You can uncompress the tar file on Windows. Uncompress it with one of the commonly available tools that can be used to extract .tar files.
 - The compressed file for VMware ESX hypervisor is in .zip format. You can uncompress it with one of the commonly available tools that can be used to extract .zip files.
5. Install IBM WebSphere MQ Hypervisor Edition on the target server or appliance.

You have these choices:

- Deploy the IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image directly into VMware ESX hypervisor; see [“Loading the IBM WebSphere MQ virtual image for VMware ESX hypervisor” on page 20](#).
- Add the IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image to the catalog of virtual images on the IBM Workload Deployer or IBM PureApplication System appliances; see [“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14](#).
- Add the IBM WebSphere MQ script packages to the catalog of script packages on the IBM Workload Deployer or IBM PureApplication System appliances; see [“Adding an IBM WebSphere MQ script package to an appliance” on page 16](#).
- Download the IBM Workload Deployer Command Line Interpreter (CLI) client, and run a command to install the image or scripts from a client workstation; see [“Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance” on page 17](#).

Related tasks

[“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14](#)

Add the IBM WebSphere MQ virtual image for IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to the image catalog in IBM Workload Deployer or IBM PureApplication System.

[“Adding an IBM WebSphere MQ script package to an appliance” on page 16](#)

Add an IBM WebSphere MQ script package for IBM WebSphere MQ Hypervisor Edition to the script catalog in IBM Workload Deployer or IBM PureApplication System.

[“Loading the IBM WebSphere MQ virtual image for VMware ESX hypervisor” on page 20](#)

Save a copy of the IBM WebSphere MQ virtual image for VMware ESX hypervisor in the VMware ESX hypervisor data store.

Related reference

[“The IBM WebSphere MQ Hypervisor Edition for AIX virtual machine image for appliances” on page 64](#)

The IBM WebSphere MQ Hypervisor Edition for AIX virtual image provides the operating system and product binary files that are required to create virtual system instances to deploy from IBM Workload Deployer or run on IBM PureApplication System.

[“The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image for appliances” on page 65](#)

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image provides the operating system and product binary files that are required to create virtual system instances to deploy from IBM Workload Deployer or run on IBM PureApplication System.

Adding an IBM WebSphere MQ virtual image to an appliance

Add the IBM WebSphere MQ virtual image for IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to the image catalog in IBM Workload Deployer or IBM PureApplication System.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Check that your user ID on the appliance has one of the following permissions:
 - Create new catalog content.
 - Cloud administration.

From the appliance user interface, click **System > Users** and in the list of users, click your user name.

- Complete the [“Installing” on page 12](#) task to download and uncompress the IBM WebSphere MQ virtual images for IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to a workstation.
- Consider adding the virtual image to the appliance from a client workstation by running a command script; see [“Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance” on page 17](#).
- If the IBM WebSphere MQ virtual image at the same version is already installed, you cannot overwrite it. You must delete it first; see [“IBM WebSphere MQ Hypervisor Edition command scripts” on page 61](#) for information about versions.

Complete these steps to add the IBM WebSphere MQ Hypervisor Edition virtual machine image that you downloaded to your virtual machine into the image catalog.

1. Load the virtual image file onto the appliance.

The IBM WebSphere MQ virtual image file, is in the root directory where you extracted the package.

The virtual image for IBM WebSphere MQ Hypervisor Edition V7.5 for AIX is called `WebSphere_MQ_version_64_AIX_7_IPAS.ova`. The virtual image for IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux Server is called `WebSphere_MQ_version_64_RHEL_6_X86.ova`.

- Configure an HTTP server to either serve or pick up the virtual image file.
 - Copy the virtual image with the Secure Copy Protocol (**scp**) command.
2. From the appliance user interface, open the **Virtual Images** window.
 - From the menu bar, click **Catalog > Virtual Images**.
 - From the Welcome page, under **"Setting up your private cloud"**, click **Add virtual images**.

The **Virtual Images** window opens

3. Click the Add icon , next to the **Virtual Images** window title to import the virtual image.

A window "**Enter the remote path of the virtual image**" opens

4. In the **OVA file location** field, enter the URL for the virtual image file you copied in step "1" on page 14.
5. If you set security on the OVA file, use the **User name** and **Password** fields to enter the required user name and password. Retype your password in the **Verify password** field.
6. Click **OK**.

A temporary Virtual Image is added to the list of Virtual Images. The properties of the virtual image are listed on the canvas. The status property shows that the virtual image is being transferred to the appliance.

7. Monitor the transfer.

- a) Click **System > Task Queue** to display the list of tasks.


The appliance displays a list of active and pending tasks steps.

- b) Click **Catalog > Virtual Images** to return to the list of virtual images.

The appliance displays a list of virtual images.

8. Accept the license agreements for the virtual image.

- a) Find the virtual image in the list of Virtual Images and click its name.

The image is named IBM WebSphere MQ followed by the numbers, *version.release.modification.fix*, without trailing zeros. If the license is not accepted, it is shown by the icon .

The canvas opens for the image. You can change the description.

- b) Click **accept...** in the **License agreement** field.

The window "**All licenses must be accepted**" opens.

- c) Click each of the lines in the window to accept each of the licenses and then click **OK**.

9. Set the permissions for the image.

To change the default settings:

- Double-click **Add more...** in the field **Access granted to** and select a user or group of users.
- Click **read** to change the access permission to **write**. Click it again to change the permission to **all**.

The IBM WebSphere MQ Hypervisor Edition virtual machine image is now available in the appliance image catalog.

Manually create patterns and script packages that you can use to deploy virtual systems.

Related tasks

["Installing" on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

Related reference

["The IBM WebSphere MQ Hypervisor Edition for AIX virtual machine image for appliances" on page 64](#)

The IBM WebSphere MQ Hypervisor Edition for AIX virtual image provides the operating system and product binary files that are required to create virtual system instances to deploy from IBM Workload Deployer or run on IBM PureApplication System.

["The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image for appliances" on page 65](#)

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image provides the operating system and product binary files that are required to create virtual system instances to deploy from IBM Workload Deployer or run on IBM PureApplication System.

Adding an IBM WebSphere MQ script package to an appliance

Add an IBM WebSphere MQ script package for IBM WebSphere MQ Hypervisor Edition to the script catalog in IBM Workload Deployer or IBM PureApplication System.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Do the task, “Installing” on [page 12](#), to download and uncompress the IBM WebSphere MQ script packages to a workstation.
- Check that your user ID on the appliance has one of the following permissions:
 - Create new catalog content.
 - Cloud administration.

From the appliance user interface, click **System > Users** and in the list of users, click your user name.

- Consider adding the script packages to the appliance from a client workstation by running a command script; see “Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance” on [page 17](#).
- If you have already installed these script packages to the appliance with the standard IBM WebSphere MQ script package names, you cannot replace them. Either you must delete the existing packages before adding the packages again, or add the new packages with different package names.

Complete these steps to add a IBM WebSphere MQ Hypervisor Edition script package to the script catalog.

IBM WebSphere MQ Hypervisor editions have the following script packages that you can import.

Table 1. IBM WebSphere MQ script packages	
File name	Script package
WMQAddToCluster.zip	WMQ: Add to Queue Manager Cluster
WMQRemoveFromCluster.zip	WMQ: Remove from Queue Manager Cluster
WMQExecuteMQSC.zip	WMQ: Run MQSC Scripts

The script packages for IBM WebSphere MQ Hypervisor Edition for AIX and IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux are the same.

1. From the appliance user interface, open the **Script Packages** window.
 - From the menu bar, click **Catalog > Script Packages**.
 - From the **Welcome** page, under **Setting up your private cloud**, click **Add script packages**.

The **Script Packages** window opens.

2. Click the new icon , next to the **Script Packages** window title, to create a script package.

A “Enter the name of the script package you want to create” window opens.

- a) Type the name from [Table 1 on page 16](#) that corresponds to the script you are going to upload into the **Script name** field. Alternatively, type a name of your own choosing.

A new script package is created, and shown in the list of script packages.

3. Click the script package to open a window to upload the script package.

- a) Click the **Script package file** field to browse for a script package.

The browse window opens on the workstation where you are running the browser.

- b) Go to the directory where you extracted the IBM WebSphere MQ script packages.
- c) Select a script package compressed file and click **Open** to return to the **Script Packages** window.
- d) Click **Upload**.
- 4. If you are uploading the WMQ: Remove from Queue Manager Cluster script package, change the default value of the **Executes** property.

The default value of the **Executes** property is set to at virtual system creation. Change this default value to at virtual system deletion.

- 5. Optional: Lock the script package.

After locking a script package, you cannot change it. You can either clone it, and edit the cloned package, or delete it.

The IBM WebSphere MQ script package image is now available in the appliance catalog.

Related tasks

[“Adding MQSC commands to a pattern” on page 31](#)

You can add one or more files of IBM WebSphere MQ commands to customize a pattern that contains IBM WebSphere MQ basic parts. The commands run when the pattern is deployed as a virtual system, when the virtual system is deleted, or when you choose to run the commands.

[“Adding a queue manager to a cluster” on page 33](#)

The WMQ: Add to Queue Manager Cluster cluster script adds a queue manager to a cluster.

[“Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance” on page 17](#)

Install the IBM WebSphere MQ virtual image and script package to an appliance from a Windows or Linux workstation.

Related reference

[“IBM WebSphere MQ Hypervisor Edition script packages” on page 60](#)

Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance

Install the IBM WebSphere MQ virtual image and script package to an appliance from a Windows or Linux workstation.

- Download the version of IBM WebSphere MQ Hypervisor Edition that you intend to install to your workstation; see [“Installing” on page 12](#). Alternatively, download IBM WebSphere MQ Hypervisor Edition to a file server rather than to the workstation where you plan to run the command script. You must be able to run the command script in the directory where you downloaded IBM WebSphere MQ Hypervisor Edition.

Tip: If you run the script in a remote directory, two network copies are involved: one to your client workstation, and one to the appliance. The virtual image is large: about 2.2 GB. If you add the virtual image directly to the appliance, only one network copy is required; see [“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14](#).

- Check that you can establish an active connection to the appliance.
- The command script MQHVE-RHEL-CreatePatternsandScripts.py requires a IBM WebSphere MQ basic part is installed on the appliance. Add the virtual image by running the command script MQHVE-RHEL-AddVirtualImage.py first, or by adding the image manually; see [“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14](#).
- If the virtual image, script packages, or basic IBM WebSphere MQ virtual system pattern are already installed, they are not replaced; see [“IBM WebSphere MQ Hypervisor Edition command scripts” on page 61](#). You can install multiple versions of the virtual image; see [“IBM WebSphere MQ Hypervisor Edition command scripts” on page 61](#).
- Check that your user ID on the appliance has one of the following permissions:
 - Create new catalog content.

- Cloud administration.

From the appliance user interface, click **System > Users** and in the list of users, click your user name.

Download the IBM Workload Deployer command-line tool to a workstation where you downloaded and expanded the IBM WebSphere MQ Hypervisor Edition installation materials. Run the command procedures provided in the installation materials to install the IBM WebSphere MQ virtual system image, default IBM WebSphere MQ Hypervisor Edition virtual system pattern, and script packages.

The examples that are shown are for Windows. Change \ to / to run the commands on Linux.

1. From the Welcome Page of the appliance user interface, click **Download Tooling > Download command line tool**, and follow the browser instructions to save the file on your workstation.

Download the tool from IBM Workload Deployer or IBM PureApplication System into a directory that you create on a Windows or Linux workstation.

In this task, the path to the directory that you create to contain the compressed file and the expanded contents, is labeled *expandDir*.

If the file is downloaded from IBM Workload Deployer it is called *deployer.cli-version.zip*. If it is downloaded from IBM PureApplication System it is called *pure.cli-version.zip*.

- "*version*" is a compound string, "*V.R.M.F-timestamp*". The *V.R.M.F* code is the version of the command-line tool.
- The rest of this task uses the label *appliance* instead of the names *deployer* or *pure*. Where you see the label *appliance*, substitute *deployer* or *pure*.

For example, *deployer.cli-version.zip* and *pure.cli-version.zip* are labeled *appliance.cli-version.zip* in the rest of this task.

2. Expand the contents of *expandDir\appliance.cli-version.zip* to *expandDir*.

The expansion creates a top-level directory *expandDir\appliance.cli*

3. Ensure that either the JAVA_HOME or the PATH environment variable is set to the location of your JRE.
4. If you are running Windows 2003 or Windows 2008 do the following step:
 - a) In the *expandDir\appliance.cli\lib\version* directory, modify the registry file and add the following line:

```
python.os=nt
```

By default, the only thing in the lib directory is a *version* subdirectory that matches firmware level of the appliance from which the CLI was downloaded. If you use this CLI installation to manage appliances at different firmware levels, you must create a subdirectory under the /lib directory for each firmware level. You must copy the registry file into each of these subdirectories; for example: *expandDir\appliance.cli\lib\3.0.0.0-12345\registry*

5. Verify the installation of the command-line interface.

Run one of the following commands from the *expandDir\appliance.cli\bin* directory.

- On Windows, run *appliance.bat*.
- On Linux, run *appliance*

```
expandDir\appliance.cli\bin\appliance
```

If the environment is set up correctly, you receive an informational message. The message says the command-line interface is working and provides further details about using the command-line interface. Type *exit* to return from the command-line interpreter.

6. Change into the directory where you downloaded and expanded IBM WebSphere MQ Hypervisor Edition.

In addition to the hypervisor virtual image, and script packages, the directory contains two python scripts.

- MQHVE-RHEL-AddVirtualImage.py
 - MQHVE-RHEL-CreatePatternsandScripts.py
7. Install the IBM WebSphere MQ Hypervisor Edition virtual image, scripts, and default IBM WebSphere MQ virtual system pattern.

Run the following command twice, first to install the virtual image, and second to install the script packages and create the default IBM WebSphere MQ virtual system pattern.

```
expandDir\appliance.cli\bin\appliance -h ipAddr -u userID -p password -f scriptName.py
```

The command script parameters are as follows:

expandDir

expandDir is the directory that contains the command-line interpreter for the appliance.

appliance

appliance is the name of the command. It is either *deployer* if the command-line interpreter is downloaded from IBM Workload Deployer or *pure* if the command-line interpreter is downloaded from IBM PureApplication System. You must use the command-line interpreter that corresponds to the platform of the appliance.

ipAddr

ipAddr is the IP address of the appliance.

userID

userID is a user that has "cloud administration" or "create new catalog content" authority. On IBM Workload Deployer, you can replace the user ID by the built-in operator ID, *cbadmin*.

The virtual image, script package, and default IBM WebSphere MQ virtual system pattern are owned by this user. Other users are given read-only access to it.

password

The password of the user or operator.

scriptName

scriptName is one of the following installation scripts:

MQHVE-RHEL-AddVirtualImage

Install the virtual image.

MQHVE-RHEL-CreatePatternsandScripts

Install the script packages and default IBM WebSphere MQ virtual system pattern.

MQHVE-AIX-AddVirtualImage

Install the virtual image.

MQHVE-AIX-CreatePatternsandScripts

Install the script packages and default IBM WebSphere MQ virtual system pattern.

Examples

```
C:\cli\pure.cli\bin\pure -h 10.0.0.1 -u admin -p password -f MQHVE-RHEL-AddVirtualImage.py
```

```
C:\cli\pure.cli\bin\pure -h 10.0.0.1 -u admin -p password -f MQHVE-AIX-AddVirtualImage.py
```

Figure 6. Installing the virtual image on IBM PureApplication System.

```
C:\cli\deployer.cli\bin\deployer -h 10.0.0.2 -u cbadmin -p cbadmin -f MQHVE-RHEL-CreatePatternsandScripts.py
```

```
C:\cli\deployer.cli\bin\deployer -h 10.0.0.2 -u cbadmin -p cbadmin -f MQHVE-AIX-CreatePatternsandScripts.py
```

Figure 7. Installing script packages on IBM Workload Deployer.

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

Related reference

[“IBM WebSphere MQ Hypervisor Edition command scripts” on page 61](#)

Loading the IBM WebSphere MQ virtual image for VMware ESX hypervisor

Save a copy of the IBM WebSphere MQ virtual image for VMware ESX hypervisor in the VMware ESX hypervisor data store.

- Install a copy of VMware vSphere Client on the workstation where you are going to perform the task.
- Do the task, [“Installing” on page 12](#), to download the compressed IBM WebSphere MQ VMware ESX hypervisor virtual image files to a workstation and uncompress the files into an empty directory.

Use the VMware vSphere Client to load the IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image into the VMware ESX hypervisor data store.

1. From the VMware vSphere Client, select the **Summary** tab. In the **Datastore** window, right-click a data store in which you want to store the virtual machine image, and click **Browse Datastore**.
2. Using the Datastore Browser, create a directory in which to store the virtual machine image.
3. Using the Datastore Browser, upload the following files:

```
os.vmdk  
os-flat.vmdk  
app.vmdk  
app-flat.vmdk  
wmqhve.vmx
```

or upload the whole directory.

Note: The Datastore Browser does not show the files `os-flat.vmdk` and `app-flat.vmdk`. By default it represents them as being part of `os.vmdk` and `app.vmdk`.

4. Register the virtual image on the hypervisor.
 - a) In the Datastore Browser, right-click on the file `wmqhve.vmx` and select **Add to inventory**, which starts a wizard.
 - b) Type a name for the virtual machine image and select **Next**.
 - c) Check that the virtual machine options are correct and select **Finish**.

The IBM WebSphere MQ Hypervisor Edition virtual image is loaded into VMware ESX hypervisor.

Configure the virtual image and start it; see [“Deploying an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor” on page 38](#).

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

[“Deploying an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor” on page 38](#)

Deploy an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor for the first time.

Related reference

[“The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image for VMware ESX hypervisor” on page 66](#)

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image provides the operating system and product binary files that are required to create virtual system instances that run in VMware ESX hypervisor.

Installing IBM WebSphere MQ Managed File Transfer

Install IBM WebSphere MQ Managed File Transfer after deploying a IBM WebSphere MQ basic part. For licensing reasons, when IBM WebSphere MQ Hypervisor Edition deploys the IBM WebSphere MQ basic part, it does not install IBM WebSphere MQ Managed File Transfer. Install IBM WebSphere MQ Managed File Transfer from its installation packages, which are deployed with the IBM WebSphere MQ basic part.

- You must have a license for some or all the IBM WebSphere MQ Managed File Transfer components, which are additional to the license for the IBM WebSphere MQ Hypervisor Edition that you installed; see [IBM WebSphere MQ Managed File Transfer](#).
- Remember the password for the `root` user ID, which is set in the IBM WebSphere MQ basic part properties, or from the password confirmation window if you deployed the IBM WebSphere MQ virtual image directly to VMware ESX hypervisor.
- You must also know the password for the `virtuser` user ID, if you are signing on with VNC from IBM Workload Deployer.
- You deployed a IBM WebSphere MQ virtual system instance from IBM Workload Deployer. Or, you loaded a IBM WebSphere MQ virtual image directly to VMware ESX hypervisor, configured, and started it successfully.
- If you loaded the IBM WebSphere MQ virtual image directly to VMware ESX hypervisor, you must remember its IP address. You set the IP address in step “7” on page 38 in [“Deploying an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor”](#) on page 38.

IBM WebSphere MQ Hypervisor Edition copies the IBM WebSphere MQ Managed File Transfer installation packages to the virtual machines where it installed IBM WebSphere MQ. If you loaded the IBM WebSphere MQ virtual machine image directly to VMware ESX hypervisor, the installation packages are included in the VMware ESX hypervisor virtual machine image.

Follow these steps to install some or all of the IBM WebSphere MQ Managed File Transfer installation packages. Your choice of which packages to install must be guided by which options you licensed. For a description of the options, see [IBM WebSphere MQ Managed File Transfer product options](#).

1. Log on to the `root` user ID.

Log on to root on the virtual machine that contains the IBM WebSphere MQ installation you want to update by following one of these procedures.

- Using VNC from the appliance user interface:
 - a. Do steps “1” on page 42 and “2” on page 42 in [“Verifying a deployed IBM WebSphere MQ basic part from a VNC session”](#) on page 41 to open a terminal session on the VNC desktop.
 - b. Run the super user command to log on to the `root` user ID:

```
su root
```

In response to the **Password:** prompt, type the password for the `root` user ID.

- Using an SSH terminal session:
 - a. Find an SSH terminal emulator, see [“Verifying a deployed IBM WebSphere MQ basic part from an SSH terminal emulator”](#) on page 39.
 - b. Find the IP address of a IBM WebSphere MQ virtual system instance, see step “1” on page 39 of [“Verifying a deployed IBM WebSphere MQ basic part from an SSH terminal emulator”](#) on page 39.
 - c. Connect the SSH terminal emulator to the virtual machine; see step “2” on page 40 of [“Verifying a deployed IBM WebSphere MQ basic part from an SSH terminal emulator”](#) on page 39. Log on as the `root` rather than the `virtuser` user ID.

2. Change to the directory where the IBM WebSphere MQ Managed File Transfer installation packages are stored.
 - a) Run the command `cd /home/virtuser/MQ75-MFT` on AIX and `cd /root/MQ75-MFT` on Linux to change to the directory.
 - b) Run the command `ls` to list the contents of the directory. Confirm the list of packages matches the list of IBM WebSphere MQ Managed File Transfer components; see [IBM WebSphere MQ components for Linux systems](#), or [IBM WebSphere MQ components for AIX systems](#).
3. Install IBM WebSphere MQ Managed File Transfer.

Depending on the virtual machine platform run the following command to install IBM WebSphere MQ Managed File Transfer.

- On Linux, do one of the following steps:

- Run the following **rpm** command to install all IBM WebSphere MQ Managed File Transfer components:

```
rpm -ivh MQSeriesFT*.rpm
```

- Run one or more of the following **rpm** commands to install IBM WebSphere MQ Managed File Transfer components; see [WebSphere MQ Managed File Transfer product options](#). If you install the components individually, they must be installed in the following three stage order. If you install any of the third stage components, install them in any order.

```
a. rpm -ivh MQSeriesFTBase*.rpm
b. rpm -ivh MQSeriesFTAgent*.rpm
c. - rpm -ivh MQSeriesFTLogger*.rpm
    - rpm -ivh MQSeriesFTTools*.rpm rpm
    - rpm -ivh MQSeriesFTService*.rpm
```

- On AIX, do one of the following steps:

- Run the following **installp** command to install all IBM WebSphere MQ Managed File Transfer components:

```
installp -acgXYd . all
```

- Run one or more of the following **installp** commands to install IBM WebSphere MQ Managed File Transfer components¹; see [WebSphere MQ Managed File Transfer product options](#):

```
installp -acgXYd . mqm.ft.agent
installp -acgXYd . mqm.ft.base
installp -acgXYd . mqm.ft.logger
installp -acgXYd . mqm.ft.service
installp -acgXYd . mqm.ft.tools
```

4. Verify that IBM WebSphere MQ Managed File Transfer is installed.

Run the command to display the version of IBM WebSphere MQ Managed File Transfer that is installed:

```
fteDisplayVersion -v
```

Configure IBM WebSphere MQ Managed File Transfer; see the listed scenarios.

Related concepts

[WebSphere MQ components for Linux systems](#)

[WebSphere MQ Managed File Transfer product options](#)

[Basic file transfer using the scripts](#)

[Two machine file transfer](#)

[Two machine file transfer using the scripts](#)

¹ Alternatively create a list of components in a file, and pass the name of the file with the `-f` flag.

[Adding audit capability to managed file transfer](#)

[WebSphere MQ components for AIX systems](#)

Related tasks

[Installing WebSphere MQ Managed File Transfer](#)

[Installing WebSphere MQ server on Linux](#)

[Installing WebSphere MQ server on AIX](#)

[Non-interactive installation of WebSphere MQ server on AIX](#)

Installing IBM WebSphere MQ Advanced Message Security (AMS)

Install IBM WebSphere MQ Advanced Message Security after deploying a IBM WebSphere MQ basic part. For licensing reasons, when IBM WebSphere MQ Advanced Message Security deploys the IBM WebSphere MQ basic part, it does not install IBM WebSphere MQ Advanced Message Security. Install IBM WebSphere MQ Advanced Message Security from its installation package, which is deployed with the IBM WebSphere MQ basic part.

- You must have a license for IBM WebSphere MQ Advanced Message Security, which is additional to the license for the IBM WebSphere MQ Hypervisor Edition that you installed; see [IBM WebSphere MQ Advanced Message Security](#).
- Remember the password for the `root` user ID, which is set in the IBM WebSphere MQ basic part properties, or from the password confirmation window if you deployed the IBM WebSphere MQ virtual image directly to VMware ESX hypervisor.
- You must also know the password for the `virtuser` user ID, if you are signing on with VNC from IBM Workload Deployer.
- You deployed a IBM WebSphere MQ virtual system instance from IBM Workload Deployer. Or, you loaded a IBM WebSphere MQ virtual image directly to VMware ESX hypervisor, configured, and started it successfully.
- If you loaded the IBM WebSphere MQ virtual image directly to VMware ESX hypervisor, you must remember its IP address. You set the IP address in step “7” on page 38 in [“Deploying an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor”](#) on page 38.

IBM WebSphere MQ Hypervisor Edition copies the IBM WebSphere MQ Advanced Message Security installation package to the virtual machines where it installed IBM WebSphere MQ. If you loaded the IBM WebSphere MQ virtual machine image directly to VMware ESX hypervisor, the installation packages are included in the VMware ESX hypervisor virtual machine image.

Follow these steps to install the IBM WebSphere MQ Advanced Message Security installation package.

1. Log on to the `root` user ID.

Log on to root on the virtual machine that contains the IBM WebSphere MQ installation you want to update by following one of these procedures.

- Using VNC from the appliance user interface:
 - a. Do steps “1” on page 42 and “2” on page 42 in [“Verifying a deployed IBM WebSphere MQ basic part from a VNC session”](#) on page 41 to open a terminal session on the VNC desktop.
 - b. Run the super user command to log on to the `root` user ID:

```
su root
```

In response to the **Password:** prompt, type the password for the `root` user ID.

- Using an SSH terminal session:
 - a. Find an SSH terminal emulator, see [“Verifying a deployed IBM WebSphere MQ basic part from an SSH terminal emulator”](#) on page 39.
 - b. Find the IP address of a IBM WebSphere MQ virtual system instance, see step “1” on page 39 of [“Verifying a deployed IBM WebSphere MQ basic part from an SSH terminal emulator”](#) on page 39.

- c. Connect the SSH terminal emulator to the virtual machine; see step “2” on page 40 of [“Verifying a deployed IBM WebSphere MQ basic part from an SSH terminal emulator”](#) on page 39. Log on as the root rather than the virtuser user ID.
2. Change to the directory where the IBM WebSphere MQ Advanced Message Security installation packages are stored.
 - a) Run the command `cd /home/virtuser/MQ75-AMS` on AIX and `cd /root/MQ75-AMS` on Linux to change to the directory.
 - b) Run the command `ls` to list the contents of the directory. Confirm the package matches the IBM WebSphere MQ Advanced Message Security component; see [IBM WebSphere MQ components for Linux systems](#), or [IBM WebSphere MQ components for AIX systems](#).
3. Install IBM WebSphere MQ Advanced Message Security.

Depending on the virtual machine platform run the following command to install the IBM WebSphere MQ Advanced Message Security component.

- On Linux:

```
rpm -ivh MQSeriesAMS*.rpm
```

- On AIX:

```
installp -a -c -Y -d mqm.ams.rte
```

4. Verify that IBM WebSphere MQ Advanced Message Security is installed.

Run the following command to display the version of IBM WebSphere MQ Managed File Transfer that is installed:

```
dspmqrver -p 128 -v
```

If IBM WebSphere MQ Advanced Message Security is installed, the command returns information about the installation.

Configure IBM WebSphere MQ Advanced Message Security; see the listed guides and scenarios.

Related concepts

[Quick Start Guide for WebSphere MQ Advanced Message Security on Windows](#)

[Quick Start Guide for WebSphere MQ Advanced Message Security on UNIX and Linux](#)

[Quick Start Guide for WebSphere MQ Advanced Message Security for Java clients](#)

Related tasks

[Installing WebSphere MQ Advanced Message Security on AIX](#)

[Installing WebSphere MQ Advanced Message Security on Linux](#)

[Protecting remote queues with WebSphere MQ Advanced Message Security](#)

[Routing messages protected by WebSphere MQ Advanced Message Security with WebSphere Message Broker](#)

[Using WebSphere MQ Advanced Message Security with WebSphere MQ Managed File Transfer](#)

Applying service to a virtual system instance that contains IBM WebSphere MQ basic parts

Apply service to the IBM WebSphere MQ basic parts in a virtual system instance. You must first load the service that is required onto the appliance. You can control when the appliance applies the service to a virtual system instance. The appliance keeps a snapshot of the instance before service is applied, in case you need to restore it, and it maintains a service history.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- You must have all access to the virtual system instance or be assigned the Appliance administration role with full permissions to carry out these steps.

- Load a new IBM WebSphere MQ virtual image that contains the fix pack or emergency fixes to apply into IBM Workload Deployer or IBM PureApplication System. The virtual image must be for the release level of a IBM WebSphere MQ basic part that is part of the pattern that you deployed. For example, if you added a IBM WebSphere MQ basic part from the IBM WebSphere MQ version 7.0.1.4 virtual image, you can apply the fix-pack in the IBM WebSphere MQ version 7.0.1.8 virtual image. The IBM WebSphere MQ version 7.5.0.0 virtual image does not contain any fix-packs that you can apply to a IBM WebSphere MQ basic part from the IBM WebSphere MQ version 7.0.1.4 virtual image.
- The virtual system instance must be running.

Select the virtual machine instance to apply service to, and choose the fixes to apply. The appliance stops the virtual machine instance, backs it up, applies the fixes, and restarts the instance. If your virtual system instance contains a number of different parts to which different fixes apply, the service procedure applies the fix you selected to the appropriate parts.

1. In the appliance user interface, click **Instances > Virtual Systems**. Click the virtual system instance you want to apply service to in the navigator window.

2. Click **Service**, .

If maintenance is available for any of the parts in the virtual system instance, the **Describe your service request** window opens.

3. Click each option to describe your service request, and then click **OK**.

- Schedule service to apply service immediately or at a later time.
- Select the service level or fixes to apply. You can choose from the alternatives that are available from virtual images that you previously loaded into the appliance.
- Ignore the option to type the user ID and password of the product administrator. The IBM WebSphere MQ basic part does not require these fields.

The appliance runs a script package that is provided with the IBM WebSphere MQ virtual image that contains a fix-pack or emergency fixes. The script identifies the parts to which the fix-pack, or fixes apply, and applies them.

The appliance stops the virtual machines in the image and takes a maintenance snapshot of the virtual system instance. It applies service and restarts the virtual machines. It logs all these actions in its history of the virtual system instance.

To restore the virtual system instance to the maintenance snapshot, click **Restore**. You must restart the virtual system instance after the snapshot is restored.

To update the snapshot, click **Recreate for update**.

Related tasks

[“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14](#)

Add the IBM WebSphere MQ virtual image for IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to the image catalog in IBM Workload Deployer or IBM PureApplication System.

Security

When a IBM WebSphere MQ basic part is added to a virtual system pattern, or when it is deployed, you can customize IBM WebSphere MQ security settings before any users log on.

Default security of the IBM WebSphere MQ basic part

When the IBM WebSphere MQ basic part is first deployed in a IBM WebSphere MQ virtual system instance, the deployment process creates the user mqm with the primary group mqm. No password is assigned, so by default you cannot log on with the mqm user ID. The deployment process runs as the mqm user to create and configure the queue manager.

The deployment process creates two other users when the virtual machine is deployed, `root` and `virtuser`. If these users run the "`sudo -u mqm command`" command, they are authorized to run `command` with the authority of the `mqm` group. The users themselves are not members of the `mqm` group.

Adding authorized users to the IBM WebSphere MQ basic part

The IBM WebSphere MQ basic part has a security property **Authorized users**¹ that you can set. You can define a list of user IDs, separated by spaces. The deployment process creates each user ID, and makes it a member of a primary group with the same name.

The deployment process adds the authorized user IDs to the list of user IDs that can run the command, "`sudo -u mqm command`". The authorized user ID acquires `mqm` group authority for the duration of the command.

Each user, and hence its primary group, is given IBM WebSphere MQ administrative authority; see [Figure 8 on page 26](#).

```
* These commands give user '@authuser@' full administrative access.
SET AUTHREC OBJTYPE(QMGR) PRINCIPAL('@authuser@') AUTHADD(CONNECT,INQ,ALLADM)
SET AUTHREC OBJTYPE(QUEUE) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(TOPIC) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(CHANNEL) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(PROCESS) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(NAMELIST) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(AUTHINFO) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(CLNNTCONN) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(LISTENER) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(SERVICE) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
SET AUTHREC OBJTYPE(COMMINFO) PROFILE('**') PRINCIPAL('@authuser@') AUTHADD(ALLADM,CRT)
* The following commands provide '@authuser@' with administrative access for MQ Explorer.
SET AUTHREC OBJTYPE(QUEUE) PROFILE(SYSTEM.MQEXPLORER.REPLY.MODEL) PRINCIPAL('@authuser@')
AUTHADD(DSP,INQ,GET)
SET AUTHREC OBJTYPE(QUEUE) PROFILE(SYSTEM.ADMIN.COMMAND.QUEUE) PRINCIPAL('@authuser@')
AUTHADD(DSP,INQ,PUT)
```

Figure 8. Administrative authorities granted to authorized users

A user that connects remotely to IBM WebSphere MQ Explorer from any IP address can assume the identity of an authorized user. To prevent this security weakness, the deployment process removes a default channel authentication rule; see [Figure 9 on page 26](#). Removing this rule prevents anyone who knows the identity of an authorized user access to IBM WebSphere MQ Explorer. This change to the IBM WebSphere MQ default configuration also removes remote access to IBM WebSphere MQ Explorer for the authorized users. You can extend permission back to a limited set of TCP/IP addresses; see [“Defining authorized IP addresses for remote administration” on page 27](#).

```
CHLAUTH(SYSTEM.ADMIN.SVRCONN) TYPE(ADDRESSMAP)
DESCR(Default rule to allow MQ Explorer access)
CUSTOM( ) ADDRESS(*)
MCAUSER( ) USERSRC(CHANNEL)
WARN(NO) ALTDAT(2012-07-05)
ALTTIME(12.08.09)
```

Figure 9. Default channel authentication rule to allow remote access to IBM WebSphere MQ Explorer

The deployment process gives the authorized user IDs permission to remotely connect to the queue manager. It adds channel authentication records for the server-connection channels `SYSTEM.DEF.SVRCONN` and `SYSTEM.ADMIN.SVRCONN`; see [Figure 10 on page 26](#). It limits the authorization to the local TCP/IP address `127.0.0.1`, and to an authenticated authorized user.

```
* This command gives @authuser@ authority to connect as a client from localhost to run MQ Explorer.
SET CHLAUTH('SYSTEM.ADMIN.SVRCONN') TYPE(USERMAP) USERSRC(CHANNEL) ADDRESS('127.0.0.1') +
CLNTUSER('@authuser@') DESCR('Only @authuser@ locally') ACTION(ADD)
* This command gives @authuser@ authority to connect as am mqi client from localhost.
SET CHLAUTH('SYSTEM.DEF.SVRCONN') TYPE(USERMAP) USERSRC(CHANNEL) ADDRESS('127.0.0.1') +
CLNTUSER('@authuser@') DESCR('Only @authuser@ locally') ACTION(ADD)
```

Figure 10. Channel authorities granted to authorized users from the TCP/IP address `127.0.0.1`

Defining authorized IP addresses for remote administration

The IBM WebSphere MQ basic part has a security property **Authorized IP addresses**¹ that you can set. Define a single IP address, or a group of IP addresses; see [Generic IP addresses](#).

The deployment process defines a server-connection channel authorization for the selected IP addresses; see [Figure 11 on page 27](#).

The authorization allows any user to start IBM WebSphere MQ Explorer from the authorized IP addresses with administrator authority. For the server-connection channel to start IBM WebSphere MQ Explorer, you must also define an authorized user.

Important: The authorization weakens the security of the IBM WebSphere MQ virtual system instance. Any user can administer IBM WebSphere MQ from the IP addresses you define. To strengthen security, you must limit access to those IP addresses to the authorized users.

```
* This command gives anyone authority to connect as a client from specific IP addresses to run
MQ Explorer.
DEFINE CHANNEL(SYSTEM.ADMIN.SVRCONN) CHLTYPE(SVRCONN) REPLACE
SET CHLAUTH('SYSTEM.ADMIN.SVRCONN') TYPE(ADDRESSMAP) USERSRC(MAP) ADDRESS('@ipaddress@') +
MCAUSER('@authuser@') DESCR('Always @authuser@ from specified IP') ACTION(ADD)
```

Figure 11. Channel authorities granted to selected IP addresses

Example: Deploy a local queue with access permissions for an authorized user

The script creates a local queue and sets up permissions for an authorized user to put and get messages to and from the queue.

Use the queue to verify a deployed IBM WebSphere MQ basic part from a remote IBM WebSphere MQ MQI client; see [“Verifying deployed IBM WebSphere MQ parts from a remote IBM WebSphere MQ MQI client.” on page 44](#).

1. Create a pattern; see [“Creating a pattern” on page 28](#).
2. Add the IBM WebSphere MQ basic part to a pattern; see [“Adding the IBM WebSphere MQ basic part to a pattern” on page 30](#).
3. Add the script package WMQ: Run MQSC Scripts to the IBM WebSphere MQ basic part; see [“Adding MQSC commands to a pattern” on page 31](#).
4. Define at least one authorized user in the IBM WebSphere MQ basic part; see [Table 2 on page 51 in “IBM WebSphere MQ basic part” on page 50](#).

The task describes how to add an MQSC command script, `verify.mqsc`, to a IBM WebSphere MQ basic part. The `verify.mqsc` command script creates a local queue and sets permissions for an authorized user to put and get messages to and from the queue. It creates a channel authorization rule to allow an authorized user to connect to a IBM WebSphere MQ basic part from a remote IBM WebSphere MQ MQI client with the `SYSTEM.DEF.SVRCONN` server-connection channel.

In step [“1” on page 27](#) make the following changes to the `verify.mqsc` command script,

1. Replace `authUser` with one of the authorized users you defined in the IBM WebSphere MQ basic part.
2. Replace `qName` with a name for the local queue that is to be created.
3. Replace `ipAddress` with a specific or generic IP address; see [Generic IP addresses](#).

Important: The authorization weakens the security of the IBM WebSphere MQ virtual system instance. Any user can administer IBM WebSphere MQ from the IP addresses you define. To strengthen security, you must limit access to those IP addresses to the authorized users.

1. Create the MQSC command file, `verify.mqsc`

```

* This command creates qName to save verification messages
DEFINE QLOCAL(qName)
* This command gives authUser authority to use the qName for verification
SET AUTHREC PROFILE('qName') OBJTYPE(Queue) PRINCIPAL('authUser') AUTHADD(BROWSE, GET, PUT, INQ)
* This command gives anyone authority to connect as a client from specific IP addresses using the
default server-connection channel.
SET CHLAUTH('SYSTEM.DEF.SVRCONN') TYPE(ADDRESSMAP) USERSRC(MAP) ADDRESS('ipAddress') +
MCAUSER('authUser') DESCR('Always connect as authUser from ipAddress') ACTION(ADD)

```

Figure 12. *verify.mqsc*

2. In step “2” on page 32 in “Adding MQSC commands to a pattern” on page 31, add the file *verify.mqsc*, and save the modified script package file as *WMQVerifyMQSC.zip*.
3. In step “3” on page 33 in “Adding MQSC commands to a pattern” on page 31, set the script to run when the pattern is first deployed.
1. Deploy the pattern; see “Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37.
2. Verify the deployment by running a IBM WebSphere MQ MQI client from a workstation that is connected on an authorized IP address; see “Verifying deployed IBM WebSphere MQ parts from a remote IBM WebSphere MQ MQI client.” on page 44.

Related tasks

[“Adding MQSC commands to a pattern” on page 31](#)

You can add one or more files of IBM WebSphere MQ commands to customize a pattern that contains IBM WebSphere MQ basic parts. The commands run when the pattern is deployed as a virtual system, when the virtual system is deleted, or when you choose to run the commands.

[“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#)

Deploy a pattern that contains one or more IBM WebSphere MQ basic parts to a cloud with IBM Workload Deployer or IBM PureApplication System.

[“Verifying deployed IBM WebSphere MQ parts from a remote IBM WebSphere MQ MQI client.” on page 44](#)

Verify the deployment of IBM WebSphere MQ parts in a virtual system pattern by running the “put” and “get” sample MQI client programs. Run the programs from inside or outside of the cloud where the parts are deployed.

[Preparing and running the sample programs](#)

Configuration

Tasks to help you configure the deployment of IBM WebSphere MQ Hypervisor editions to an appliance or to VMware ESX hypervisor.

The tasks assist you with the installation and deployment of virtual machines and virtual system patterns that include IBM WebSphere MQ. To configure IBM WebSphere MQ, see [Configuring WebSphere MQ](#)

Creating a pattern

Create a virtual system pattern.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Check that you have either `Create new patterns` or `Create new catalog content` permission.


From the appliance user interface, click **System > Users** and in the list of users, click your user name.

Complete the following steps to create a pattern.

1. From the appliance user interface, open the **Virtual Systems Patterns** window.
 - From the menu bar, click **Patterns > Virtual Systems**.

- From the **Welcome** page, in the section "**Working with virtual systems**", click **Create a virtual system pattern**.

The **Virtual Systems Patterns** window opens.

2. Click the new icon , next to the "**Virtual Systems Patterns**" window title, to create a virtual system pattern.

The "**Describe the pattern you want to add**" window opens.

- a) Type a unique name into the field **Name**.
 - b) Type a description into the field **Description**.
3. Click **OK** to close the window.
 4. Click **Done Editing** to finish editing the pattern.

The pattern is saved.

Configure the pattern.

Related concepts

["Parts and patterns" on page 8](#)

IBM WebSphere MQ Hypervisor editions contain an IBM WebSphere MQ basic part which you include in virtual system patterns. Virtual system patterns are made up of parts and parts have properties. Each part represents a single virtual machine. A *pattern* provides a topology definition for repeatable deployment that can be shared. Patterns describe the function provided by each virtual machine in a virtual system. Each function is identified as a part in the pattern

Related tasks

["Adding the IBM WebSphere MQ basic part to a pattern" on page 30](#)

Add the IBM WebSphere MQ basic part to an IBM WebSphere MQ pattern, and edit its properties to start configuring the pattern.

["Deploying a pattern that contains an IBM WebSphere MQ basic part" on page 37](#)

Deploy a pattern that contains one or more IBM WebSphere MQ basic parts to a cloud with IBM Workload Deployer or IBM PureApplication System.

Copying a pattern

Make a new pattern by copying and modifying an existing pattern. Copy a read-only pattern, and modify the copy. Copies are called clones. When you clone a pattern you can associate it with a different version of the IBM WebSphere MQ virtual image.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Check that you have either **Create new patterns** or **Create new catalog content** permission.

From the appliance user interface, click **System > Users** and in the list of users, click your user name.

Follow these steps to copy a pattern.

1. From the appliance user interface, open the **Virtual Systems Patterns** window.
 - From the menu bar, click **Patterns > Virtual Systems**.
 - From the **Welcome** page, in the section "**Working with virtual systems**", click **Create a virtual system pattern**.

The **Virtual Systems Patterns** window opens.

2. Click the pattern you want to copy in the list of Virtual Systems Patterns.

The canvas for the selected pattern opens.

3. Click the clone icon .

The clone icon is in the menu bar.

The "**Describe the pattern you want to add**" window opens.

- a) Type a unique name into the field **Name**.
 - b) Type a description into the field **Description**.
 - c) Click the virtual image to associate the pattern with.
4. Click **OK** to close the window.
 5. Click **Done Editing** to finish editing the pattern.

The pattern is saved.

Configure the pattern.

Related concepts

[“Parts and patterns” on page 8](#)

IBM WebSphere MQ Hypervisor editions contain a IBM WebSphere MQ basic part which you include in virtual system patterns. Virtual system patterns are made up of parts and parts have properties. Each part represents a single virtual machine. A *pattern* provides a topology definition for repeatable deployment that can be shared. Patterns describe the function provided by each virtual machine in a virtual system. Each function is identified as a part in the pattern

Related tasks

[“Adding the IBM WebSphere MQ basic part to a pattern” on page 30](#)

Add the IBM WebSphere MQ basic part to an IBM WebSphere MQ pattern, and edit its properties to start configuring the pattern.

Related reference

[“IBM WebSphere MQ virtual system pattern” on page 59](#)

Adding the IBM WebSphere MQ basic part to a pattern

Add the IBM WebSphere MQ basic part to an IBM WebSphere MQ pattern, and edit its properties to start configuring the pattern.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Add the IBM WebSphere MQ basic part to the appliance virtual image catalog; see [“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14](#).
- Create a pattern; see [“Creating a pattern” on page 28](#).
- Review the information in [“IBM WebSphere MQ basic part” on page 50](#) to understand the part.
- Check that you have either **Create new patterns** or **Create new catalog content** permission.


From the appliance user interface, click **System > Users** and in the list of users, click your user name.

Complete the following steps to start configuring a pattern containing a IBM WebSphere MQ basic part:

1. From the appliance user interface, open the **Virtual Systems Patterns** window.
 - From the menu bar, click **Patterns > Virtual Systems**.
 - From the **Welcome** page, in the section "**Working with virtual systems**", click **Create a virtual system pattern**.

The **Virtual Systems Patterns** window opens.

2. Open a pattern that contains a IBM WebSphere MQ basic part for editing.
 - a) Find the pattern in the list of Virtual Systems Patterns.

A draft icon  that follows the pattern name signifies it is editable.


The pattern opens in the properties window.

- b) Click the edit icon  in the menu bar.

The pattern editor opens, with lists of parts, scripts, and add-ons in the navigator, and an editing canvas.


3. Click the **Parts** tab in the navigator

The list of available parts is displayed.

4. Drag the IBM WebSphere MQ basic part into the pattern.
 - a) Find the IBM WebSphere MQ basic parts in the list of parts
 - b) Drag the correct part for the target platform onto the canvas
5. Click the drop-down list on the part to select a release level.
6. Click the properties icon  on the part to edit its properties.

For the properties of the part, see [“IBM WebSphere MQ basic part” on page 50](#).

For the two passwords that are required, type passwords that meet the requirements of your organization. You are creating the initial passwords for `root` and `virtuser`.

7. Click the padlock icon  to lock or unlock the property. You can modify unlocked properties when you deploy the pattern.
8. Click **OK** to close the window.
9. Click **Done Editing** to finish editing the pattern.

The pattern is saved.

Add more parts or scripts to the pattern, including multiple instances of the IBM WebSphere MQ basic part, and then deploy the pattern.

Related concepts

[“Parts and patterns” on page 8](#)

IBM WebSphere MQ Hypervisor editions contain a IBM WebSphere MQ basic part which you include in virtual system patterns. Virtual system patterns are made up of parts and parts have properties. Each part represents a single virtual machine. A *pattern* provides a topology definition for repeatable deployment that can be shared. Patterns describe the function provided by each virtual machine in a virtual system. Each function is identified as a part in the pattern

Related tasks

[“Creating a pattern” on page 28](#)

Create a virtual system pattern.

[“Adding MQSC commands to a pattern” on page 31](#)

You can add one or more files of IBM WebSphere MQ commands to customize a pattern that contains IBM WebSphere MQ basic parts. The commands run when the pattern is deployed as a virtual system, when the virtual system is deleted, or when you choose to run the commands.

[“Adding a queue manager to a cluster” on page 33](#)

The `WMQ: Add to Queue Manager Cluster` cluster script adds a queue manager to a cluster.

[“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#)

Deploy a pattern that contains one or more IBM WebSphere MQ basic parts to a cloud with IBM Workload Deployer or IBM PureApplication System.

Related reference

[“IBM WebSphere MQ basic part” on page 50](#)

Adding MQSC commands to a pattern

You can add one or more files of IBM WebSphere MQ commands to customize a pattern that contains IBM WebSphere MQ basic parts. The commands run when the pattern is deployed as a virtual system, when the virtual system is deleted, or when you choose to run the commands.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.

- Check that you have Create new catalog content permission.

From the appliance user interface, click **System > Users** and in the list of users, click your user name.

- Add the IBM WebSphere MQ virtual image to the appliance; see [“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14.](#)
- Add a IBM WebSphere MQ basic part to a pattern; see [“Adding the IBM WebSphere MQ basic part to a pattern” on page 30.](#)
- Add the WMQ: Run MQSC Scripts script package to the appliance; see [“Adding an IBM WebSphere MQ script package to an appliance” on page 16.](#)
- Prepare one or more files of MQSC commands to customize a IBM WebSphere MQ basic part in the pattern.

Add one or more files of IBM WebSphere MQ commands to a copy of the "WMQ: Run MQSC Scripts" script package. Attach the script package to the IBM WebSphere MQ basic part in a pattern.

1. Create a script package.

In this task, you add file of MQSC commands to run to the script package.


- a) From the appliance user interface, open the **Script Packages** window.

From the menu bar, click **Catalog > Script Packages**.

The **Script Packages** window opens.

- b) In the list of **Script Packages**, click **WMQ: Run MQSC Scripts**.

The **WMQ: Run MQSC Scripts** properties window opens

- c) Click the clone icon .


The clone icon is in the menu bar.

"**A new script package**" window opens.

- d) Type a unique name for the copy of the script package in the **Name** field and click **OK**.

2. Add one or more files of MQSC commands to the script package.

- a) In the **Script package files** section of the canvas, under "The script package is in

WMQExecuteMQSC.zip", click  Download.

- b) Save WMQExecuteMQSC.zip on your workstation.

Tip: Change the name of WMQExecuteMQSC.zip when you download it to avoid confusion with other script packages you download. For example, give it the name of the script package.

Restriction: If you add multiple script packages that contain MQSC command files to a IBM WebSphere MQ basic part, you must modify the cbscript.json file in each script package.

Change the temporary directory in the line, "location": "/tmp/mq/mqsc" to "location": "/tmp/mq/dir", where *dir* is the name you choose for a new directory. The appliance creates the new temporary directory, which contains the MQSC command files from this script package. It then runs all the commands from the command files in the directory. If you do not create a different temporary directory for each script package, the command files accumulate in the single directory. Each time the appliance processes a script package, it runs commands from all the accumulated files in the single temporary directory, which might give unexpected results.

- c) Add one or more MQSC files with the extension .mqsc to WMQExecuteMQSC.zip.

With some compressed files tools you, can add files directly to WMQExecuteMQSC.zip. With other tools, extract the files from WMQExecuteMQSC.zip, and re-create WMQExecuteMQSC.zip, adding the MQSC file.


- d) In the **Script package files** section of the canvas, in the **Browse...** field, click or type the path to the modified WMQExecuteMQSC.zip file. Then click **Upload**.

If you renamed WMQExecuteMQSC.zip, you can upload the file with its new name.

3. Choose when to run the script by selecting an option in the **Executes** drop-down list on the canvas.
4. From the appliance user interface, open the **Virtual Systems Patterns** window.
 - From the menu bar, click **Patterns > Virtual Systems**.
 - From the **Welcome** page, in the section "**Working with virtual systems**", click **Create a virtual system pattern**.

The **Virtual Systems Patterns** window opens.

5. Open a pattern that contains a IBM WebSphere MQ basic part for editing.
 - a) Find the pattern in the list of Virtual Systems Patterns.

A draft icon  that follows the pattern name signifies it is editable.

The pattern opens in the properties window.

- b) Click the edit icon  in the menu bar.

The pattern editor opens, with lists of parts, scripts, and add-ons in the navigator, and an editing canvas.

6. Click the **Scripts** tab in the navigator

Available scripts are listed.

7. Drag your script package onto the part, which is on the canvas.
8. Click **OK** to close the window.
9. Click **Done Editing** to finish editing the pattern.

The pattern is saved.

1. Do the example, [“Example: Deploy a local queue with access permissions for an authorized user” on page 27](#).
2. Add more parts or scripts to the pattern, including multiple instances of the IBM WebSphere MQ basic part, and then deploy the pattern.

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

[“Adding an IBM WebSphere MQ script package to an appliance” on page 16](#)

Add an IBM WebSphere MQ script package for IBM WebSphere MQ Hypervisor Edition to the script catalog in IBM Workload Deployer or IBM PureApplication System.

[“Example: Deploy a local queue with access permissions for an authorized user” on page 27](#)

Related reference

[“IBM WebSphere MQ Hypervisor Edition script packages” on page 60](#)

Adding a queue manager to a cluster

The WMQ: Add to Queue Manager Cluster cluster script adds a queue manager to a cluster.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Deploy a pattern or a real server that contains a full repository first, so that you know the IP address of the full repository.
- Add the IBM WebSphere MQ basic part to a pattern; see [“Adding the IBM WebSphere MQ basic part to a pattern” on page 30](#).
- Check that you have either **Create new patterns** or **Create new catalog content** permission.

From the appliance user interface, click **System > Users** and in the list of users, click your user name.

- Add the script package, **WMQ: Add to Queue Manager Cluster** to the script catalog in the appliance; see [“Adding an IBM WebSphere MQ script package to an appliance”](#) on page 16.
- Review the information in [Clustering: Best practices](#) to understand queue manager clustering.

The **WMQ: Add to Queue Manager Cluster** script connects the queue manager in the part to a full repository queue manager that is already running elsewhere. The script is complete, but limited in what it does for you. Examine the script as an example of configuring the IBM WebSphere MQ basic part.


One of the limitations is you cannot link the queue manager in the part to a full repository in the same virtual system pattern. The problem is you do not know the IP address that is allocated to a full repository that is a different part in the pattern. Nor can you configure more advanced cluster properties. For example you cannot make the queue manager a member of two clusters, and you cannot set cluster channel properties.

Complete these steps to add a queue manager to a cluster.

1. From the appliance user interface, open the **Virtual Systems Patterns** window.
 - From the menu bar, click **Patterns > Virtual Systems**.
 - From the **Welcome** page, in the section **"Working with virtual systems"**, click **Create a virtual system pattern**.

The **Virtual Systems Patterns** window opens.

2. Open a pattern that contains a IBM WebSphere MQ basic part for editing.
 - a) Find the pattern in the list of Virtual Systems Patterns.

A draft icon  that follows the pattern name signifies it is editable.


The pattern opens in the properties window.

- b) Click the edit icon  in the menu bar.

The pattern editor opens, with lists of parts, scripts, and add-ons in the navigator, and an editing canvas.


3. Click the **Scripts** tab in the navigator

Available scripts are listed.

4. Drag the **WMQ: Add to Queue Manager Cluster** script package onto the part, which is on the canvas.
5. Click the parameters icon  in the script package to edit the script parameters.
6. Provide values for the four properties

MQ_ADD_TO_CLUSTER_NAME
MQ_ADD_TO_FULL_REPOSITORY_QMGR_NAME
MQ_ADD_TO_FULL_REPOSITORY_CONNNNAME
MQ_ADD_TO_FULL_REPOSITORY_PORT

For parameter descriptions; see [Table 10 on page 60](#) in [“IBM WebSphere MQ Hypervisor Edition script packages”](#) on page 60.

7. Click the padlock icon  to lock or unlock the property. You can modify unlocked properties when you deploy the pattern.
8. Click **OK** to close the window.
9. Click **Done Editing** to finish editing the pattern.

The pattern is saved.

- Add more parts or scripts to the pattern, including multiple instances of the IBM WebSphere MQ basic part, and then deploy the pattern.

- Add the script `WMQ: Remove from Queue Manager Cluster` to the part; see [“Removing a queue manager from a cluster”](#) on page 35.

Related tasks

[“Adding an IBM WebSphere MQ script package to an appliance”](#) on page 16

Add an IBM WebSphere MQ script package for IBM WebSphere MQ Hypervisor Edition to the script catalog in IBM Workload Deployer or IBM PureApplication System.

[“Adding the IBM WebSphere MQ basic part to a pattern”](#) on page 30

Add the IBM WebSphere MQ basic part to an IBM WebSphere MQ pattern, and edit its properties to start configuring the pattern.

[“Removing a queue manager from a cluster”](#) on page 35

The `WMQ: Remove from Queue Manager Cluster` script cleanly removes a queue manager from a cluster. It removes the queue manager from the cluster and removes its entries from the full repositories in the cluster.

Related reference

[“IBM WebSphere MQ Hypervisor Edition script packages”](#) on page 60

Removing a queue manager from a cluster

The `WMQ: Remove from Queue Manager Cluster` script cleanly removes a queue manager from a cluster. It removes the queue manager from the cluster and removes its entries from the full repositories in the cluster.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- You must know the queue manager cluster name and the name of the full repository queue manager that a IBM WebSphere MQ basic part is connected to.
- You added the script package, `WMQ: Add to Queue Manager Cluster` to the IBM WebSphere MQ basic part; see [“Adding a queue manager to a cluster”](#) on page 33.
- Add the script package, `WMQ: Remove from Queue Manager Cluster` to the script catalog in the appliance; see [“Adding an IBM WebSphere MQ script package to an appliance”](#) on page 16.
- Check the `WMQ: Remove from Queue Manager Cluster` script is set to run either on demand, or when you delete the virtual instance that contains a IBM WebSphere MQ basic part that contains the script.

1. From the appliance user interface, open the **Script Packages** window.

From the menu bar, click **Catalog > Script Packages**.

The **Script Packages** window opens.

2. In the list of **Script Packages**, click **Remove from Queue Manager Cluster**.
3. Check what value the **Executes:** property is set to. It is typically set to **at virtual system deletion**.

If you created patterns that embed the `WMQ: Remove from Queue Manager Cluster` script, and you want to alter the script, consider altering a cloned script, rather than altering this script.

Add the `WMQ: Remove from Queue Manager Cluster` script to a IBM WebSphere MQ basic part. The script removes the queue manager in the part from the cluster, and from the full repository queue managers that maintain the cluster configuration. Typically you would set the script to run if the virtual system instance that contains a IBM WebSphere MQ basic part that is part of a cluster is deleted.


When the script runs, it sends messages to one of the full repositories for the cluster. The full repository and the virtual system instance that contains the IBM WebSphere MQ basic part that includes the `WMQ: Remove from Queue Manager Cluster` script must be running when the script is run. For example, you can set the script to run when the virtual system instance that contains it is deleted. If you do so, the virtual system instance must be running when you delete it. If the virtual instance is stopped, start it first, before deleting it and thus removing queue managers from clusters.

As with all IBM WebSphere MQ commands, the commands run asynchronously. The time cluster commands take to complete depends on the network, whether both repositories are running, and how busy the network and repositories are.

1. From the appliance user interface, open the **Virtual Systems Patterns** window.
 - From the menu bar, click **Patterns > Virtual Systems**.
 - From the **Welcome** page, in the section "**Working with virtual systems**", click **Create a virtual system pattern**.

The **Virtual Systems Patterns** window opens.

2. Open a pattern that contains a IBM WebSphere MQ basic part for editing.
 - a) Find the pattern in the list of Virtual Systems Patterns.

A draft icon  that follows the pattern name signifies it is editable.


The pattern opens in the properties window.

- b) Click the edit icon  in the menu bar.

The pattern editor opens, with lists of parts, scripts, and add-ons in the navigator, and an editing canvas.

3. Drag the WMQ: Remove from Queue Manager Cluster script onto the part that you want to add it to. The parts are on the canvas.


If the part does not already have a WMQ: Add to Queue Manager Cluster script, are you editing the correct part? It is unusual, but not incorrect, to have a WMQ: Remove from Queue Manager Cluster script without a WMQ: Add to Queue Manager Cluster script in a IBM WebSphere MQ basic part.

4. Click the parameters icon  in the script package to edit the script parameters.
5. Provide values for the two properties

MQ_REMOVE_FROM_CLUSTER_NAME

MQ_REMOVE_FROM_FULL_REPOSITORY_QMGR_NAME

For parameter descriptions; see [Table 11 on page 60](#) in "[IBM WebSphere MQ Hypervisor Edition script packages](#)" on page 60.

6. Click the padlock icon  to lock or unlock the property. You can modify unlocked properties when you deploy the pattern.
 7. Click **OK** to close the window.
 8. Click **Done Editing** to finish editing the pattern.
- Add more parts or scripts to the pattern, including multiple instances of the IBM WebSphere MQ basic part, and then deploy the pattern.

Related tasks

["Adding an IBM WebSphere MQ script package to an appliance" on page 16](#)

Add an IBM WebSphere MQ script package for IBM WebSphere MQ Hypervisor Edition to the script catalog in IBM Workload Deployer or IBM PureApplication System.

["Adding the IBM WebSphere MQ basic part to a pattern" on page 30](#)

Add the IBM WebSphere MQ basic part to an IBM WebSphere MQ pattern, and edit its properties to start configuring the pattern.

["Adding a queue manager to a cluster" on page 33](#)

The WMQ: Add to Queue Manager Cluster cluster script adds a queue manager to a cluster.

[Removing a queue manager from a cluster](#)

Related reference

["IBM WebSphere MQ Hypervisor Edition script packages" on page 60](#)

Deploying a pattern that contains an IBM WebSphere MQ basic part

Deploy a pattern that contains one or more IBM WebSphere MQ basic parts to a cloud with IBM Workload Deployer or IBM PureApplication System.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Configure a pattern that contains one or more IBM WebSphere MQ basic parts; see [“Adding the IBM WebSphere MQ basic part to a pattern” on page 30](#).
- Check that you have `Deploy patterns in the cloud` permission.

From the appliance user interface, click **System > Users** and in the list of users, click your user name

- Check that the cloud you are deploying to has correct and sufficient resources to run the virtual system.

Select a pattern that contains one or more IBM WebSphere MQ basic parts to deploy it to a cloud. Provide any missing properties that are required.

1. From the appliance user interface, open the **Virtual Systems Patterns** window.

- From the menu bar, click **Patterns > Virtual Systems**.
- From the **Welcome** page, in the section **"Working with virtual systems"**, click **Create a virtual system pattern**.

The **Virtual Systems Patterns** window opens.

2. Click the pattern you want to deploy in the list of Virtual Systems Patterns.

The canvas for the selected pattern opens.

3. Click the icon, , to deploy the pattern in the menu bar.

The window **"Describe the virtual system you want to deploy"** opens.

All options must have this check mark, , before you can deploy the pattern.

- a) Type a unique name in the **Virtual system name** field to identify the deployed instance of the pattern.
- b) Click the other lines to modify the deployment settings, and provide missing parameters.

Choose Environment

Choose an IP version, and provide either a cloud group to deploy to, or select an environment profile that controls the selection of the deployment environment.

Schedule deployment

Schedule the start and duration of the deployment.

Configure virtual parts

Click **Configure virtual parts** to modify the properties of the parts that are deployed, and provide values for properties that are required but have no value.

The modified properties are set in the virtual system pattern instance before it is deployed. They are not saved in the pattern.

4. Click **OK** to deploy the pattern.

The appliance deploys the virtual system pattern as a virtual system instance in the cloud.

Related concepts

[Environment profiles overview](#)

[“Configuration” on page 28](#)

Tasks to help you configure the deployment of IBM WebSphere MQ Hypervisor editions to an appliance or to VMware ESX hypervisor.

Related tasks

[Deploying virtual system patterns](#)

[“Adding the IBM WebSphere MQ basic part to a pattern” on page 30](#)

Add the IBM WebSphere MQ basic part to an IBM WebSphere MQ pattern, and edit its properties to start configuring the pattern.

[“Creating a pattern” on page 28](#)

Create a virtual system pattern.

Related reference

[CloudGroup](#)

[IP group command-line interface reference](#)

Deploying an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor

Deploy an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor for the first time.

Load the IBM WebSphere MQ virtual image into the VMware ESX hypervisor data store; see [“Loading the IBM WebSphere MQ virtual image for VMware ESX hypervisor” on page 20](#).

Follow these steps to deploy a IBM WebSphere MQ virtual image directly to VMware ESX hypervisor for the first time.

1. From the VMware vSphere Client, select the **Summary** tab. In the **Datastore** window, right-click a data store in which you stored the virtual image, and click **Browse Datastore**.
2. Configure the network settings of the IBM WebSphere MQ virtual image.
 - a) In the **Inventory** window, right-click on the virtual image, and click **Edit Settings**.
 - b) In the **"Hardware"** window, click **Network Adapter 1**.
 - c) In the **"Network Connection"** window, click the network to which the virtual machine connects to from the Network label pull-down menu.
 - d) Click **OK**.
3. In the **Inventory** window, right-click on the virtual image, click **Power On**.

If you moved or copied the virtual image, a pop-up window appears. Click **I copied it > OK**.

4. Log in to the operating system with the default root credentials.

The user name is `root` and the password is `password`.

The **"Language selection"** window opens.

5. Choose a language, then click **OK** and press **Enter**.

The **"License"** window opens.

6. To proceed, you must review and accept all the license agreements for Red Hat Enterprise Linux and IBM WebSphere MQ.

The **"Network"** window opens.

7. Choose a network protocol.

The virtual machine image supports both static addressing and Dynamic Host Configuration Protocol (DHCP) addressing. If you plan to use multiple virtual machines, configure the network between the virtual machines.

- If you are using DHCP, enter values for the **Host name** and **Domain** fields.
- If you are using the static protocol, enter values for the **IP address**, **Subnet mask**, **Default gateway**, and **DNS server** fields. The value for the DNS server must be an IP address.

- a) Click **OK** and press **Enter**.

The network confirmation window opens.

- b) Click **Yes** and press **Enter**.

The **Password Configuration** window opens.

8. Type a password for the root user ID in the **Password**, and **Password confirmation** fields and press enter.

You must change the password from the default value.

9. Type a password for the virtuser user ID in the **Password**, and **Password confirmation** fields and press enter.

The virtuser user ID is the default non-root user ID for the installation.

The system boots up. The values you entered are saved and you do not need to enter the values again.

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

[“Loading the IBM WebSphere MQ virtual image for VMware ESX hypervisor” on page 20](#)

Save a copy of the IBM WebSphere MQ virtual image for VMware ESX hypervisor in the VMware ESX hypervisor data store.

Related reference

[“The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image for VMware ESX hypervisor” on page 66](#)

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image provides the operating system and product binary files that are required to create virtual system instances that run in VMware ESX hypervisor.

Verifying a deployed IBM WebSphere MQ basic part from an SSH terminal emulator

Verify a IBM WebSphere MQ basic part locally, without authorizing remote users to the virtuser user ID. Test the deployment from an SSH terminal emulator.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Deploy the pattern successfully; see [“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#).
- Open an SSH terminal emulator window on the client workstation you are using to connect to virtual machine in the virtual system instance.
 - On UNIX and Linux, run the **ssh** command.
 - On Windows, obtain an SSH terminal emulator such as **PuTTY**. The **PuTTY** website is [PuTTY: A Free Telnet/SSH Client](#). When a connection is established with the deployed virtual machine, the steps in the task are the same.

The browser-based SSH emulator provided with IBM Workload Deployer and IBM PureApplication System is suitable only for short commands.

Verify the deployment of a IBM WebSphere MQ basic part by running the "put" and "get" sample programs on the virtual machine where the IBM WebSphere MQ basic part is deployed.

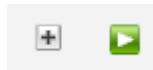
The user ID virtuser is not a member of the mqm group. The example demonstrates the use of the **sudo** command to make virtuser a temporary member of the mqm group for the duration of the command. The example also sets up the minimum set of permissions for virtuser to run the "put" and "get" sample programs, and then removes them. As an alternative, the example also shows running the **sudo** command to run the "put" and "get" sample programs.

1. Obtain the host name or IP address of the virtual machine that contains a deployed IBM WebSphere MQ basic part.
 - a) In the appliance user interface, click **Instances > Virtual Systems**. Click the virtual system instance in the navigator.

The property window for the virtual system instance opens.

- b) In the property window, click the plus sign next to **Virtual machines** to expand the list of virtual machines.

A list of the virtual machines in the system instance opens. Each virtual machine has a plus sign and



an icon to show that it is running:

- c) Click the plus sign next to the virtual machine you want to connect to.

The properties of the virtual machine are listed.

The hostname and IP address is listed next to each **Network interface** property in the list of **Hardware and network** properties.

2. Connect the SSH terminal emulator you are running to the virtual machine.

- On UNIX and Linux:

- a. In a command shell window type `ssh virtuser@hostname/IP address`.

The system responds with a password prompt.

Note: The first time you connect an SSH session, you must respond to the host authentication prompt. To continue, respond yes.

- b. Type the password of the `virtuser` user ID.

The system responds with a list of some key properties of the IBM WebSphere MQ and the name of the queue manager.

- On Windows:

- a. Open the **PuTTY** configuration window by running the **PuTTY** command.

- b. In the Session tab, type the **Host Name (or IP address)** and click **Open**. Leave the **Port** as 22, and the **Connection type** as SSH.

The system responds by opening a window and the prompt, login as:

- c. Type `virtuser`.

The system responds with a password prompt.

Note: The first time you connect an SSH session, you must respond to the host authentication prompt. To continue, respond yes.

- d. Type the password of the `virtuser` user ID.

The system responds with a list of some key properties of the IBM WebSphere MQ and the name of the queue manager.

3. Authorize the user ID `virtuser` to run the verification application.

- a) Give `virtuser` permission to connect to the queue manager.

```
sudo -u mqm setmqaut -m QmgrName -t qmgr -p virtuser +connect
```

The system responds with:

```
[sudo] password for virtuser:
```

Type the password for the user ID `virtuser`.

The system responds with:

The `setmqaut` command completed successfully.

- b) Give `virtuser` permission to use put, get, and inquire MQI calls against the queue used for verification.

```
sudo -u mqm setmqaut -m QmgrName -n SYSTEM.DEFAULT.LOCAL.QUEUE -t queue -p virtuser +put +get +inq
```


The system responds with:

The setmqaut command completed successfully.

4. Put a message on SYSTEM.DEFAULT.LOCAL.QUEUE.

```
/opt/mqm/samp/bin/amqspout SYSTEM.DEFAULT.LOCAL.QUEUE QmgrName
```

The system response:

```
Sample AMQSPUT0 start  
target queue is qName
```

5. Type in a message followed by two new lines

```
Hello world
```

The system response:

```
Sample AMQSPUT0 end
```

6. Get the message from SYSTEM.DEFAULT.LOCAL.QUEUE.

```
/opt/mqm/samp/bin/amqsget SYSTEM.DEFAULT.LOCAL.QUEUE QmgrName
```

The system response:

```
Sample AMQSGET0 start  
message <Hello world>  
15 seconds delay  
no more messages  
Sample AMQSGET0 end
```

You demonstrated the queue manager is working correctly.

You have completed the verification task. You might choose to remove the changes you made, and try the **sudo** command to run the "put" and "get" sample programs.

1. Remove the authorizations you set by running the following commands:

```
sudo -u mqm setmqaut -m QmgrName -t qmgr -p virtuser -connect  
sudo -u mqm dspmqaut -m QmgrName -t qmgr -p virtuser  
sudo -u mqm setmqaut -m QmgrName -n SYSTEM.DEFAULT.LOCAL.QUEUE -t queue -p virtuser -put  
-get -inq  
sudo -u mqm dspmqaut -m QmgrName -n SYSTEM.DEFAULT.LOCAL.QUEUE -t queue -p virtuser
```

2. Run the verification again with the **sudo** command:

- a. Put a message onto SYSTEM.DEFAULT.LOCAL.QUEUE

```
sudo -u mqm /opt/mqm/samp/bin/amqspout SYSTEM.DEFAULT.LOCAL.QUEUE QmgrName
```

Type a message followed by two new lines.

- b. Get the message from SYSTEM.DEFAULT.LOCAL.QUEUE

```
sudo -u mqm /opt/mqm/samp/bin/amqsget SYSTEM.DEFAULT.LOCAL.QUEUE QmgrName
```

Wait for 15 seconds for the program to finish.

Verifying a deployed IBM WebSphere MQ basic part from a VNC session

Verify a IBM WebSphere MQ basic part locally, without authorizing remote users to the virtuser user ID. Test the deployment from a VNC session.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.

- Set the **VNC** property in the IBM WebSphere MQ basic part to its default value, `true`, before you deploy the pattern.
- Remember the password to the `virtuser` user ID, which is set in the IBM WebSphere MQ basic part properties.

Restriction: VNC requires that a password must have at least six characters, and it treats only the first eight characters of a password as significant. If the password has fewer than six characters, VNC appends the ASCII "0" character to the password to make it six characters long; for example: "pass" becomes "pass00".

- Deploy the pattern successfully; see [“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37.](#)

Verify the deployment of a IBM WebSphere MQ basic part by running the "put" and "get" sample programs on the virtual machine where the IBM WebSphere MQ basic part is deployed.

The user ID `virtuser` is not a member of the `mqm` group. The example demonstrates the use of the **sudo** command to make `virtuser` a temporary member of the `mqm` group for the duration of the command. The example also sets up the minimum set of permissions for `virtuser` to run the "put" and "get" sample programs, and then removes them. As an alternative, the example also shows running the **sudo** command to run the "put" and "get" sample programs.

1. Log on to a VNC terminal session.

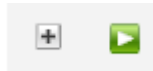
- a) In the appliance user interface, click **Instances > Virtual Systems**. Click the virtual system instance in the navigator.

The property window for the virtual system instance opens.

- b) In the property window, click the plus sign next to **Virtual machines** to expand the list of virtual machines.

A list of the virtual machines in the system instance opens. Each virtual machine has a plus sign and

an icon to show that it is running:



- c) Click the plus sign next to the virtual machine you want to connect to.

The properties of the virtual machine are listed.

- d) Scroll down to the **Consoles** section of the page and click **VNC**.

The VNC console opens in a browser window. It prompts you for a password.

- e) Type the password for the `virtuser` user ID and click **OK**.

The password is one of the parameters of the IBM WebSphere MQ basic part. It is defined in the pattern, or when the pattern is deployed.

The VNC window opens showing the desktop.

2. Open a terminal in the VNC window.

- a) Click **Applications > System Tools > Terminal**.

The terminal session opens and lists some key properties of the installation of IBM WebSphere MQ, the name of the queue manager, and the terminal session properties and environment.

3. Authorize the user ID `virtuser` to run the verification application.

- a) Give `virtuser` permission to connect to the queue manager.

```
sudo -u mqm setmqaut -m QmgrName -t qmgr -p virtuser +connect
```

The system responds with:

```
[sudo] password for virtuser:
```

Type the password for the user ID `virtuser`.

The system responds with:

The setmqaut command completed successfully.

- b) Give virtuser permission to use put, get, and inquire MQI calls against the queue used for verification.

```
sudo -u mqm setmqaut -m QmgrName -n SYSTEM.DEFAULT.LOCAL.QUEUE -t queue -p virtuser +put  
+get +inq
```

The system responds with:

The setmqaut command completed successfully.

4. Put a message on SYSTEM.DEFAULT.LOCAL.QUEUE.

```
/opt/mqm/samp/bin/amqspout SYSTEM.DEFAULT.LOCAL.QUEUE QmgrName
```

The system response:

Sample AMQSPUT0 start
target queue is qName

5. Type in a message followed by two new lines

Hello world

The system response:

Sample AMQSPUT0 end

6. Get the message from SYSTEM.DEFAULT.LOCAL.QUEUE.

```
/opt/mqm/samp/bin/amqsget SYSTEM.DEFAULT.LOCAL.QUEUE QmgrName
```

The system response:

Sample AMQSGET0 start
message <Hello world>
15 seconds delay
no more messages
Sample AMQSGET0 end

You demonstrated the queue manager is working correctly.

You have completed the verification task. You might choose to remove the changes you made, and try the **sudo** command to run the "put" and "get" sample programs.

1. Remove the authorizations you set by running the following commands:

```
sudo -u mqm setmqaut -m QmgrName -t qmgr -p virtuser -connect  
sudo -u mqm dspmqaut -m QmgrName -t qmgr -p virtuser  
sudo -u mqm setmqaut -m QmgrName -n SYSTEM.DEFAULT.LOCAL.QUEUE -t queue -p virtuser -put  
-get -inq  
sudo -u mqm dspmqaut -m QmgrName -n SYSTEM.DEFAULT.LOCAL.QUEUE -t queue -p virtuser
```

2. Run the verification again with the **sudo** command:

- a. Put a message onto SYSTEM.DEFAULT.LOCAL.QUEUE

```
sudo -u mqm /opt/mqm/samp/bin/amqspout SYSTEM.DEFAULT.LOCAL.QUEUE QmgrName
```

Type a message followed by two new lines.

- b. Get the message from SYSTEM.DEFAULT.LOCAL.QUEUE

```
sudo -u mqm /opt/mqm/samp/bin/amqsget SYSTEM.DEFAULT.LOCAL.QUEUE QmgrName
```

Wait for 15 seconds for the program to finish.

Verifying deployed IBM WebSphere MQ parts from a remote IBM WebSphere MQ MQI client.

Verify the deployment of IBM WebSphere MQ parts in a virtual system pattern by running the "put" and "get" sample MQI client programs. Run the programs from inside or outside of the cloud where the parts are deployed.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- Define at least one authorized user, and one authorized IP address in the IBM WebSphere MQ parts that you add to the IBM WebSphere MQ virtual system pattern; see [Table 2 on page 51](#) in "IBM WebSphere MQ basic part" on page 50.
- Add the `verify.mqsc` script to the IBM WebSphere MQ basic parts in the IBM WebSphere MQ virtual system pattern before you deploy the pattern; see ["Example: Deploy a local queue with access permissions for an authorized user" on page 27](#).

Note: If you have not added `verify.mqsc` to a IBM WebSphere MQ part, you must log on to a terminal session for the virtual machine you want to verify in the virtual system pattern. Log on as the `virtuser` user ID; see ["Verifying a deployed IBM WebSphere MQ basic part from a VNC session" on page 41](#) or ["Verifying a deployed IBM WebSphere MQ basic part from a VNC session" on page 41](#). Run the following command:

```
sudo -u mqm runmqsc qMgrName < verify.mqsc
```

- Deploy the pattern successfully; see ["Deploying a pattern that contains an IBM WebSphere MQ basic part" on page 37](#).
- Log on to a Windows, UNIX, or Linux workstation. The workstation must have a TCP/IP connection to the virtual machine where the IBM WebSphere MQ part you want to verify is running. The address of the workstation must be in the range authorized in the IBM WebSphere MQ basic part.

Verify the deployment of a IBM WebSphere MQ part by running the "put" and "get" MQI client sample programs. You can run the programs from inside or outside of the cloud where the part you want to verify is running.

1. Find an IBM WebSphere MQ installation on the workstation where you are going to run the "put" and "get" sample MQI client programs.
 - If the "put" and "get" sample MQI client programs are not installed, you must install them. Install either the IBM WebSphere MQ server or the IBM WebSphere MQ client on the workstation.
 - You might have multiple installations of the IBM WebSphere MQ client or server on the workstation. If so, select an installation to run the samples from.
 - a. On Windows:

```
MQ_INSTALLATION_PATH\bin\setmqenv -s
```

- b. On UNIX and Linux:

```
. MQ_INSTALLATION_PATH/bin/setmqenv -s
```

2. Locate the "put" and "get" sample programs.
 - On UNIX and Linux, change into the `MQ_INSTALLATION_PATH/samp/bin` directory.
 - On Windows the path to the sample programs is already set in the system path environment variable.
3. Find the IP address of the virtual machine you want to verify.

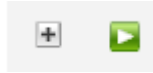
The same instructions apply to IBM Workload Deployer and IBM PureApplication System.

- a) In the appliance user interface, click **Instances > Virtual Systems**. Click the virtual system instance in the navigator.

The property window for the virtual system instance opens.

- b) In the property window, click the plus sign next to **Virtual machines** to expand the list of virtual machines.

A list of the virtual machines in the system instance opens. Each virtual machine has a plus sign and



an icon to show that it is running:

- c) Click the plus sign next to the virtual machine you want to connect to.

The properties of the virtual machine are listed.

- d) Find the IP address of the virtual machine.

The hostname and IP address is listed next to each **Network interface** property in the list of **Hardware and network** properties.

4. Find the port number of the IBM WebSphere MQ listener for the queue manager created by the IBM WebSphere MQ part.

- a) In the same property page, click `remote_std_out.log`

It is in the property page section headed **Script packages**.

The output from the script contains a line that identifies the TCP/IP port the queue manager listener is listening on, `MQ_TCP_LISTENER_PORT=2414`.

5. Set the MQSERVER environment variable.

Set the MQSERVER environment variable in the command window you are going to run the sample programs from.

- On Windows:

```
SET MQSERVER=SYSTEM.DEF.SVRCONN/TCP/n.n.n.n(port)
```

- On UNIX and Linux:

```
export MQSERVER=SYSTEM.DEF.SVRCONN/TCP/'n.n.n.n:port'
```

6. Start the sample "put" program by entering one of the following commands.

qMgrName is the name of the queue manager created by a IBM WebSphere MQ part. The name is set either when the part is incorporated in to a virtual system pattern, or when the pattern is deployed.

qName is the name of the local queue that was authorized in the `verify.mqsc` script.

- On Windows:

```
amqsputc qName qMgrName
```

- On UNIX and Linux:

```
./amqsputc qName qMgrName
```

The system response:

```
Sample AMQSPUT0 start  
target queue is qName
```

7. Type in a message followed by two new lines

```
Hello world
```

The system response:

```
Sample AMQSPUT0 end
```

8. Start the same "get" program by entering one of the following commands.

- On Windows:

```
amqspgetc qName qMgrName
```

- On UNIX and Linux:

```
./amqsgetc qName qMgrName
```

The system response:

```
Sample AMQSGETO start
message <Hello world>
15 seconds delay
no more messages
Sample AMQSGETO end
```

Run IBM WebSphere MQ Explorer on your workstation from the SupportPac MS0T; see [“Running IBM WebSphere MQ Explorer”](#) on page 46.

Related tasks

[Preparing and running the sample programs](#)

[Connecting a WebSphere MQ MQI client to a queue manager, using the MQSERVER environment variable](#)

[Example: Deploy a local queue with access permissions for an authorized user](#)

Administration

The tasks in this section help you administer the deployment of IBM WebSphere MQ Hypervisor Edition on an appliance or from VMware ESX hypervisor.

The tasks in the section assist you with the installation and deployment of virtual machines and virtual system patterns that include IBM WebSphere MQ. To administer IBM WebSphere MQ, see [Administering WebSphere MQ](#).

Running IBM WebSphere MQ Explorer

Start IBM WebSphere MQ Explorer locally from a VNC terminal session, or run it remotely as a IBM WebSphere MQ MQI client.

If you choose to run IBM WebSphere MQ Explorer locally from a VNC session, log on to the `virtuser` user ID and use the **sudo** command to delegate `mqm` authority to `virtuser` to run IBM WebSphere MQ Explorer. IBM WebSphere MQ Explorer runs in bindings mode, and you have full administrative authority to create, start, and manage queue managers and manage clusters.

If you choose to run IBM WebSphere MQ Explorer as a IBM WebSphere MQ MQI client, you have authority to manage queue managers and clusters. You cannot create queue managers, or start them remotely. The advantage of administering the virtual system instance remotely is that it is convenient for administering all the queue managers in the virtual system instance.

Choose one of the tasks.

- [“Running IBM WebSphere MQ Explorer locally from a VNC terminal session”](#) on page 46
- [“Running IBM WebSphere MQ Explorer remotely as a IBM WebSphere MQ MQI client”](#) on page 48

Running IBM WebSphere MQ Explorer locally from a VNC terminal session

Running IBM WebSphere MQ Explorer locally with the permissions of the `mqm` group give you access to all the capabilities of explorer. You must configure the permissions of the window that IBM WebSphere MQ Explorer opens in VNC.

1. The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.

2. Set the **VNC** property in the IBM WebSphere MQ basic part to its default value, `true`, before you deploy the pattern.
3. Remember the password to the `virtuser` user ID, which is set in the IBM WebSphere MQ basic part properties.

Restriction: VNC requires that a password must have at least six characters, and it treats only the first eight characters of a password as significant. If the password has fewer than six characters, VNC appends the ASCII "0" character to the password to make it six characters long; for example: "pass" becomes "pass00".

4. Deploy the pattern successfully; see [“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#).

Although you can start IBM WebSphere MQ Explorer from the VNC desktop menus, start it from a terminal window. Unless you made `virtuser` a member of the `mqm` group, when started from the desktop menus IBM WebSphere MQ Explorer is not authorized to view or change queue manager properties. When you start IBM WebSphere MQ Explorer from a terminal, you can delegate it `mqm` membership with the **sudo** command.

1. Log on to a VNC terminal session.

- a) In the appliance user interface, click **Instances > Virtual Systems**. Click the virtual system instance in the navigator.

The property window for the virtual system instance opens.

- b) In the property window, click the plus sign next to **Virtual machines** to expand the list of virtual machines.

A list of the virtual machines in the system instance opens. Each virtual machine has a plus sign and



an icon to show that it is running:

- c) Click the plus sign next to the virtual machine you want to connect to.

The properties of the virtual machine are listed.

- d) Scroll down to the **Consoles** section of the page and click **VNC**.

The VNC console opens in a browser window. It prompts you for a password.

- e) Type the password for the `virtuser` user ID and click **OK**.

The password is one of the parameters of the IBM WebSphere MQ basic part. It is defined in the pattern, or when the pattern is deployed.

The VNC window opens showing the desktop.

2. Open a terminal in the VNC window.

- a) Click **Applications > System Tools > Terminal**.

The terminal session opens and lists some key properties of the installation of IBM WebSphere MQ, the name of the queue manager, and the terminal session properties and environment.

3. Authorize the display that IBM WebSphere MQ Explorer connects to.

Type the following command to add the local IP address to the access control list.

```
xhost +127.0.0.1
```

The system responds:

```
127.0.0.1 being added to access control list
```

4. Start IBM WebSphere MQ Explorer with delegated authority from the `mqm` group.

- a) Type the following command to start IBM WebSphere MQ Explorer.

```
sudo -u mqm strmqcfig
```


The system responds by prompting for a password: `[sudo:] password for virtuser.`

b) Type the password for the `virtuser` user ID.

The system responds by starting IBM WebSphere MQ Explorer in a new window and writing a response from the **`strmqcfcg`** command in the current terminal.

Open the queue manager created by the IBM WebSphere MQ basic part in IBM WebSphere MQ Explorer.

Related tasks

[“Verifying a deployed IBM WebSphere MQ basic part from a VNC session” on page 41](#)

Verify a IBM WebSphere MQ basic part locally, without authorizing remote users to the `virtuser` user ID.
Test the deployment from a VNC session.

Running IBM WebSphere MQ Explorer remotely as a IBM WebSphere MQ MQI client

Run IBM WebSphere MQ Explorer remotely, if you want to manage multiple queue managers in a IBM WebSphere MQ virtual system instance.

In this example, IBM WebSphere MQ Explorer runs with the identity of an authorized user ID with IBM WebSphere MQ administrative authority. The authorized user ID is mapped to the server-connection channel that connects IBM WebSphere MQ Explorer to the queue manager in the IBM WebSphere MQ basic part.

- The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
- You must have defined at least one authorized user ID, and at least one IP address for remote administration when you deployed the IBM WebSphere MQ virtual system pattern; see [“Security” on page 25](#).
- Deploy the pattern successfully; see [“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#).
- Log on to a Windows, UNIX, or Linux workstation. The workstation must have a TCP/IP connection to the virtual machine where the IBM WebSphere MQ basic part you want to verify is running. The address of the workstation must be in the range you authorized in the IBM WebSphere MQ basic part.

Start IBM WebSphere MQ Explorer by connecting it remotely from one of the IP addresses authorized in the IBM WebSphere MQ basic part. When started from an authorized IP address, IBM WebSphere MQ Explorer connects with the server-connection channel `SYSTEM.ADMIN.SVRCONN`, which takes the identity of one of the authorized user IDs.

1. Install IBM WebSphere MQ Explorer on your workstation from the SupportPac MS0T; see [MS0T: IBM WebSphere MQ Explorer](#).

See the instructions in the SupportPac to install IBM WebSphere MQ Explorer.

2. Find the IP address of the virtual machine you want to verify.

The same instructions apply to IBM Workload Deployer and IBM PureApplication System.

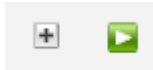
- a) In the appliance user interface, click **Instances > Virtual Systems**. Click the virtual system instance in the navigator.

The property window for the virtual system instance opens.

- b) In the property window, click the plus sign next to **Virtual machines** to expand the list of virtual machines.

A list of the virtual machines in the system instance opens. Each virtual machine has a plus sign and

an icon to show that it is running:



- c) Click the plus sign next to the virtual machine you want to connect to.

The properties of the virtual machine are listed.

- d) Find the IP address of the virtual machine.

The hostname and IP address is listed next to each **Network interface** property in the list of **Hardware and network** properties.

3. Find the port number of the IBM WebSphere MQ listener for the queue manager created by the IBM WebSphere MQ part.

- a) In the same property page, click `remote_std_out.log`

It is in the property page section headed **Script packages**.

The output from the script contains a line that identifies the TCP/IP port the queue manager listener is listening on, `MQ_TCP_LISTENER_PORT=2414`.

4. Start IBM WebSphere MQ Explorer on your workstation.

See the instructions in the SupportPac how to start IBM WebSphere MQ Explorer.

5. Right-click the **Queue Managers folder** > **Add Remote Queue Manager**.

6. In the **Queue manager name** field, type the name of the queue manager created by the IBM WebSphere MQ basic part. Click **Next**.

7. Type the **Host name or IP address** and **Port number**. Find the IP address and port number in steps 2 and 3. Click **Finish**.

IBM WebSphere MQ Explorer responds by adding the queue manager to the queue manager folder and connecting to it.

Open the queue manager created by the IBM WebSphere MQ basic part in IBM WebSphere MQ Explorer.

Collecting IBM WebSphere MQ error logs from a virtual system instance

Upload the IBM WebSphere MQ error logs from a IBM WebSphere MQ basic part in a virtual system instance. Run the **Must gather** command to collect the logs from the appliance.

1. The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor.
2. The appliance administrator must configure the "plug-ins for virtual systems" before you deploy the virtual system pattern from the appliance.

The steps to configure the plug-ins for virtual systems are as follows:

- a. From the appliance user interface, click **Cloud** > **System Plugins** > **Foundation Pattern Type 2.0** > **virtualsystem** > **Configure**
- b. Check **Enable plugins on virtual systems**.
3. The virtual system instance contains IBM WebSphere MQ basic parts and is running.

Obtain the IBM WebSphere MQ error logs from the appliance. Run the **Execute now** command of the **Must Gather Logs** script on the appliance to copy the logs to the appliance, and then transfer the logs to your workstation for analysis. The script gathers the IBM WebSphere MQ error logs from the IBM WebSphere MQ basic part in a virtual machine in a virtual system instance.

1. Open the properties of a virtual system instance that contain a IBM WebSphere MQ basic part.

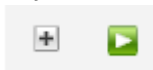
- a) In the appliance user interface, click **Instances** > **Virtual Systems**. Click the virtual system instance in the navigator.

The property window for the virtual system instance opens.




- b) In the property window, click the plus sign next to **Virtual machines** to expand the list of virtual machines.

A list of the virtual machines in the system instance opens. Each virtual machine has a plus sign and

an icon to show that it is running:



- c) Click the plus sign next to the virtual machine you want to connect to.

- The properties of the virtual machine are listed.
2. Capture the current logs.
 - a) In the Script packages section, find the **Must Gather Logs** script package and click **Execute now**, .
 - The appliance responds with a prompt for an administrator user ID and password.
 - b) Click **OK**
 - No user ID or password is required.
 - The appliance creates a **Must Gather logs** list item in the **Script Packages** properties. The list entry has an hour glass icon  while the logs are created. When the logs are available the hour glass changes to a check mark icon .
 3. Transfer the logs to your workstation.
 - a) In the **Must Gather logs** list item in the **Script Packages**, click the compressed file; for example, `cloudburst_collect1340982954021.zip`.
 - b) Follow your browser instructions to save the file on your workstation.
 4. Expand the compressed file to retrieve the log files
- The compressed file contains a tape archive `mq.tar`. The archive contains the IBM WebSphere MQ logs.
- You must have a compressed file utility on your workstation, such as 7-Zip on Windows, which expands both `.zip` and `.tar` files. Expand both the Must Gather compressed folder, and the `mq.tar` that it contains.
- Expanding the `mq.tar` archive creates the directories `var/mqm/qmgrs` and `var/mqm/errors`, and the IBM WebSphere MQ log files contained in those directories, from the IBM WebSphere MQ basic part in the virtual machine.

Reference

IBM WebSphere MQ Hypervisor Edition reference information

For IBM WebSphere MQ reference information see [IBM WebSphere MQ reference](#).

Related concepts

[WebSphere MQ configuration reference](#)

[WebSphere MQ administration reference](#)

[WebSphere MQ developing applications reference](#)

[WebSphere MQ security reference](#)

[WebSphere MQ performance and monitoring reference](#)

[WebSphere MQ troubleshooting and support reference](#)

Related reference

[WebSphere MQ telemetry reference](#)

[WebSphere MQ messages](#)

IBM WebSphere MQ basic part

The IBM WebSphere MQ basic part contains a queue manager. The properties and installed components of the part are listed.

The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor

The part is included when you download IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux from Passport Advantage. It is also included in the default IBM WebSphere MQ virtual system pattern that is created by the installation

command script; see [“Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance”](#) on page 17. You can update the part IBM Workload Deployer by downloading IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux from Passport Advantage. Downloading from Passport Advantage is the only way to install the part on IBM PureApplication System.

Properties are listed in [“Properties of the IBM WebSphere MQ basic part”](#) on page 51. You can change the properties when you configure a pattern, and by running a script. The installed IBM WebSphere MQ components and messages are listed in [“Components and messages installed in the IBM WebSphere MQ basic part”](#) on page 53.

Properties of the IBM WebSphere MQ basic part

Table 2 on page 51 lists the common properties of the IBM WebSphere MQ basic part. Table 3 on page 52 and Table 4 on page 53 list the platform-specific properties of the IBM WebSphere MQ basic part.

The preinstalled part on IBM Workload Deployer might be IBM WebSphere MQ Version 7.0.1 or IBM WebSphere MQ Version 7.5, depending on the manufacturing date.

All the fields in the IBM WebSphere MQ basic part must be provided before you deploy a part. You can provide values for the fields either when you edit the part, or when you deploy the pattern. You can lock the fields in the part when you edit it to prevent modification when the part is deployed. Unlike a locked pattern, which you cannot unlock, you can unlock locked fields in a part. Lock or unlock them when you edit a part in a pattern.

Table 2. Common Properties of the IBM WebSphere MQ basic part		
Property	Default value	Description
Image memory size (MB)	2048	The memory size of the image in megabytes.
Password (root)		The password for the root user ID. This password is for the operating system for the virtual machine that is represented by this part in the pattern. In the Verify password field, reenter the password.
Password (virtuser)		The virtual user password for the virtuser user ID of the operating system. In the Verify password field, reenter the password.
Queue Manager		A unique name of the queue manager; see Specify a unique queue manager name .
Queue Manager Description		The description of the queue manager.
Queue Manager TCP/IP listener port	2414	The TCP/IP listener port number.
Authorized users	null ¹	Each user in the list is created as a matching user ID and primary group name. Separate multiple user names with spaces. The groups have IBM WebSphere MQ administrative authority; see “Security” on page 25.

Table 2. Common Properties of the IBM WebSphere MQ basic part (continued)

Property	Default value	Description
Authorized IP addresses	null ¹	A single IP address or group of addresses in dot-decimal notation, such as the IPv4 private address 10.0.0.1, or the IPv6 address range, 2001:*.*.1; see Generic IP addresses . The IP addresses are authorized to connect remotely to IBM WebSphere MQ Explorer. Any remote user assumes the identity of an authorized user when they connect over the SYSTEM.DEF.SVRCONN; see “Security” on page 25 .
Queue Manager Dead Letter Queue	SYSTEM.DEAD.LETTER.QUEUE	The queue manager dead letter queue.
Queue Manager uses linear logging	False	Determines whether the queue manager uses linear logging. If False is specified, circular logging is used.
Queue Manager log pages	1024	The number of queue manager log pages.
Primary Logs	20	The number of primary IBM WebSphere MQ log files.
Secondary Logs	12	The number of secondary IBM WebSphere MQ log files.
Log Path	/var/mqm/log ³	The file system directory for log data.
Data Path	/var/mqm/qmgrs ³	The file system directory for queue manager data.
Error Path	/var/mqm/errors ³	The file system directory for error data.

Note:

1. The values of **Authorized Users** and **Authorized IP Addresses** are coupled together. For either to take effect, you must provide both values. If you specify only one of the values, it is ignored.
2. The value **null** is a special string that is used to denote a valid empty value.
3. For **Log Path**, **Data Path**, and **Error Path**, the directory must be specified by using a fully qualified path that is prefixed with the root '/' directory. If the path is not prefixed with '/', then '/' is added to the beginning of the path. If the path does not exist, then it is created. The characters that are used in a path must be valid for the platform that you are using.

Table 3. Platform-specific properties of the IBM WebSphere MQ Hypervisor Edition for AIX part

Property	Default value	Description
Virtual CPUs	1	The number of virtual processors to expose within the image.
Physical Processor Count	0.3	The number of physical processors that are allocated to this virtual machine (LPAR)

Table 3. Platform-specific properties of the IBM WebSphere MQ Hypervisor Edition for AIX part (continued)

Property	Default value	Description
Reserve physical CPUs	false	Determines whether physical processors are reserved.

Table 4. Platform-specific properties of the IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux part

Property	Default value	Description
Number of CPUs	1	The number of processors to expose within the image.
Enable VNC	False	Determines whether the Virtual network computing (VNC) server is enabled for the virtual machine that is associated with the part in the pattern. Acceptable values are True or False. When enabled, the VNC client accesses the virtual machine with the user ID virtuser . By default, this property is locked. You can change the value only during pattern creation; it cannot be changed during deployment. You must supply a password. This password is preset to password .

Components and messages installed in the IBM WebSphere MQ basic part

Table 5 on page 53 lists the IBM WebSphere MQ components that are installed with the IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux. Table 6 on page 56 lists the message catalogs for IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux.

Table 5. IBM WebSphere MQ components for Linux systems.

A six-column table listing the available product components, what source they are available from, and the name of the component.

Component	Description	Server DVD	Client DVD	Hypervisor	Component name
Runtime	Contains files that are common to both server and client installations. Note: This component must be installed.	✓	✓	✓	MQSeriesRuntime
Server	You can use the server to run queue managers on your system and connect to other systems over a network. Provides messaging and queuing services to applications, and support for IBM WebSphere MQ client connections.	✓		✓	MQSeriesServer

Table 5. IBM WebSphere MQ components for Linux systems.

A six-column table listing the available product components, what source they are available from, and the name of the component.

(continued)

Component	Description	Server DVD	Client DVD	Hypervisor	Component name
Standard Client	The IBM WebSphere MQ MQI client is a small subset of IBM WebSphere MQ, without a queue manager, that uses the queue manager and queues on other (server) systems. It can be used only when the system it is on is connected to another system that is running a full server version of IBM WebSphere MQ. The client and the server can be on the same system if required.	✓	✓	✓	MQSeriesClient
SDK	The SDK is required for compiling applications. It includes sample source files, and the bindings (files .H, .LIB, .DLL, and others), that you need to develop applications to run on IBM WebSphere MQ.	✓	✓	✓	MQSeriesSDK
Sample programs	The sample application programs are needed if you want to check your IBM WebSphere MQ installation using the verification procedures.	✓	✓	✓	MQSeriesSamples
Java messaging	The files needed for messaging using Java (includes Java Messaging Service).	✓	✓	✓	MQSeriesJava
Man pages	UNIX man pages, in U.S. English, for: control commands MQI commands MQSC commands	✓	✓	✓	MQSeriesMan
Java JRE	A Java Runtime Environment, Version 6.0, that is used by those parts of IBM WebSphere MQ that are written in Java.	✓	✓	✓	MQSeriesJRE
Message Catalogs	For available languages, see the table of message catalogs that follows.	✓	✓	✓	
IBM Global Security Kit	IBM Global Security Kit V8 Certificate and SSL Base Runtime.	✓	✓	✓	MQSeriesGSKit

Table 5. IBM WebSphere MQ components for Linux systems.

A six-column table listing the available product components, what source they are available from, and the name of the component.

(continued)

Component	Description	Server DVD	Client DVD	Hypervisor	Component name
MQ Telemetry	<p>MQ Telemetry supports the connection of Internet Of Things (IOT) devices (that is, remote sensors, actuators and telemetry devices) that use the MQ Telemetry Transport (MQTT) protocol. The MQ Telemetry component comprises:</p> <ul style="list-style-type: none"> • The telemetry (MQXR) service. It enables a queue manager to act as an MQTT server, and communicate with MQTT client apps. • A set of MQTT client libraries. These libraries help you write the MQTT client apps that IOT devices use to communicate with MQTT servers. <p>MQ Telemetry is only available on Linux for System x (64 bit) and Linux for System z.</p> <p>The most up-to-date version of the MQTT client libraries is available in the free download Mobile Messaging and M2M Client Pack.</p> <p>See also Installing IBM WebSphere MQ Telemetry.</p>	✓	✓	✓	MQSeriesXRService MQSeriesXRClients
MQ Explorer	Use IBM WebSphere MQ MQ Explorer to administer and monitor resources on Linux x86 and x86-64 systems.	✓		✓	MQSeriesExplorer
Managed File Transfer	MQ Managed File Transfer transfers files between systems in a managed and auditable way, regardless of file size or the operating systems used. For information about the function of each component, see WebSphere MQ Managed File Transfer product options .	✓		✓	MQSeriesFTAgent MQSeriesFTBase MQSeriesFTLogger MQSeriesFTService MQSeriesFTTools

Table 5. IBM WebSphere MQ components for Linux systems.

A six-column table listing the available product components, what source they are available from, and the name of the component.

(continued)

Component	Description	Server DVD	Client DVD	Hypervisor	Component name
Advanced Message Security	Provides a high level of protection for sensitive data flowing through the IBM WebSphere MQ network, while not impacting the end applications. You must install this component on all IBM WebSphere MQ installations that host queues you want to protect. You must install the IBM Global Security Kit component on any IBM WebSphere MQ installation that is used by a program that puts or gets messages to or from a protected queue, unless you are using only Java client connections.	✓		✓	MQSeriesAMS

Table 6. IBM WebSphere MQ message catalogs for Linux systems.

A two-column table listing the available message catalogs.

Message catalog language	Component name
Brazilian Portuguese	MQSeriesMsg_pt
Czech	MQSeriesMsg_cs
French	MQSeriesMsg_fr
German	MQSeriesMsg_de
Hungarian	MQSeriesMsg_hu
Italian	MQSeriesMsg_it
Japanese	MQSeriesMsg_ja
Korean	MQSeriesMsg_ko
Polish	MQSeriesMsg_pl
Russian	MQSeriesMsg_ru
Spanish	MQSeriesMsg_es
Simplified Chinese	MQSeriesMsg_Zh_CN
Traditional Chinese	MQSeriesMsg_Zh_TW
U.S. English	not applicable

Table 7 on page 57 lists the IBM WebSphere MQ components that are installed with the IBM WebSphere MQ Hypervisor Edition for AIX. [Table 8 on page 59](#) lists the message catalogs for IBM WebSphere MQ Hypervisor Edition for AIX.

Table 7. IBM WebSphere MQ components for AIX systems.

A six-column table listing the available product components, what source they are available from, and the name of the component.

Component	Description	Server DVD	Client DVD	Hypervisor	Component name
Runtime	Contains files that are common to both server and client installations. Note: This component must be installed.	✓	✓	✓	mqm.base.runtime
Server	You can use the server to run queue managers on your system and connect to other systems over a network. Provides messaging and queuing services to applications, and support for IBM WebSphere MQ client connections.	✓		✓	mqm.server
Standard Client	The IBM WebSphere MQ MQI client is a small subset of IBM WebSphere MQ, without a queue manager, that uses the queue manager and queues on other (server) systems. It can be used only when the system it is on is connected to another system that is running a full server version of IBM WebSphere MQ. The client and the server can be on the same system if required.	✓	✓	✓	mqm.client.rte
SDK	The SDK is required for compiling applications. It includes sample source files, and the bindings (files .H, .LIB, .DLL, and others), that you need to develop applications to run on IBM WebSphere MQ.	✓	✓	✓	mqm.base.sdk
Sample programs	The sample application programs are needed if you want to check your IBM WebSphere MQ installation using the verification procedures.	✓	✓	✓	mqm.base.samples
Java messaging	The files needed for messaging using Java (includes Java Messaging Service).	✓	✓	✓	mqm.java.rte
Man pages	UNIX man pages, in U.S. English, for: control commands MQI commands MQSC commands	✓	✓	✓	mqm.man.en_US.data
Java JRE	A Java Runtime Environment, Version 6.0, that is used by those parts of IBM WebSphere MQ that are written in Java.	✓	✓	✓	mqm.jre.rte
Message Catalogs	For available languages, see the table of message catalogs that follows.	✓	✓	✓	
IBM Global Security Kit	IBM Global Security Kit V8 Certificate and SSL Base Runtime.	✓	✓	✓	mqm.gskit.rte

Table 7. IBM WebSphere MQ components for AIX systems.

A six-column table listing the available product components, what source they are available from, and the name of the component.

(continued)

Component	Description	Server DVD	Client DVD	Hypervisor	Component name
MQ Telemetry	<p>MQ Telemetry supports the connection of Internet Of Things (IOT) devices (that is, remote sensors, actuators and telemetry devices) that use the MQ Telemetry Transport (MQTT) protocol. The MQ Telemetry component comprises:</p> <ul style="list-style-type: none"> • The telemetry (MQXR) service enables a queue manager to act as an MQTT server, and communicate with MQTT client apps. • A set of MQTT client libraries. These libraries help you write the MQTT client apps that IOT devices use to communicate with MQTT servers. <p>The most up-to-date version of the MQTT client libraries is available in the free download Mobile Messaging and M2M Client Pack.</p> <p>See also Installing IBM WebSphere MQ Telemetry.</p>	✓	✓	✓	mqm.xr.service mqm.xr.clients
Managed File Transfer	<p>MQ Managed File Transfer transfers files between systems in a managed and auditable way, regardless of file size or the operating systems used. For information about the function of each component, see WebSphere MQ Managed File Transfer product options.</p>	✓		✓	mqm.ft.agent mqm.ft.base mqm.ft.logger mqm.ft.service mqm.ft.tools
Advanced Message Security	<p>Provides a high level of protection for sensitive data flowing through the IBM WebSphere MQ network, while not impacting the end applications. You must install this component on all IBM WebSphere MQ installations that host queues you want to protect.</p> <p>You must install the IBM Global Security Kit component on any IBM WebSphere MQ installation that is used by a program that puts or gets messages to or from a protected queue, unless you are using only Java client connections.</p>	✓		✓	mqm.ams.rte

Table 8. IBM WebSphere MQ message catalogs for AIX systems.

A two-column table listing the available message catalogs.

Message catalog language	Component name
Brazilian Portuguese	mqm.msg.pt_BR
Czech	mqm.msg.cs_CZ
French	mqm.msg.fr_FR
German	mqm.msg.de_DE
Hungarian	mqm.msg.hu_HU
Italian	mqm.msg.it_IT
Japanese	mqm.msg.ja_JP, mqm.msg.Ja_JP
Korean	mqm.msg.ko_KR
Polish	mqm.msg.pl_PL
Russian	mqm.msg.ru_RU
Spanish	mqm.msg.es_ES
Simplified Chinese	mqm.msg.zh_CN, mqm.msg.Zh_CN
Traditional Chinese	mqm.msg.zh_TW, mqm.msg.Zh_TW
U.S. English	mqm.msg.en_US

Related concepts

[“Parts and patterns” on page 8](#)

IBM WebSphere MQ Hypervisor editions contain a IBM WebSphere MQ basic part which you include in virtual system patterns. Virtual system patterns are made up of parts and parts have properties. Each part represents a single virtual machine. A *pattern* provides a topology definition for repeatable deployment that can be shared. Patterns describe the function provided by each virtual machine in a virtual system. Each function is identified as a part in the pattern

[“Security” on page 25](#)

Related tasks

[“Adding the IBM WebSphere MQ basic part to a pattern” on page 30](#)

Add the IBM WebSphere MQ basic part to an IBM WebSphere MQ pattern, and edit its properties to start configuring the pattern.

[“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#)

Deploy a pattern that contains one or more IBM WebSphere MQ basic parts to a cloud with IBM Workload Deployer or IBM PureApplication System.

IBM WebSphere MQ virtual system pattern

You can create a IBM WebSphere MQ virtual system pattern that includes queue managers by adding the IBM WebSphere MQ basic part to patterns that you create or copy. You can run a command script to install a default IBM WebSphere MQ virtual system pattern.

The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor

Related concepts

[“Parts and patterns” on page 8](#)

IBM WebSphere MQ Hypervisor editions contain a IBM WebSphere MQ basic part which you include in virtual system patterns. Virtual system patterns are made up of parts and parts have properties. Each part

represents a single virtual machine. A *pattern* provides a topology definition for repeatable deployment that can be shared. Patterns describe the function provided by each virtual machine in a virtual system. Each function is identified as a part in the pattern

Related tasks

[“Copying a pattern” on page 29](#)

Make a new pattern by copying and modifying an existing pattern. Copy a read-only pattern, and modify the copy. Copies are called clones. When you clone a pattern you can associate it with a different version of the IBM WebSphere MQ virtual image.

[“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#)

Deploy a pattern that contains one or more IBM WebSphere MQ basic parts to a cloud with IBM Workload Deployer or IBM PureApplication System.

[“Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance” on page 17](#)

Install the IBM WebSphere MQ virtual image and script package to an appliance from a Windows or Linux workstation.

Related reference

[“IBM WebSphere MQ Hypervisor Edition command scripts” on page 61](#)

IBM WebSphere MQ Hypervisor Edition script packages

IBM WebSphere MQ Hypervisor Edition script packages contain command scripts and queue manager commands to configure the IBM WebSphere MQ basic part.

The topic applies only if you are using IBM Workload Deployer or IBM PureApplication System. It does not apply to direct virtual machine deployments onto VMware ESX hypervisor

With the tools in the appliances, you can create additional script packages.

You deploy an IBM WebSphere MQ script package by dragging it onto a IBM WebSphere MQ basic part.

Table 9. IBM WebSphere MQ script packages

File name	Script package
WMQAddToCluster.zip	WMQ: Add to Queue Manager Cluster
WMQRemoveFromCluster.zip	WMQ: Remove from Queue Manager Cluster
WMQExecuteMQSC.zip	WMQ: Run MQSC Scripts

Table 10. WMQ: Add to Queue Manager Cluster

Property	Value
MQ_ADD_TO_CLUSTER_NAME	Cluster name; see Cluster (CLUSTER) .
MQ_ADD_TO_FULL_REPOSITORY_QMGR_NAME	Name of one of the full repository queue managers for the cluster; see Cluster repository .
MQ_ADD_TO_FULL_REPOSITORY_CONNAME	Connection name of the full repository queue manager; see Connection name (CONNAME) .
MQ_ADD_TO_FULL_REPOSITORY_PORT	TCP/IP port number of the full repository queue manager.

Table 11. WMQ: Remove from Queue Manager Cluster

Property	Value
MQ_REMOVE_FROM_CLUSTER_NAME	Cluster name; see Cluster (CLUSTER) .

Table 11. WMQ: Remove from Queue Manager Cluster (continued)	
Property	Value
MQ_REMOVE_FROM_FULL_REPOSITORY_QMGR_NAME	Name of one of the full repository queue managers for the cluster; see Cluster repository .

Table 12. WMQ: Run MQSC Scripts	
Property	Value
Executes	Select whether to run the script at virtual system creation, deletion, or on demand by clicking the Execute now button

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

[“Adding an IBM WebSphere MQ script package to an appliance” on page 16](#)

Add an IBM WebSphere MQ script package for IBM WebSphere MQ Hypervisor Edition to the script catalog in IBM Workload Deployer or IBM PureApplication System.

[“Adding MQSC commands to a pattern” on page 31](#)

You can add one or more files of IBM WebSphere MQ commands to customize a pattern that contains IBM WebSphere MQ basic parts. The commands run when the pattern is deployed as a virtual system, when the virtual system is deleted, or when you choose to run the commands.

[“Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance” on page 17](#)

Install the IBM WebSphere MQ virtual image and script package to an appliance from a Windows or Linux workstation.

[“Adding a queue manager to a cluster” on page 33](#)

The WMQ: Add to Queue Manager Cluster cluster script adds a queue manager to a cluster.

[“Removing a queue manager from a cluster” on page 35](#)

The WMQ: Remove from Queue Manager Cluster script cleanly removes a queue manager from a cluster. It removes the queue manager from the cluster and removes its entries from the full repositories in the cluster.

IBM WebSphere MQ Hypervisor Edition command scripts

The IBM WebSphere MQ Hypervisor Edition command scripts install the IBM WebSphere MQ Hypervisor Edition virtual images, and script packages, and create a default IBM WebSphere MQ virtual system pattern.

The IBM WebSphere MQ Hypervisor Edition command scripts are listed in [Table 13 on page 62](#).

Run the command scripts on a Linux or Windows workstation with the command-line interpreter for the IBM Workload Deployer or IBM PureApplication System.

To download the command-line interpreter, click **Download Tooling > Download command-line interpreter** on the welcome page of the appliance user interface.

Syntax

```
expandDir\appliance.cli\bin\appliance -h ipAddr -u userID -p password -f scriptName.py
```

The command script parameters are as follows:

expandDir

expandDir is the directory that contains the command-line interpreter for the appliance.

appliance

appliance is the name of the command. It is either `deployer` if the command-line interpreter is downloaded from IBM Workload Deployer or `pure` if the command-line interpreter is downloaded from IBM PureApplication System. You must use the command-line interpreter that corresponds to the platform of the appliance.

ipAddr

ipAddr is the IP address of the appliance.

userID

userID is a user that has "cloud administration" or "create new catalog content" authority. On IBM Workload Deployer, you can replace the user ID by the built-in operator ID, `cbadmin`.

The virtual image, script package, and default IBM WebSphere MQ virtual system pattern are owned by this user. Other users are given read-only access to it.

password

The password of the user or operator.

scriptName

scriptName is one of the following installation scripts:

MQHVE-RHEL-AddVirtualImage

Install the virtual image.

MQHVE-RHEL-CreatePatternsandScripts

Install the script packages and default IBM WebSphere MQ virtual system pattern.

MQHVE-AIX-AddVirtualImage

Install the virtual image.

MQHVE-AIX-CreatePatternsandScripts

Install the script packages and default IBM WebSphere MQ virtual system pattern.

Command scripts

Table 13. IBM WebSphere MQ Hypervisor Edition command-line scripts	
Command script	What it does

Table 13. IBM WebSphere MQ Hypervisor Edition command-line scripts (continued)

Command script	What it does
	<p>The command script requires that a IBM WebSphere MQ basic part is on the appliance. The version of the command script must match the version of a IBM WebSphere MQ virtual image on the appliance.</p> <p>It creates script packages on the appliance from the following compressed files:</p> <p style="padding-left: 40px;">WMQExecuteMQSC.zip WMQAddtoQueueManagerCluster.zip WMQRemovefromQueueManagerCluster.zip</p> <p>It gives the script packages the following names:</p> <p style="padding-left: 40px;">WMQ: Run MQSC Scripts WMQ: Add to Queue Manager Cluster WMQ: Remove from Queue Manager Cluster</p> <p>It changes the default value of the Executes property of the WMQ: Remove from Queue Manager Cluster script to "at virtual system deletion".</p> <p>You can store one set of script packages that have the standard IBM WebSphere MQ assigned names on the appliance. The script packages are not versioned. Delete the script packages to replace them.</p> <p>The command script creates a IBM WebSphere MQ virtual system pattern on the appliance. The pattern is based on the version of the IBM WebSphere MQ basic part that corresponds to version of the command script.</p> <p>It gives the pattern its name based on the version of the IBM WebSphere MQ basic part that it used to create the pattern. It names the pattern WebSphere MQ V.R.M.F Basic.</p>

Examples

```
C:\cli\pure.cli\bin\pure -h 10.0.0.1 -u admin -p password -f MQHVE-RHEL-AddVirtualImage.py
```

```
C:\cli\pure.cli\bin\pure -h 10.0.0.1 -u admin -p password -f MQHVE-AIX-AddVirtualImage.py
```

Figure 13. Installing the virtual image on IBM PureApplication System.

```
C:\cli\deployer.cli\bin\deployer -h 10.0.0.2 -u cbadmin -p cbadmin -f MQHVE-RHEL-CreatePatternsandScripts.py
```

```
C:\cli\deployer.cli\bin\deployer -h 10.0.0.2 -u cbadmin -p cbadmin -f MQHVE-AIX-CreatePatternsandScripts.py
```

Figure 14. Installing script packages on IBM Workload Deployer.

Related tasks

[“Running a command script to install the IBM WebSphere MQ virtual image and script package to an appliance” on page 17](#)

Install the IBM WebSphere MQ virtual image and script package to an appliance from a Windows or Linux workstation.

The IBM WebSphere MQ Hypervisor Edition for AIX virtual machine image for appliances

The IBM WebSphere MQ Hypervisor Edition for AIX virtual image provides the operating system and product binary files that are required to create virtual system instances to deploy from IBM Workload Deployer or run on IBM PureApplication System.

Available virtual images

The IBM WebSphere MQ Hypervisor Edition for AIX virtual image is pre-installed in the IBM Workload Deployer catalog. You can also download the package for the selected IBM WebSphere MQ Hypervisor Edition for AIX from the [Passport Advantage and Passport Advantage Express web site](#) and then load the virtual machine image onto IBM Workload Deployer or IBM PureApplication System. The following virtual machine image is available:

- IBM WebSphere MQ Hypervisor Edition V7.5 for AIX.

After the virtual machine image has been added to the image catalog, you can create, manage, and deploy virtual system patterns to the cloud.

On IBM Workload Deployer, the image requires a cloud provisioned with POWER® servers and the PowerVM hypervisor.

Virtual image contents

The virtual image contains the following preinstalled software:

- AIX version 6.1
- IBM WebSphere MQ Version 7.5

Other software provided in the installation package

The installation package includes script packages and command scripts.

Maintenance

Maintenance for virtual machines is applied using the Apply Maintenance mechanism provided with the appliance.

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

[“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14](#)

Add the IBM WebSphere MQ virtual image for IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to the image catalog in IBM Workload Deployer or IBM PureApplication System.

[“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#)

Deploy a pattern that contains one or more IBM WebSphere MQ basic parts to a cloud with IBM Workload Deployer or IBM PureApplication System.

[“Applying service to a virtual system instance that contains IBM WebSphere MQ basic parts” on page 24](#)

Apply service to the IBM WebSphere MQ basic parts in a virtual system instance. You must first load the service that is required onto the appliance. You can control when the appliance applies the service to a virtual system instance. The appliance keeps a snapshot of the instance before service is applied, in case you need to restore it, and it maintains a service history.

Related reference

[“IBM WebSphere MQ Hypervisor Edition script packages” on page 60](#)

[“IBM WebSphere MQ Hypervisor Edition command scripts” on page 61](#)

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image for appliances

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image provides the operating system and product binary files that are required to create virtual system instances to deploy from IBM Workload Deployer or run on IBM PureApplication System.

Available virtual images

You can download the package for IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux from the [Passport Advantage and Passport Advantage Express web site](#) and then load the virtual machine image onto IBM Workload Deployer or IBM PureApplication System. The following virtual machine images are available:

- IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux Version 7.0.1 64 bit Red Hat Enterprise Linux Server.
- IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux Version 7.5 64 bit Red Hat Enterprise Linux Server.

After the virtual machine image has been added to the image catalog, you can create, manage, and deploy virtual system patterns to the cloud.

On IBM Workload Deployer, the image requires a cloud provisioned with servers capable of running 64 bit Red Hat Enterprise Linux Server and the VMware ESX hypervisor.

Virtual image contents

The virtual image contains the following preinstalled software:

- 64 bit Red Hat Enterprise Linux Server
- IBM WebSphere MQ Version 7.5
- IBM WebSphere MQ Explorer

Other software provided in the installation package

The installation package includes script packages and command scripts.

Maintenance

Maintenance for virtual machines is applied using the Apply Maintenance mechanism provided with the appliance.

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

[“Adding an IBM WebSphere MQ virtual image to an appliance” on page 14](#)

Add the IBM WebSphere MQ virtual image for IBM WebSphere MQ Hypervisor Edition for AIX or IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux to the image catalog in IBM Workload Deployer or IBM PureApplication System.

[“Deploying a pattern that contains an IBM WebSphere MQ basic part” on page 37](#)

Deploy a pattern that contains one or more IBM WebSphere MQ basic parts to a cloud with IBM Workload Deployer or IBM PureApplication System.

[“Applying service to a virtual system instance that contains IBM WebSphere MQ basic parts” on page 24](#)

Apply service to the IBM WebSphere MQ basic parts in a virtual system instance. You must first load the service that is required onto the appliance. You can control when the appliance applies the service to a virtual system instance. The appliance keeps a snapshot of the instance before service is applied, in case you need to restore it, and it maintains a service history.

Related reference

[“IBM WebSphere MQ Hypervisor Edition script packages” on page 60](#)

[“IBM WebSphere MQ Hypervisor Edition command scripts” on page 61](#)

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image for VMware ESX hypervisor

The IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux virtual image provides the operating system and product binary files that are required to create virtual system instances that run in VMware ESX hypervisor.

Available virtual images

The following virtual machine image is available for download from the [Passport Advantage and Passport Advantage Express web site](#).

- IBM WebSphere MQ Hypervisor Edition for Red Hat Enterprise Linux Version 7.5 64 bit Red Hat Enterprise Linux Server.

After the virtual image has been added to the VMware ESX hypervisor data store, you can create, manage, and deploy virtual images to a cloud.

Virtual image contents

The virtual image contains the following preinstalled software:

- 64 bit Red Hat Enterprise Linux Server
- IBM WebSphere MQ Version 7.5
- IBM WebSphere MQ Explorer

Maintenance

Maintenance for the IBM WebSphere MQ virtual image is applied directly to IBM WebSphere MQ running in VMware ESX hypervisor, in the same way as maintenance is applied on a physical server.

Related tasks

[“Installing” on page 12](#)

Download the latest version of the IBM WebSphere MQ Hypervisor editions from Passport Advantage. Uncompress the installation files and add the files you require onto the target server.

[“Loading the IBM WebSphere MQ virtual image for VMware ESX hypervisor” on page 20](#)

Save a copy of the IBM WebSphere MQ virtual image for VMware ESX hypervisor in the VMware ESX hypervisor data store.

[“Deploying an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor” on page 38](#)
Deploy an IBM WebSphere MQ virtual image directly to VMware ESX hypervisor for the first time.

Glossary

This glossary includes terms and definitions for IBM WebSphere MQ.

The following cross-references are used in this glossary:

- See refers you from a term to a preferred synonym, or from an acronym or abbreviation to the defined full form.
- See also refers you to a related or contrasting term.

[“A” on page 67](#) [“B” on page 70](#) [“C” on page 71](#) [“D” on page 75](#) [“E” on page 77](#) [“F” on page 79](#) [“G” on page 80](#) [“H” on page 80](#) [“I” on page 81](#) [“J” on page 83](#) [“K” on page 83](#) [“L” on page 84](#) [“M” on page 85](#) [“N” on page 89](#) [“O” on page 90](#) [“P” on page 91](#) [“Q” on page 94](#) [“R” on page 95](#) [“S” on page 98](#) [“T” on page 103](#) [“U” on page 105](#) [“V” on page 106](#) [“W” on page 106](#) [“X” on page 107](#)

A

abend reason code

A 4-byte hexadecimal code that uniquely identifies a problem with a program that runs on the z/OS operating system..

abstract class

In object-oriented programming, a class that represents a concept; classes derived from it represent implementations of the concept. An object cannot be constructed from an abstract class; that is, it cannot be instantiated. See also [parent class](#).

access control

In computer security, the process of ensuring that users can access only those resources of a computer system for which they are authorized.

access control list (ACL)

In computer security, a list associated with an object that identifies all the subjects that can access the object and their access rights.

accountability

The quality of being responsible for one's actions.

ACL

See [access control list](#).

active log

A data set with a fixed size where recovery events are recorded as they occur. When the active log is full, the contents of the active log are copied to the archive log.

active queue manager instance

The instance of a running multi-instance queue manager that is processing requests. There is only one active instance of a multi-instance queue manager.

adapter

An intermediary software component that allows two other software components to communicate with one another.

address space (ASID)

The range of addresses available to a computer program or process. Address space can refer to physical storage, virtual storage, or both. See also [allied address space](#), [buffer pool](#).

administration bag

In the WebSphere MQ Administration Interface (MQAI), a type of data bag that is created for administering WebSphere MQ by implying that it can change the order of data items, create lists, and check selectors within a message.

administrative topic object

An object that allows you to assign specific, non-default attributes to topics.

administrator command

A command used to manage WebSphere MQ objects, such as queues, processes, and namelists.

Advanced Program-to-Program Communication (APPC)

An implementation of the SNA LU 6.2 protocol that allows interconnected systems to communicate and share the processing of programs.

advanced telemetry client

See [telemetry advanced client](#).

affinity

An association between objects that have some relationship or dependency upon each other.

alert

A message or other indication that signals an event or an impending event.

alert monitor

In WebSphere MQ for z/OS, a component of the CICS adapter that handles unscheduled events occurring as a result of connection requests to WebSphere MQ for z/OS.

alias queue

A WebSphere MQ object, the name of which is an alias for a base queue or topic that is defined to the local queue manager. When an application or a queue manager uses an alias queue, the alias name is resolved and the requested operation is performed on the associated base object.

alias queue object

A WebSphere MQ object, the name of which is an alias for a base queue defined to the local queue manager. When an application or a queue manager uses an alias queue, the alias name is resolved and the requested operation is performed on the associated base queue.

allied address space

A z/OS address space that is connected to WebSphere MQ for z/OS.

ally

See [allied address space](#).

alternate user authority

The ability of a user ID to supply a different user ID for security checks. When an application opens a WebSphere MQ object, it can supply a user ID on the MQOPEN, MQPUT1, or MQSUB call that the queue manager uses for authority checks instead of the one associated with the application.

alternate user security

On z/OS, the authority checks that are performed when an application requests alternate user authority when opening a WebSphere MQ object.

APAR

See [authorized program analysis report](#).

APF

See [authorized program facility](#).

API-crossing exit

A user written program that is similar in concept to an API exit. It is supported only for CICS applications on WebSphere MQ for z/OS.

API exit

A user-written program that monitors or modifies the function of an MQI call. For each MQI call issued by an application, the API exit is invoked before the queue manager starts to process the call and again after the queue manager has completed processing the call. The API exit can inspect and modify any of the parameters on the MQI call.

APPC

See [Advanced Program-to-Program Communication](#).

application-defined format

Application data in a message for which the user application defines the meaning. See also [built-in format](#).

application environment

The environment that includes the software and the server or network infrastructure that supports it.

application level security

The security services that are invoked when an application issues an MQI call.

application log

In Windows systems, a log that records significant application events.

application queue

A local queue which, when it has triggering set on and when the triggering conditions are met, requires that trigger messages are written.

archive log

A data set on a storage device to which WebSphere MQ copies the contents of each active log data set when the active log reaches its size limit. See also [recovery log](#).

ARM

See [automatic restart manager](#).

ASID

See address space.

asymmetric key cryptography

A system of cryptography that uses two keys: a public key known to everyone and a private key known only to the receiver or sender of the message. See also [symmetric key cryptography](#).

asynchronous consumption

A process that uses a set of MQI calls that allow an application to consume messages from a set of queues. Messages are delivered to the application by invoking a unit of code identified by the application, passing either the message or a token representing the message.

asynchronous messaging

A method of communication between programs in which a program places a message on a message queue, then proceeds with its own processing without waiting for a reply to its message. See also [synchronous messaging](#).

asynchronous put

A put of a message by an application, without waiting for a response from the queue manager.

attribute

1. In object oriented programming, a property of an object or class that can be distinguished distinctly from any other properties. Attributes often describe state information.
2. A characteristic or trait of an entity that describes the entity; for example, the telephone number of an employee is one of the employee attributes. See also [entity](#).

authentication

A security service that provides proof that a user of a computer system is genuinely who that person claims to be. Common mechanisms for implementing this service are passwords and digital signatures.

authentication information object

An object that provides the definitions needed to check certificate revocation lists (CRLs) using LDAP servers, in support for Secure Sockets Layer (SSL) security.

authority check

See [authorization check](#).

authorization

The process of granting a user, system, or process either complete or restricted access to an object, resource, or function.

authorization check

A security check that is performed when a user or application attempts to access a system resource; for example, when an administrator attempts to issue a command to administer WebSphere MQ or when an application attempts to connect to a queue manager.

authorization file

A file that provides security definitions for an object, a class of objects, or all classes of objects.

authorization service

In WebSphere MQ on UNIX and Linux systems and WebSphere MQ for Windows, a service that provides authority checking of commands and MQI calls for the user identifier associated with the command or call.

authorized program analysis report (APAR)

A request for correction of a defect in a supported release of a program supplied by IBM.

authorized program facility (APF)

In a z/OS environment, a facility that permits the identification of programs that are authorized to use restricted functions.

automatic restart manager (ARM)

A z/OS recovery function that can automatically restart batch jobs and started tasks after they or the system on which they are running end unexpectedly.

B**backout**

An operation that reverses all changes to resources made during the current unit of work. See also [commit](#).

bag

See [data bag](#).

bar

A z/OS memory limit, which in 64-bit systems is set at 2GB. The bar separates storage below the 2-gigabyte address from storage above the 2-gigabyte address. The area above the bar is intended for data; no programs run above the bar.

basic mapping support (BMS)

An interface between CICS and application programs that formats input and output display data and routes multiple-page output messages without regard for control characters used by various terminals.

behavior

In object-oriented programming, the functionality embodied within a method.

BMS

See [basic mapping support](#).

Booch methodology

An object-oriented methodology that helps users design systems using the object-oriented paradigm.

bootstrap data set (BSDS)

A VSAM data set that contains an inventory of all active and archived log data sets known to WebSphere MQ for z/OS, and a wrap-around inventory of all recent WebSphere MQ for z/OS activity. The BSDS is required to restart the WebSphere MQ for z/OS subsystem.

browse

In message queuing, to copy a message without removing it from the queue. See also [get](#), [put](#).

browse cursor

In message queuing, an indicator used when browsing a queue to identify the message that is next in sequence.

BSDS

See [bootstrap data set](#).

buffer pool

An area of memory into which data pages are read and in which they are modified and held during processing. See also [address space](#).

built-in format

Application data in a message for which the queue manager defines the meaning. See also [application-defined format](#).

C**CA**

See [certificate authority](#).

CAF

See [Client Attachment feature](#).

callback

A message consumer or an event handler routine.

CCDT

See [client channel definition table](#).

CCF

See [channel control function](#).

CCSID

See [coded character set identifier](#).

CDF

See [channel definition file](#).

certificate authority (CA)

A trusted third-party organization or company that issues the digital certificates in response to a certificate signing request. The certificate authority verifies the identity of the individuals who are granted the unique certificate. See also [Secure Sockets Layer](#).

certificate chain

A hierarchy of certificates that are cryptographically related to one another, starting with the personal certificate and ending with root at the top of the chain.

certificate expiration

A digital certificate contains a date range when the certificate is valid. Outside the valid date range, the certificate is said to be "expired".

certificate revocation list (CRL)

A list of certificates that have been revoked before their scheduled expiration date. Certificate revocation lists are maintained by the certificate authority and used, during a Secure Sockets Layer (SSL) handshake to ensure that the certificates involved have not been revoked.

certificate store

The Windows name for a key repository.

certificate signing request (CSR)

A request that contains the public key and subject distinguished name of a utility or organization. Sent to the CA so that the CA issues a digital signature to that utility.

CF

See [coupling facility](#).

CFSTRUCT

A WebSphere MQ object used to describe the queue manager's use of a Coupling Facility list structure

channel

A WebSphere MQ object that defines a communication link between two queue managers (message channel) or between a client and a queue manager (MQI channel). See also [message channel](#), [MQI channel](#).

channel callback

A mechanism that ensures that the channel connection is established to the correct machine. In a channel callback, a sender channel calls back the original requester channel using the sender's definition.

channel control function (CCF)

A program to move messages from a transmission queue to a communication link, and from a communication link to a local queue, together with an operator panel interface to allow the setup and control of channels.

channel definition file (CDF)

A file containing communication channel definitions that associate transmission queues with communication links.

channel event

An event reporting conditions detected during channel operations, such as when a channel instance is started or stopped. Channel events are generated on the queue managers at both ends of the channel.

channel exit program

A user-written program that is called from one of a defined number of places in the processing sequence of a message channel agent (MCA).

channel initiator

A component of WebSphere MQ distributed queuing that monitors the initiation queue to see when triggering criteria have been met and then starts the sender channel.

channel listener

A component of WebSphere MQ distributed queuing that monitors the network for a startup request and then starts the receiving channel.

checkpoint

A place in a program at which a check is made, or at which a recording of data is made to allow the program to be restarted in case of interruption.

CI

See [control interval](#).

CipherSpec

The combination of encryption algorithm and hash function applied to an SSL message after authentication completes.

cipher suite

The combination of authentication, key exchange algorithm, and the Secure Sockets Layer (SSL) cipher specification used for the secure exchange of data.

ciphertext

Data that has been encrypted. Ciphertext is unreadable until it has been converted into plaintext (decrypted) with a key. See also [cleartext](#).

circular logging

In WebSphere MQ on UNIX and Linux systems and WebSphere MQ for Windows, the process of keeping all restart data in a ring of log files. See also [linear logging](#).

CL

See [Command Language](#).

class

In object-oriented design or programming, a model or template that can be used to create objects with a common definition and common properties, operations, and behavior. An object is an instance of a class.

class hierarchy

The relationships between classes that share a single inheritance.

class library

In object-oriented programming, a collection of prewritten classes or coded templates, any of which can be specified and used by a programmer when developing an application.

cleartext

A string of characters sent over a network in readable form. They may be encoded for the purposes of compression, but can easily be decoded. See also [ciphertext](#).

client

A runtime component that provides access to queuing services on a server for local user applications. The queues used by the applications reside on the server. See also [WebSphere MQ MQI client](#), [WebSphere MQ Java client](#), [WebSphere MQ fully-managed .NET client](#).

client application

An application, running on a workstation and linked to a client, that gives the application access to queuing services on a server.

Client Attachment feature (CAF)

An option that supports the attachment of clients to z/OS.

client channel definition table (CCDT)

A file that contains one or more client-connection channel definitions.

client-connection channel type

The type of MQI channel definition associated with a WebSphere MQ client. See also [server-connection channel type](#).

CLUSRCVR

See [cluster-receiver channel](#).

CLUSSDR

See [cluster-sender channel](#).

cluster

In WebSphere MQ, a group of two or more queue managers on one or more computers, providing automatic interconnection, and allowing queues to be advertised among them for load balancing and redundancy.

cluster queue

A local queue that is hosted by a cluster queue manager, and defined as a target for messages being put from an application connected to any queue manager within the cluster. All applications retrieving messages must be locally connected.

cluster queue manager

A queue manager that is a member of a cluster. A queue manager can be a member of more than one cluster.

cluster-receiver channel (CLUSRCVR)

A channel on which a cluster queue manager can receive messages from other queue managers in the cluster, and cluster information from the repository queue managers.

cluster-sender channel (CLUSSDR)

A channel on which a cluster queue manager can send messages to other queue managers in the cluster, and cluster information to the repository queue managers.

cluster topic

An administrative topic that is defined on a cluster queue manager and made available to other queue managers in the cluster.

cluster transmission queue

A transmission queue that holds all messages from a queue manager destined for another queue manager that is in the same cluster. The queue is called SYSTEM.CLUSTER.TRANSMIT.QUEUE.

CMS key database

A CMS key database is the format of the Database supported by Windows systems, UNIX systems, Linux, and the clients of those platforms. Files ending with .kdb are CMS format. The .kdb files contain the certificates and the keys.

coded character set identifier (CCSID)

A 16-bit number that includes a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and other information that uniquely identifies the coded graphic-character representation.

coexistence

The ability of two or more different versions of WebSphere MQ to function on the same computer.

command

A statement used to initiate an action or start a service. A command consists of the command name abbreviation, and its parameters and flags if applicable.

command bag

In the MQAI, a type of bag that is created for administering WebSphere MQ objects, but cannot change the order of data items or create lists within a message.

command event

A notification that an MQSC or PCF command has been executed successfully.

Command Language (CL)

In WebSphere MQ for iSeries, a language that can be used to issue commands, either at the command line or by writing a CL program.

command prefix

1. A 1- to 8-character command identifier. The command prefix distinguishes the command as belonging to an application or subsystem rather than to z/OS.
2. In WebSphere MQ for z/OS, a character string that identifies the queue manager to which WebSphere MQ for z/OS commands are directed, and from which WebSphere MQ for z/OS operator messages are received.

command server

The WebSphere MQ component that reads commands from the system-command input queue, verifies them, and passes valid commands to the command processor.

commit

To apply all the changes made during the current unit of recovery (UR) or unit of work (UOW). After the operation is complete, a new UR or UOW can begin.

common name (CN)

The component in a Distinguished Name (DN) attribute of an X.509 certificate that represents the name normally associated with the owner of the certificate. For people, the CN is usually their actual name. For web servers, the CN is the fully qualified host and domain name of the server. For WebSphere MQ there are no specific requirements on this field, however many administrators use the name of the queue manager.

See also [Distinguished Name](#)

completion code

A return code indicating how a message queue interface (MQI) call has ended.

confidentiality

The security service that protects sensitive information from unauthorized disclosure. Encryption is a common mechanism for implementing this service.

configuration event

Notifications about the attributes of an object. The notifications are generated when the object is created, changed, or deleted and also by explicit requests.

connection affinity

A channel attribute that specifies the client channel definition that client applications use to connect to the queue manager, if multiple connections are available.

connection factory

A set of configuration values that produces connections that enable a Java EE component to access a resource. Connection factories provide on-demand connections from an application to an enterprise information system (EIS) and allow an application server to enroll the EIS in a distributed transaction.

connection handle

The identifier or token by which a program accesses the queue manager to which it is connected.

constructor

In object-oriented programming, a special method used to initialize an object.

consume

To remove a message from a queue and return its contents to the calling application.

consumer

An application that receives and processes messages. See also [message consumer](#).

context security

On z/OS, the authority checks that are performed when an application opens a queue and specifies that it will set the context in messages that it puts on the queue, or pass the context from messages that it has received to messages that it puts on the queue.

control command

In WebSphere MQ on UNIX and Linux systems and WebSphere MQ for Windows, a command that can be entered interactively from the operating system command line. Such a command requires only that the WebSphere MQ product be installed; it does not require a special utility or program to run it.

control interval (CI)

A fixed-length area of direct access storage in which VSAM stores records and creates distributed free space. The control interval is the unit of information that VSAM transmits to or from direct access storage. A control interval always includes an integral number of physical records.

controlled shutdown

See [quiesced shutdown](#).

correlation identifier

A field in a message that provides a means of identifying related messages. Correlation identifiers are used, for example, to match request messages with their corresponding reply message.

coupling facility (CF)

A special logical partition that provides high-speed caching, list processing, and locking functions in a sysplex.

CPF

See [command prefix](#).

CR (certificate request)

Synonym for [certificate signing request](#).

CRL

See [certificate revocation list](#).

cross-system coupling facility (XCF)

A component that provides functions to support cooperation between authorized programs running within a sysplex.

cryptography

Protecting information by transforming it (encrypting it) into an unreadable format, called ciphertext. Only those who possess a secret key can decipher (or decrypt) the message into plaintext.

D

DAE

See [dump analysis and elimination](#).

daemon

A program that runs unattended to perform continuous or periodic functions, such as network control.

data bag

A container of object properties that the MQAI uses in administering queue managers. There are three types of data bag: user (for user data), administration (for administration with assumed options), and command (for administration with no options assumed).

data-conversion interface (DCI)

The WebSphere MQ interface to which customer- or vendor-written programs that convert application data between different machine encodings and CCSIDs must conform. A part of the WebSphere MQ Framework.

data-conversion service

A service that converts application data to the character set and encoding that are required by applications on other platforms.

datagram

A form of asynchronous messaging in which an application sends a message, but does not require a response. See also [request/reply](#).

data integrity

The security service that detects whether there has been unauthorized modification of data, or tampering. The service detects only whether data has been modified; it does not restore data to its original state if it has been modified.

data item

In the MQAI, an item contained within a data bag. This can be an integer item or a character-string item, and a user item or a system item.

DCE

See [Distributed Computing Environment](#).

DCE principal

A user ID that uses the distributed computing environment.

DCI

See [data-conversion interface](#).

DCM

See [Digital Certificate Manager](#).

dead-letter queue (DLQ)

A queue to which a queue manager or application sends messages that cannot be delivered to their correct destination.

dead-letter queue handler

A utility that monitors a dead-letter queue (DLQ) and processes messages on the queue in accordance with a user-written rules table. A sample dead letter queue handler is provided by WebSphere MQ.

decryption

The process of decoding data that has been encrypted into a secret format. Decryption requires a secret key or password.

default object

A definition of an object (for example, a queue) with all attributes defined. If a user defines an object but does not specify all possible attributes for that object, the queue manager uses default attributes in place of any that were not specified.

deferred connection

A pending event that is activated when a CICS subsystem tries to connect to WebSphere MQ for z/OS before it has started.

derivation

In object-oriented programming, the refinement or extension of one class from another.

destination

1. In JMS, an object that specifies where and how messages should be sent and received.
2. An end point to which messages are sent, such as a queue or topic.

Diffie-Hellman key exchange

A public, key-exchange algorithm that is used for securely establishing a shared secret over an insecure channel.

digital certificate

An electronic document used to identify an individual, a system, a server, a company, or some other entity, and to associate a public key with the entity. A digital certificate is issued by a certification authority and is digitally signed by that authority.

Digital Certificate Manager (DCM)

On IBM i systems, the method of managing digital certificates and using them in secure applications on the iSeries server. Digital Certificate Manager requests and processes digital certificates from certification authorities (CAs) or other third-parties.

digital signature

Information that is encrypted with a private key and is appended to a message or object to assure the recipient of the authenticity and integrity of the message or object. The digital signature proves that the message or object was signed by the entity that owns, or has access to, the private key or shared-secret symmetric key.

disconnect

To break the connection between an application and a queue manager.

distinguished name (DN)

A set of name-value pairs (such as CN=person name and C=country) that uniquely identifies an entity in a digital certificate. Note that the Distinguished Name is unique only within the namespace of a given certificate authority. It is entirely possible that certificates with identical distinguished names can be issued by different certificate authorities. Therefore, ensure that a key repository contains as few trusted root CA certificates as possible, preferably no more than one. See also [certificate authority](#), [digital certificate](#), [X509](#).

distributed application

In message queuing, a set of application programs that can each be connected to a different queue manager, but that collectively comprise a single application.

Distributed Computing Environment (DCE)

In network computing, a set of services and tools that supports the creation, use, and maintenance of distributed applications across heterogeneous operating systems and networks.

distributed queue management

In message queuing, the setup and control of message channels to queue managers on other systems.

distribution list

A list of queues to which a message can be put with a single statement.

DLQ

See [dead-letter queue](#).

DN

See [distinguished name](#).

dual logging

A method of recording WebSphere MQ for z/OS activity, where each change is recorded on two data sets, so that if a restart is necessary and one data set is unreadable, the other can be used. See also [single logging](#).

dual mode

See [dual logging](#).

dump analysis and elimination (DAE)

A z/OS service that enables an installation to suppress SVC dumps and ABEND SYSUDUMP dumps that are not needed because they duplicate previously written dumps.

durable subscription

A subscription that is retained when a subscribing application's connection to the queue manager is closed. When the subscribing application disconnects, the durable subscription remains in place and publications continue to be delivered. When the application reconnects, it can use the same subscription by specifying the unique subscription name. See also [nondurable subscription](#).

dynamic queue

A local queue created when a program opens a model queue object.

E**eavesdropping**

A breach of communication security in which the information remains intact, but its privacy is compromised. See also [impersonation](#), [tampering](#).

Eclipse

An open-source initiative that provides independent software vendors (ISVs) and other tool developers with a standard platform for developing plug-compatible application development tools.

encapsulation

In object-oriented programming, the technique that is used to hide the inherent details of an object, function, or class from client programs.

encryption

In computer security, the process of transforming data into an unintelligible form in such a way that the original data either cannot be obtained or can be obtained only by using a decryption process.

enqueue

To put a message or item in a queue.

entity

A user, group, or resource that is defined to a security service, such as RACF

environment variable

A variable that specifies how an operating system or another program runs, or the devices that the operating system recognizes.

ESM

See [external security manager](#).

ESTAE

See [extended specify task abnormal exit](#).

event data

In an event message, the part of the message data that contains information about the event (such as the queue manager name, and the application that gave rise to the event). See also [event header](#).

event header

In an event message, the part of the message data that identifies the event type of the reason code for the event. See also [event data](#).

event message

A message that contains information (such as the category of event, the name of the application that caused the event, and queue manager statistics) relating to the origin of an instrumentation event in a network of WebSphere MQ systems.

event queue

The queue onto which the queue manager puts an event message after it detects an event. Each category of event (queue manager, performance, configuration, instrumentation, or channel event) has its own event queue.

Event Viewer

A tool provided by Windows systems to examine and manage log files.

exception listener

An instance of a class that can be registered by an application and for which the `onException()` method is called to pass a JMS exception to the application asynchronously.

exclusive method

In object-oriented programming, a method that is not intended to exhibit polymorphism; one with specific effect.

extended specify task abnormal exit (ESTAE)

A z/OS macro that provides recovery capability and gives control to the user-specified exit routine for processing, diagnosing an abend, or specifying a retry address.

external security manager (ESM)

A security product that performs security checking on users and resources. RACF is an example of an ESM.

F

failover

An automatic operation that switches to a redundant or standby system in the event of a software, hardware, or network interruption.

FAP

See [Formats and Protocols](#).

FFDC

See [first-failure data capture](#).

FFST

See [First Failure Support Technology](#).

FFST file

See [First Failure Support Technology file](#).

FIFO

See [first-in first-out](#).

FIPS

United States Federal Information Processing Standards

first-failure data capture (FFDC)

1. A problem diagnosis aid that identifies errors, gathers and logs information about these errors, and returns control to the affected runtime software.
2. The IBM i implementation of the FFST architecture providing problem recognition, selective dump of diagnostic data, symptom string generation, and problem log entry.

First Failure Support Technology (FFST)

An IBM architecture that defines a single approach to error detection through defensive programming techniques. These techniques provide proactive (passive until required) problem recognition and a description of diagnostic output required to debug a software problem.

First Failure Support Technology file (FFST file)

A file containing information for use in detecting and diagnosing software problems. In WebSphere MQ, FFST files have a file type of FDC.

first-in first-out (FIFO)

A queuing technique in which the next item to be retrieved is the item that has been in the queue for the longest time.

forced shutdown

A type of shutdown of the CICS adapter where the adapter immediately disconnects from WebSphere MQ for z/OS, regardless of the state of any currently active tasks. See also [quiesced shutdown](#).

format

In message queuing, a term used to identify the nature of application data in a message.

Formats and Protocols (FAP)

In message queuing, a definition of how queue managers communicate with each other, and of how clients communicate with server queue managers.

Framework

In WebSphere MQ, a collection of programming interfaces that allow customers or vendors to write programs that extend or replace certain functions provided in WebSphere MQ products. The interfaces are the following: data conversion interface (DCI), message channel interface (MCI), name service interface (NSI), security enabling interface (SEI), trigger monitor interface (TMI).

friend class

A class in which all member functions are granted access to the private and protected members of another class. It is named in the declaration of another class and uses the keyword friend as a prefix to the class.

FRR

See [functional recovery routine](#).

full repository

A complete set of information about every queue manager in a cluster. This set of information is called the repository or sometimes the full repository and is usually held by two of the queue managers in the cluster. See also [partial repository](#).

function

A named group of statements that can be called and evaluated and can return a value to the calling statement.

functional recovery routine (FRR)

A z/OS recovery and termination manager that enables a recovery routine to gain control in the event of a program interrupt.

G

gateway queue manager

A cluster queue manager that is used to route messages from an application to other queue managers in the cluster.

generalized trace facility (GTF)

A z/OS service program that records significant system events such as I/O interrupts, SVC interrupts, program interrupts, and external interrupts.

Generic Security Services API

See [Generic Security Services application programming interface](#).

Generic Security Services application programming interface (Generic Security Services API, GSS API)

A common application programming interface (API) for accessing security services.

get

In message queuing, to use the MQGET call to remove a message from a queue and return its contents to the calling application. See also [browse](#), [put](#).

globally defined object

On z/OS, an object whose definition is stored in the shared repository. The object is available to all queue managers in the queue-sharing group. See also [locally defined object](#).

global trace

A WebSphere MQ for z/OS trace option where the trace data comes from the entire WebSphere MQ for z/OS subsystem.

global transaction

A recoverable unit of work performed by one or more resource managers in a distributed transaction environment and coordinated by an external transaction manager.

GSS API

See [Generic Security Services application programming interface](#).

GTF

See [generalized trace facility](#).

H

handshake

The exchange of messages at the start of a Secure Sockets Layer session that allows the client to authenticate the server using public key techniques (and, optionally, for the server to authenticate the client) and then allows the client and server to cooperate in creating symmetric keys for encryption, decryption, and detection of tampering.

hardened message

A message that is written to auxiliary (disk) storage so that the message is not lost in the event of a system failure.

header

See [message header](#).

heartbeat

A signal that one entity sends to another to convey that it is still active.

heartbeat flow

A pulse that is passed from a sending message channel agent (MCA) to a receiving MCA when there are no messages to send. The pulse unblocks the receiving MCA, which would otherwise remain in a wait state until a message arrived or the disconnect interval expired.

heartbeat interval

The time, in seconds, that is to elapse between heartbeat flows.

hierarchy

In publish/subscribe messaging topology, a local queue manager connected to a parent queue manager.

HTTP

See [Hypertext Transfer Protocol](#).

Hypertext Transfer Protocol (HTTP)

An Internet protocol that is used to transfer and display hypertext and XML documents on the web.

I

identity context

Information that identifies the user of the application that puts the message on a queue first.

identification

The security service that enables each user of a computer system to be identified uniquely. A common mechanism for implementing this service is to associate a user ID with each user.

identity context

Information that identifies the user of the application that first puts the message on a queue

IFCID

See [instrumentation facility component identifier](#).

ILE

See [Integrated Language Environment](#).

immediate shutdown

In WebSphere MQ, a shutdown of a queue manager that does not wait for applications to disconnect. Current message queue interface (MQI) calls are allowed to complete, but new MQI calls fail after an immediate shutdown has been requested. See also [preemptive shutdown](#), [quiesced shutdown](#).

impersonation

A breach of communication security in which the information is passed to a person posing as the intended receiver or information is sent by a person posing as someone else. See also [eavesdropping](#), [tampering](#).

inbound channel

A channel that receives messages from another queue manager.

in-built format

See [built-in format](#).

index

In the WebSphere MQ Administration Interface (MQAI), a means of referencing data items.

in-doubt unit of recovery

The status of a unit of recovery for which a syncpoint has been requested but not yet confirmed.

inflight

The state of a resource or unit of recovery that has not yet completed the prepare phase of the commit process.

inheritance

An object-oriented programming technique in which existing classes are used as a basis for creating other classes. Through inheritance, more specific elements incorporate the structure and behavior of more general elements.

initialization input data set

A data set used by WebSphere MQ for z/OS when it starts up.

initiation queue

A local queue on which the queue manager puts trigger messages.

initiator

In distributed queueing, a program that requests network connections on another system. See also [responder](#).

input parameter

A parameter of an MQI call in which you supply information when you make the call.

insertion order

In the WebSphere MQ Administration Interface (MQAI), the order that data items are placed into a data bag.

installable service

In WebSphere MQ on UNIX and Linux systems and WebSphere MQ for Windows, additional functionality provided as independent component. The installation of each component is optional: in-house or third-party components can be used instead.

instance

A specific occurrence of an object that belongs to a class. See also [object](#).

instance data

In object-oriented programming, state information associated with an object.

instrumentation event

A way of monitoring queue manager resource definitions, performance conditions, and channel conditions in a network of WebSphere MQ systems.

instrumentation facility component identifier (IFCID)

In Db2 for z/OS, a value that names and identifies a trace record of an event. As a parameter on the START TRACE and MODIFY TRACE commands, it specifies that the corresponding event is to be traced.

Integrated Language Environment (ILE)

A set of constructs and interfaces that provides a common runtime environment and run-time bindable application program interfaces (APIs) for all ILE-conforming high-level languages.

Interactive Problem Control System (IPCS)

A component of MVS and z/OS that permits online problem management, interactive problem diagnosis, online debugging for disk-resident abend dumps, problem tracking, and problem reporting.

Interactive System Productivity Facility (ISPF)

An IBM licensed program that serves as a full-screen editor and dialog manager. Used for writing application programs, it provides a means of generating standard screen panels and interactive dialogs between the application programmer and terminal user.

Intermediate certificate

A signer certificate that is not the root certificate.

interface

In object-oriented programming, an abstract model of behavior; a collection of functions or methods.

Internet Protocol (IP)

A protocol that routes data through a network or interconnected networks. This protocol acts as an intermediary between the higher protocol layers and the physical network. See also [Transmission Control Protocol](#).

interprocess communication (IPC)

The process by which programs send messages to each other. Sockets, semaphores, signals, and internal message queues are common methods of interprocess communication. See also [client](#).

intersystem communication (ISC)

A CICS facility that provides inbound and outbound support for communication from other computer systems.

IP

See [Internet Protocol](#).

IPC

See [interprocess communication](#).

IPCS

See [Interactive Problem Control System](#).

ISC

See [intersystem communication](#).

ISPF

See [Interactive System Productivity Facility](#).

J**JAAS**

See [Java Authentication and Authorization Service](#).

Java Authentication and Authorization Service (JAAS)

In Java EE technology, a standard API for performing security-based operations. Through JAAS, services can authenticate and authorize users while enabling the applications to remain independent from underlying technologies.

Java Message Service (JMS)

An application programming interface that provides Java language functions for handling messages. See also [Message Queue Interface](#).

Java runtime environment (JRE)

A subset of a Java developer kit that contains the core executable programs and files that constitute the standard Java platform. The JRE includes the Java virtual machine (JVM), core classes, and supporting files.

JMS

See [Java Message Service](#).

JMSAdmin

An administration tool that enables administrators to define the properties of JMS objects and to store them within a JNDI namespace

journal

A feature of OS/400 that WebSphere MQ for iSeries uses to control updates to local objects. Each queue manager library contains a journal for that queue manager.

JRE

See [Java runtime environment](#).

K**keepalive**

A TCP/IP mechanism where a small packet is sent across the network at predefined intervals to determine whether the socket is still working correctly.

Kerberos

A network authentication protocol that is based on symmetric key cryptography. Kerberos assigns a unique key, called a ticket, to each user who logs on to the network. The ticket is embedded in messages that are sent over the network. The receiver of a message uses the ticket to authenticate the sender.

key authentication

See [authentication](#).

key repository

Generic term for a store for digital certificates and their associated keys. Different types of key repository include Certificate Management System (CMS), Java Keystore (JKS), Java Cryptography Extension Keystore (JCEKS), Public Key Cryptography Standard 12 (PKCS12) Keystore, and RACF key

rings. When it is important to differentiate between key repository types, the documentation refers to the key repository type by its specific name. In contexts applicable to multiple keystore types, the generic term key repository is used.

key ring

In computer security, a file that contains public keys, private keys, trusted roots, and certificates.

key store

The place for a private key and corresponding personal certificate. See also [trust store](#)

L**last will and testament**

An object that is registered by a client with a monitor, and used by the monitor if the client ends unexpectedly.

LDAP

See [Lightweight Directory Access Protocol](#).

Lightweight Directory Access Protocol (LDAP)

An open protocol that uses TCP/IP to provide access to directories that support an X.500 model and that does not incur the resource requirements of the more complex X.500 Directory Access Protocol (DAP). For example, LDAP can be used to locate people, organizations, and other resources in an Internet or intranet directory.

linear logging

In WebSphere MQ on UNIX and Linux systems, and WebSphere MQ for Windows, the process of keeping restart data in a sequence of files. New files are added to the sequence as necessary. The space in which the data is written is not reused. See also [circular logging](#).

link level security

The security services that are invoked, directly or indirectly, by a message channel agent (MCA), the communications subsystem, or a combination of the two working together.

listener

A program that detects incoming requests and starts the associated channel.

local definition of a remote queue

A WebSphere MQ object belonging to a local queue manager that defines the attributes of a queue that is owned by another queue manager. In addition, it is used for queue-manager aliasing and reply-to-queue aliasing.

locale

A setting that identifies language or geography and determines formatting conventions such as collation, case conversion, character classification, the language of messages, date and time representation, and numeric representation.

locally defined object

On z/OS, an object whose definition is stored on page set zero. The definition can be accessed only by the queue manager that defined it. See also [globally defined object](#).

local queue

A queue that belongs to the local queue manager. A local queue can contain a list of messages waiting to be processed. See also [remote queue](#).

local queue manager

The queue manager to which the program is connected and that provides message queuing services to the program. See also [remote queue manager](#).

log

In WebSphere MQ, a file recording the work done by queue managers while they receive, transmit, and deliver messages, to enable them to recover in the event of failure.

log control file

In WebSphere MQ on UNIX and Linux systems, and WebSphere MQ for Windows, the file containing information needed to monitor the use of log files (for example, their size and location, and the name of the next available file).

log file

In WebSphere MQ on UNIX and Linux systems, and WebSphere MQ for Windows, a file in which all significant changes to the data controlled by a queue manager are recorded. If the primary log files become full, WebSphere MQ allocates secondary log files.

logical unit (LU)

An access point through which a user or application program accesses the SNA network to communicate with another user or application program.

logical unit 6.2 (LU 6.2)

An SNA logical unit that supports general communication between programs in a distributed processing environment.

logical unit of work identifier (LUWID)

A name that uniquely identifies a thread within a network. This name consists of a fully qualified logical unit network name, a logical unit of work instance number, and a logical unit of work sequence number.

log record

A set of data that is treated as a single unit in a log file.

log record sequence number (LRSN)

A unique identifier for a log record that is associated with a data sharing member. Db2 for z/OS uses the LRSN for recovery in the data sharing environment.

LRSN

See [log record sequence number](#).

LU

See [logical unit](#).

LU 6.2

See [logical unit 6.2](#).

LU 6.2 conversation

In SNA, a logical connection between two transaction programs over an LU 6.2 session that enables them to communicate with each other.

LU 6.2 conversation level security

In SNA, a conversation level security protocol that enables a partner transaction program to authenticate the transaction program that initiated the conversation. LU 6.2 conversation level security is also known as end user verification.

LU 6.2 session

In SNA, a session between two logical units (LUs) of type 6.2.

LU name

The name by which VTAM refers to a node in a network.

LUWID

See [logical unit of work identifier](#).

M

managed destination

A queue that is provided by the queue manager, as the destination to which published messages are to be sent, for an application that elects to use a managed subscription. See also [managed subscription](#).

managed handle

An identifier that is returned by the MQSUB call when a queue manager is specified to manage the storage of messages that are sent to the subscription.

managed subscription

A subscription for which the queue manager creates a subscriber queue to receive publications because the application does not require a specific queue to be used. See also [managed destination](#).

marshalling

See [serialization](#).

MCA

See [message channel agent](#).

MCI

See [message channel interface](#).

media image

In WebSphere MQ on UNIX and Linux systems and WebSphere MQ for Windows, the sequence of log records that contain an image of an object. The object can be re-created from this image.

message

1. A communication sent from a person or program to another person or program.
2. In system programming, information intended for the terminal operator or system administrator.

message affinity

The relationship between conversational messages that are exchanged between two applications, where the messages must be processed by a particular queue manager or in a particular sequence.

message channel

In distributed message queuing, a mechanism for moving messages from one queue manager to another. A message channel comprises two message channel agents (a sender at one end and a receiver at the other end) and a communication link. See also [channel](#).

message channel agent (MCA)

A program that transmits prepared messages from a transmission queue to a communication link, or from a communication link to a destination queue. See also [Message Queue Interface](#).

message channel interface (MCI)

The WebSphere MQ interface to which customer- or vendor-written programs that transmit messages between a WebSphere MQ queue manager and another messaging system must conform. A part of the WebSphere MQ Framework. See also [Message Queue Interface](#).

message consumer

1. A program or function that gets and processes messages. See also [consumer](#).
2. In JMS, an object that is created within a session to receive messages from a destination.

message context

Information about the originator of a message that is held in fields in the message descriptor. There are two categories of context information: identity context and origin context.

message descriptor

Control information describing the message format and presentation that is carried as part of a WebSphere MQ message. The format of the message descriptor is defined by the MQMD structure.

message exit

A type of channel exit program that is used to modify the contents of a message. Message exits usually work in pairs, one at each end of a channel. At the sending end of a channel, a message exit is called after the message channel agent (MCA) has got a message from the transmission queue. At the receiving end of a channel, a message exit is called before the message channel agent (MCA) puts a message on its destination queue.

message flow control

A distributed queue management task that involves setting up and maintaining message routes between queue managers.

Message Format Service (MFS)

An IMS editing facility that allows application programs to deal with simple logical messages instead of device-dependent data, thus simplifying the application development process.

message group

A logical group of related messages. The relationship is defined by the application putting the messages, and ensures that the messages will be retrieved in the sequence put if both the producer and consumer honor the grouping.

message handle

A reference to a message. The handle can be used to obtain access to the message properties of the message.

message header

The part of a message that contains control information such as a unique message ID, the sender and receiver of the message, the message priority, and the type of message.

message input descriptor (MID)

The Message Format Service (MFS) control block that describes the format of the data presented to the application program. See also [message output descriptor](#).

message listener

An object that acts as an asynchronous message consumer.

message output descriptor (MOD)

The Message Format Service (MFS) control block that describes the format of the output data produced by the application program. See also [message input descriptor](#).

message priority

In WebSphere MQ, an attribute of a message that can affect the order in which messages on a queue are retrieved, and whether a trigger event is generated.

message producer

In JMS, an object that is created by a session and that is used to send messages to a destination.

message property

Data associated with a message, in name-value pair format. Message properties can be used as message selectors to filter publications or to selectively get messages from queues. Message properties can be used to include business data or state information about processing without having to alter the message body.

Message Queue Interface (MQI)

The programming interface provided by WebSphere MQ queue managers. The programming interface allows application programs to access message queuing services. See also [Java Message Service](#), [message channel agent](#), [message channel interface](#).

message queue management (MQM)

In WebSphere MQ for HP NonStop Server, a facility that provides access to PCF command formats and control commands to manage queue managers, queues, and channels.

message queuing

A programming technique in which each program within an application communicates with the other programs by putting messages on queues.

message-retry

An option available to an MCA that is unable to put a message. The MCA can wait for a predefined amount of time and then try to put the message again.

message segment

One of a number of segments of a message that is too large either for the application or for the queue manager to handle.

message selector

In application programming, a variable-length string that is used by an application to register its interest in only those messages whose properties satisfy the Structured Query Language (SQL) query that the selection string represents. The syntax of a message selector is based on a subset of the SQL92 conditional expression syntax.

message sequence numbering

A programming technique in which messages are given unique numbers during transmission over a communication link. This enables the receiving process to check whether all messages are received, to place them in a queue in the original order, and to discard duplicate messages.

message token

A unique identifier of a message within an active queue manager.

method

In object-oriented design or programming, the software that implements the behavior specified by an operation.

MFS

See [Message Format Service](#).

MGAS

See [mostly global address space](#).

Microsoft Cluster Server (MSCS)

A technology that provides high availability by grouping computers running Windows into MSCS clusters. If one of the computers in the cluster hits any one of a range of problems, MSCS shuts down the disrupted application in an orderly manner, transfers its state data to another computer in the cluster, and re-initiates the application there.

Microsoft Transaction Server (MTS)

A facility that helps Windows users run business logic applications in a middle tier server. MTS divides work up into activities, which are short independent chunks of business logic.

MID

See [message input descriptor](#).

MOD

See [message output descriptor](#).

model queue object

A set of queue attributes that act as a template when a program creates a dynamic queue.

mostly global address space (MGAS)

A flexible virtual address space model, used in systems such as HP-UX, that preserves most of the address space for shared applications. This can enhance performance for processes that share a lot of data. See also [mostly private address space](#).

mostly private address space (MPAS)

A flexible virtual address space model, used in systems such as HP-UX, that can allocate larger address space blocks to processes. This can enhance performance for processes that require a lot of data space. See also [mostly global address space](#).

MPAS

See [mostly private address space](#).

MQAI

See [WebSphere MQ Administration Interface](#).

MQI

See [Message Queue Interface](#).

MQI channel

A connection between a WebSphere MQ client and a queue manager on a server system. An MQI channel transfers only MQI calls and responses in a bidirectional manner. See also [channel](#).

MQM

See [message queue management](#).

MQSC

See [WebSphere MQ script commands](#).

MQSeries

A previous name for WebSphere MQ.

MQ Telemetry Transport

MQ Telemetry Transport (MQTT) is an open, lightweight publish/subscribe protocol flowing over TCP/IP to connect large numbers of devices such as servos, actuators, smart phones, vehicles, homes, health, remote sensors, and control devices. MQTT is designed to work in environments where the network might be constrained by bandwidth, or the device might be constrained by memory or processors for example.

MQTT

See [MQ Telemetry Transport](#).

MQTT client

An MQTT client application connects to MQTT capable servers such as WebSphere MQ Telemetry channels. You can write your own clients to use the published protocol, or use one of the clients supplied with the installation of WebSphere MQ Telemetry. A typical client is responsible for collecting information from a telemetry device and publishing the information to the server. It can also subscribe to topics, receive messages, and use this information to control the telemetry device. Some clients are provided with WebSphere MQ Telemetry; see [Telemetry clients](#) and [Telemetry advanced clients](#).

MQTT server

An MQTT server handles the server side of the MQTT protocol. It typically allows many MQTT clients to connect to it at the same time, and provides a hub for messages distribution to the MQTT clients. A WebSphere MQ queue manager with the telemetry (MQXR) service is an MQTT server.

MSCS

See [Microsoft Cluster Server](#).

MTS

See [Microsoft Transaction Server](#).

multi-hop

To pass through one or more intermediate queue managers when there is no direct communication link between a source queue manager and the target queue manager.

multi-instance queue manager

A queue manager that is configured to share the use of queue manager data with other queue manager instances. One instance of a running multi-instance queue manager is active, other instances are on standby ready to take over from the active instance. See also [single instance queue manager](#).

N**namelist**

A WebSphere MQ object that contains a list of object names, for example, queue names.

name service

In WebSphere MQ on UNIX and Linux systems and WebSphere MQ for Windows, the facility that determines which queue manager owns a specified queue.

name service interface (NSI)

The WebSphere MQ interface to which customer- or vendor-written programs that resolve queue-name ownership must conform. A part of the WebSphere MQ Framework.

name transformation

In WebSphere MQ on UNIX and Linux systems and WebSphere MQ for Windows, an internal process that changes a queue manager name so that it is unique and valid for the system being used. Externally, the queue manager name remains unchanged.

nested bag

In the WebSphere MQ Administration Interface (MQAI), a system bag that is inserted into another data bag

nesting

In the WebSphere MQ Administration Interface (MQAI), a means of grouping information returned from WebSphere MQ.

NetBIOS (Network Basic Input/Output System)

A standard interface to networks and personal computers that is used on local area networks to provide message, print-server, and file-server functions. Application programs that use NetBIOS do not have to handle the details of LAN data link control (DLC) protocols.

Network Basic Input/Output System

See [NetBIOS](#).

New Technology File System (NTFS)

One of the native file systems in Windows operating environments.

node

In Microsoft Cluster Server (MSCS), each computer in the cluster.

nondurable subscription

A subscription that exists only as long as the subscribing application's connection to the queue manager remains open. The subscription is removed when the subscribing application disconnects from the queue manager either deliberately or by loss of connection. See also [durable subscription](#).

nonpersistent message

A message that does not survive a restart of the queue manager. See also [persistent message](#).

NSI

See [name service interface](#).

NTFS

See [New Technology File System](#).

NUL

See [null character](#).

null character (NUL)

A control character with the value of X'00' that represents the absence of a displayed or printed character.

O

OAM

See [object authority manager](#).

object

1. In WebSphere MQ, a queue manager, queue, process definition, channel, namelist, authentication information object, administrative topic object, listener, service object, or (on z/OS only) a CF structure object or storage class.
2. In object-oriented design or programming, a concrete realization (instance) of a class that consists of data and the operations associated with that data. An object contains the instance data that is defined by the class, but the class owns the operations that are associated with the data.

object authority manager (OAM)

In WebSphere MQ on UNIX and Linux systems, WebSphere MQ for IBM i, and WebSphere MQ for Windows, the default authorization service for command and object management. The OAM can be replaced by, or run in combination with, a customer-supplied security service.

object descriptor

A data structure that identifies a particular WebSphere MQ object. Included in the descriptor are the name of the object and the object type.

object handle

The identifier or token by which a program accesses the WebSphere MQ object with which it is working.

object-oriented programming

A programming approach based on the concepts of data abstraction and inheritance. Unlike procedural programming techniques, object-oriented programming concentrates not on how something is accomplished but instead on what data objects comprise the problem and how they are manipulated.

OCSP

Online Certificate Status Protocol. A method of checking if a certificate is revoked.

offloading

In WebSphere MQ for z/OS, an automatic process whereby a queue manager's active log is transferred to its archive log.

one way authentication

In this method of authentication, the queue manager presents the certificate to the client, but the authentication is not checked from the client to the queue manager.

open

To establish access to an object, such as a queue or a topic

open systems interconnection (OSI)

The interconnection of open systems in accordance with standards of the International Organization for Standardization (ISO) for the exchange of information.

Open Transaction Manager Access (OTMA)

A component of IMS that implements a transaction-based, connectionless client/server protocol in an MVS sysplex environment. The domain of the protocol is restricted to the domain of the z/OS Cross-System Coupling Facility (XCF). OTMA connects clients to servers so that the client can support a large network (or a large number of sessions) while maintaining high performance.

OPM

See [original program model](#).

original program model (OPM)

The set of functions for compiling source code and creating high-level language programs before the Integrated Language Environment (ILE) model was introduced.

OSGi Alliance

A consortium of more than 20 companies, including IBM, that creates specifications to outline open standards for the management of voice, data and multimedia wireless and wired networks.

OSI

See [open systems interconnection](#).

OSI directory standard

The standard, known as X.500, that defines a comprehensive directory service, including an information model, a namespace, a functional model, and an authentication framework. X.500 also defines the Directory Access Protocol (DAP) used by clients to access the directory. The Lightweight Directory Access Protocol (LDAP) removes some of the burden of X.500 access from directory clients, making the directory available to a wider variety of machines and applications.

OTMA

See [Open Transaction Manager Access](#).

outbound channel

A channel that takes messages from a transmission queue and sends them to another queue manager.

output log-buffer

In WebSphere MQ for z/OS, a buffer that holds recovery log records before they are written to the archive log.

output parameter

A parameter of an MQI call in which the queue manager returns information when the call completes or fails.

overloading

In object-oriented programming, the capability of an operator or method to have different meanings depending on the context. For example, in C++, a user can redefine functions and most standard operators when the functions and operators are used with class types. The method name or operator remains the same, but the method parameters differ in type, number, or both. This difference is collectively called the function's or the operator's signature and each signature requires a separate implementation.

P

page set

A VSAM data set used when WebSphere MQ for z/OS moves data (for example, queues and messages) from buffers in main storage to permanent backing storage (DASD).

parent class

A class from which another class inherits instance methods, attributes, and instance variables. See also [abstract class](#).

partial repository

A partial set of information about queue managers in a cluster. A partial repository is maintained by all cluster queue managers that do not host a full repository. See also [full repository](#).

partner queue manager

See [remote queue manager](#).

PassTicket

In RACF secured sign-on, a dynamically generated, random, one-time-use, password substitute that a workstation or other client can use to sign on to the host rather than sending a RACF password across the network.

PCF

See [programmable command format](#).

pending event

An unscheduled event that occurs as a result of a connect request from a CICS adapter.

percolation

In error recovery, the passing along a preestablished path of control from a recovery routine to a higher-level recovery routine.

performance event

A category of event indicating that a limit condition has occurred.

performance trace

A WebSphere MQ trace option where the trace data is to be used for performance analysis and tuning.

permanent dynamic queue

A dynamic queue that is deleted when it is closed only if deletion is explicitly requested. Permanent dynamic queues are recovered if the queue manager fails, so they can contain persistent messages. See also [temporary dynamic queue](#).

persistent message

A message that survives a restart of the queue manager. See also [nonpersistent message](#).

personal certificate

Certificate for which you own the corresponding private key. Associated with queue managers or applications.

PGM

See [Pragmatic General Multicast](#).

PID

See [process ID](#).

ping

The command that sends an Internet Control Message Protocol (ICMP) echo-request packet to a gateway, router, or host with the expectation of receiving a reply.

PKCS

Public Key Cryptography Standards. A set of standards for cryptography, of which:

- 7 is for messages
- 11 is for hardware security modules
- 12 is for the file format used in the key repository

PKI

See [public key infrastructure](#).

plain text

See [cleartext](#).

point of recovery

In WebSphere MQ for z/OS, a set of backup copies of WebSphere MQ for z/OS page sets and the corresponding log data sets required to recover these page sets. These backup copies provide a potential restart point in the event of page set loss (for example, page set I/O error).

poison message

In a queue, an incorrectly formatted message that the receiving application cannot process. The message can be repeatedly delivered to the input queue and repeatedly backed out by the application.

polymorphism

An object-oriented programming characteristic that allows a method to perform differently, depending on the class that implements it. Polymorphism allows a subclass to override an inherited method without affecting the parent class's method. Polymorphism also enables a client to access two or more implementations of an object from a single interface.

Pragmatic General Multicast (PGM)

A reliable multicast transport protocol that provides a reliable sequence of packets to multiple recipients simultaneously.

preemptive shutdown

In WebSphere MQ, a shutdown of a queue manager that does not wait for connected applications to disconnect, or for current MQI calls to complete. See also [immediate shutdown](#), [quiesced shutdown](#).

preferred computer

The primary computer used by an application running under Microsoft Cluster Server control. After a failover to another computer, MSCS monitors the preferred computer until it is repaired, and as soon as it is running correctly again, moves the application back to it.

principal

An entity that can communicate securely with another entity. A principal is identified by its associated security context, which defines its access rights.

privately defined object

See [locally defined object](#).

private methods and instance data

In object-oriented programming, methods and instance data that are only accessible to the implementation of the same class.

process definition object

A WebSphere MQ object that contains the definition of a WebSphere MQ application. For example, a queue manager uses the definition when it works with trigger messages.

process ID (PID)

The unique identifier that represents a process. A process ID is a positive integer and is not reused until the process lifetime ends.

producer

An application that creates and sends messages. See also [publisher](#), [message producer](#).

programmable command format (PCF)

A type of WebSphere MQ message used by the following applications: user administration applications, to put PCF commands onto the system command input queue of a specified queue manager, user administration applications, to get the results of a PCF command from a specified queue manager, and a queue manager, as a notification that an event has occurred. See also [WebSphere MQ script commands](#).

program temporary fix (PTF)

For System i, System p, and System z products, a package containing individual or multiple fixes that is made available to all licensed customers. A PTF resolves defects and might provide enhancements.

property

A characteristic of an object that describes the object. A property can be changed or modified. Properties can describe an object name, type, value, or behavior, among other things.

protected methods and instance data

In object-oriented programming, methods and instance data that are only accessible to the implementations of the same or derived classes, or from friend classes.

PTF

See [program temporary fix](#).

public key

The key known to everyone. This key is usually embedded in a digital certificate that specifies the owner of the public key.

public key cryptography

A cryptography system that uses two keys: a public key known to everyone and a private or secret key known only to the recipient of the message. The public and private keys are related in such a way, that anything encrypted with one key can be decrypted only by the corresponding private key.

public key infrastructure (PKI)

A system of digital certificates, certification authorities, and other registration authorities that verify and authenticate the validity of each party involved in a network transaction.

public methods and instance data

In object oriented programming, methods and instance data that are accessible to all classes.

publish

To make information about a specified topic available to a queue manager in a publish/subscribe system.

publisher

An application that makes information about a specified topic available to a broker in a publish/subscribe system.

publish/subscribe

A type of messaging interaction in which information, provided by publishing applications, is delivered by an infrastructure to all subscribing applications that have expressed interest in that type of information.

publish/subscribe cluster

A set of queue managers that are fully interconnected and that form part of a multi-queue manager network for publish/subscribe applications.

put

In message queuing, to use the MQPUT or MQPUT1 calls to place messages on a queue. See also [browse](#), [get](#).

Q

queue

An object that holds messages for message-queueing applications. A queue is owned and maintained by a queue manager.

queue index

In WebSphere MQ for z/OS, a list of message identifiers or a list of correlation identifiers that can be used to increase the speed of MQGET operations on the queue.

queue manager

A component of a message queuing system that provides queuing services to applications.

queue manager event

An event that indicates one of the following: an error condition has occurred in relation to the resources used by a queue manager. For example, a queue is unavailable, or a significant change has occurred in the queue manager. For example, a queue manager has stopped or started.

queue manager group

In a client channel definition table (CCDT), the group of queue managers a client tries to connect to when a connection is established to a server.

queue manager level security

In WebSphere MQ for z/OS, the authorization checks that are performed using RACF profiles specific to a queue manager.

queue manager set

A grouping of queue managers in WebSphere MQ Explorer that allows a user to perform actions on all of the queue managers in the group.

queue-sharing group

In WebSphere MQ for z/OS, a group of queue managers in the same sysplex that can access a single set of object definitions stored in the shared repository, and a single set of shared queues stored in the coupling facility. See also [shared queue](#).

queue-sharing group level security

In WebSphere MQ for z/OS, the authorization checks that are performed using RACF profiles that are shared by all queue managers in a queue-sharing group.

quiesce

To end a process or shut down a system after allowing normal completion of active operations.

quiesced shutdown

1. A type of shutdown of the CICS adapter where the adapter disconnects from WebSphere MQ, but only after all the currently active tasks have been completed. See also [forced shutdown](#).
2. In WebSphere MQ, a shutdown of a queue manager that allows all connected applications to disconnect. See also [immediate shutdown](#), [preemptive shutdown](#).

quiescing

In WebSphere MQ, the state of a queue manager before it stops. In this state, programs are allowed to finish processing, but no new programs are allowed to start.

quorum disk

The disk accessed exclusively by Microsoft Cluster Server to store the cluster recovery log, and to determine whether a server is up or down. Only one server can own the quorum disk at a time. Servers in the cluster can negotiate for the ownership.

R

RACF

See [Resource Access Control Facility](#).

RAID

See [Redundant Array of Independent Disks](#).

RBA

See [relative byte address](#).

RC

See [return code](#).

read ahead

An option that allows messages to be sent to a client before an application requests them.

reason code

A return code that describes the reason for the failure or partial success of a Message Queue Interface (MQI) call.

receive exit

A type of channel exit program that is called just after the message channel agent (MCA) has regained control following a communications receive and has received a unit of data from a communications connection. See also [send exit](#).

receiver channel

In message queuing, a channel that responds to a sender channel, takes messages from a communication link, and puts them on a local queue.

recovery log

In WebSphere MQ for z/OS, data sets containing information needed to recover messages, queues, and the WebSphere MQ subsystem. See also [archive log](#).

recovery termination manager (RTM)

A program that handles all normal and abnormal termination of tasks by passing control to a recovery routine associated with the terminating function.

Redundant Array of Independent Disks (RAID)

A collection of two or more physical disk drives that present to the host an image of one or more logical disk drives. In the event of a physical device failure, the data can be read or regenerated from the other disk drives in the array due to data redundancy.

reference message

A message that refers to a piece of data that is to be transmitted. The reference message is handled by message exit programs, which attach and detach the data from the message so allowing the data to be transmitted without having to be stored on any queues.

registry

A repository that contains access and configuration information for users, systems, and software.

Registry Editor

In Windows, the program item that allows the user to edit the registry.

registry hive

In Windows systems, the structure of the data stored in the registry.

relative byte address (RBA)

The offset of a data record or control interval from the beginning of the storage space that is allocated to the data set or file to which it belongs.

reliable multicast messaging (RMM)

A high-throughput low-latency transport fabric designed for one-to-many data delivery or many-to-many data exchange, in a message-oriented middleware publish/subscribe fashion. RMM exploits the IP multicast infrastructure to ensure scalable resource conservation and timely information distribution.

remote queue

A queue that belongs to a remote queue manager. Programs can put messages on remote queues, but they cannot get messages from remote queues. See also [local queue](#).

remote queue manager

A queue manager to which a program is not connected, even if it is running on the same system as the program. See also [local queue manager](#).

remote queue object

A WebSphere MQ object belonging to a local queue manager. This object defines the attributes of a queue that is owned by another queue manager. In addition, it is used for queue-manager aliasing and reply-to-queue aliasing.

remote queuing

In message queuing, the provision of services to enable applications to put messages on queues belonging to other queue managers.

reply message

A type of message used for replies to request messages. See also [report message](#), [request message](#).

reply-to queue

The name of a queue to which the program that issued an MQPUT call wants a reply message or report message sent.

report message

A type of message that gives information about another message. A report message can indicate that a message has been delivered, has arrived at its destination, has expired, or could not be processed for some reason. See also [reply message](#), [request message](#).

repository

A collection of information about the queue managers that are members of a cluster. This information includes queue manager names, their locations, their channels, and what queues they host.

repository queue manager

A queue manager that hosts the full repository of information about a cluster.

requester channel

In message queuing, a channel that can be started locally to initiate operation of a server channel. See also [server channel](#).

request message

A type of message used to request a reply from another program. See also [reply message](#), [report message](#).

request/reply

A type of messaging application in which a request message is used to request a reply from another application. See also [datagram](#).

RESLEVEL

In WebSphere MQ for z/OS, an option that controls the number of user IDs checked for API-resource security.

resolution path

The set of queues that are opened when an application specifies an alias or a remote queue on input to an MQOPEN call.

resource

A facility of a computing system or operating system required by a job, task, or running program. Resources include main storage, input/output devices, the processing unit, data sets, files, libraries, folders, application servers, and control or processing programs.

Resource Access Control Facility (RACF)

An IBM licensed program that provides access control by identifying users to the system; verifying users of the system; authorizing access to protected resources; logging unauthorized attempts to enter the system; and logging accesses to protected resources.

resource adapter

An implementation of the Java Enterprise Edition Connector Architecture that allows JMS applications and message driven beans, running in an application server, to access the resources of a WebSphere MQ queue manager.

resource manager

An application, program, or transaction that manages and controls access to shared resources such as memory buffers and data sets. WebSphere MQ, CICS, and IMS are resource managers.

Resource Recovery Services (RRS)

A component of z/OS that uses a sync point manager to coordinate changes among participating resource managers.

responder

In distributed queuing, a program that replies to network connection requests from another system. See also [initiator](#).

resynch

In WebSphere MQ, an option to direct a channel to start up and resolve any in-doubt status messages, but without restarting message transfer.

return code (RC)

A value returned by a program to indicate the result of its processing. Completion codes and reason codes are examples of return codes.

return-to-sender

An option available to an MCA that is unable to deliver a message. The MCA can send the message back to the originator.

Rivest-Shamir-Adleman algorithm (RSA)

A public-key encryption technology developed by RSA Data Security, Inc, and used in the IBM implementation of SSL.

RMM

See [reliable multicast messaging](#).

rollback

See [backout](#).

root certificate

The top certificate in the chain. If this is a self-signed certificate, it is used only for signing other certificates. See also [self-signed certificate](#)

RRS

See [Resource Recovery Services](#).

RSA

See [Rivest-Shamir-Adleman algorithm](#).

RTM

See [recovery termination manager](#).

rules table

A control file containing one or more rules that the dead-letter queue handler applies to messages on the dead letter queue (DLQ).

S**Scalable Parallel 2 (SP2)**

IBM's parallel UNIX system: effectively parallel AIX systems on a high-speed network.

SDK

See [software development kit](#).

SDWA

See [system diagnostic work area](#).

SECMEC

See [security mechanism](#).

Secure Sockets Layer (SSL)

A security protocol that provides communication privacy. With SSL, client/server applications can communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery. See also [certificate authority](#).

security enabling interface (SEI)

The WebSphere MQ interface to which customer- or vendor-written programs that check authorization, supply a user identifier, or perform authentication must conform. A part of the WebSphere MQ Framework.

security exit

A channel exit program that is called immediately after the initial data negotiation has completed on channel startup. Security exits normally work in pairs and can be called on both message channels and MQI channels. The primary purpose of the security exit is to enable the message channel agent (MCA) at each end of a channel to authenticate its partner.

security identifier (SID)

On Windows systems, a supplement to the user ID that identifies the full user account details on the Windows security account manager database where the user is defined.

security mechanism (SECMEC)

A technical tool or technique that is used to implement a security service. A mechanism might operate by itself, or in conjunction with others, to provide a particular service. Examples of security mechanisms include access control lists, cryptography, and digital signatures.

security message

One of the messages, sent by security exits that are called at both ends of a channel, to communicate with each other. The format of a security message is not defined and is determined by the user.

security service

A service within a computer system that protect its resources. Access control is an example of a security service.

Security Support Provider Interface (SSI)

The means for networked applications to call one of several security support providers (SSPs) to establish authenticated connections and to exchange data securely over those connections. It is available for use on Windows systems.

self-signed certificate

The digital signature in the certificate is generated using the private key corresponding to the public key in the certificate.

segmentation

The division of a message that is too large for a queue manager, queue, or application, into a number of smaller physical messages, which are then reassembled by the receiving queue manager or application.

SEI

See [security enabling interface](#).

selector

An identifier for a data item. In the WebSphere MQ Administration Interface (MQAI), there are two types of selector: a user selector and a system selector.

semaphore

In UNIX and Linux systems, a general method of communication between two processes that extends the features of signals.

sender channel

In message queuing, a channel that initiates transfers, removes messages from a transmission queue, and moves them over a communication link to a receiver or requester channel.

send exit

A type of channel exit program that is called just before a message channel agent (MCA) issues a communications send to send a unit of data over a communications connection. See also [receive exit](#).

Sequenced Packet Exchange protocol (SPX)

A session-oriented network protocol that provides connection-oriented services between two nodes on the network, and is used primarily by client/server applications. It relies on the Internet Packet Exchange (IPX) protocol, provides flow control and error recovery, and guarantees reliability of the physical network.

sequence number wrap value

In WebSphere MQ, a method of ensuring that both ends of a communication link reset their current message sequence numbers at the same time. Transmitting messages with a sequence number ensures that the receiving channel can reestablish the message sequence when storing the messages.

serialization

In object-oriented programming, the writing of data in sequential fashion to a communications medium from program memory.

server

1. A queue manager that provides queue services to client applications running on a remote workstation.
2. A software program or a computer that provides services to other software programs or other computers. See also [client](#).

server channel

In message queuing, a channel that responds to a requester channel, removes messages from a transmission queue, and moves them over a communication link to the requester channel. See also [requester channel](#).

server-connection channel type

The type of MQI channel definition associated with the server that runs a queue manager. See also [client-connection channel type](#).

service interval

A time interval, against which the elapsed time between a put or a get and a subsequent get is compared by the queue manager in deciding whether the conditions for a service interval event have been met. The service interval for a queue is specified by a queue attribute.

service interval event

An event related to the service interval.

service object

An object that can start additional processes when the queue manager starts and can stop the processes when the queue manager stops.

session

A logical or virtual connection between two stations, software programs, or devices on a network that allows the two elements to communicate and exchange data for the duration of the session.

session ID

In WebSphere MQ for z/OS, the CICS-unique identifier that defines the communication link to be used by a message channel agent when moving messages from a transmission queue to a link.

session-level authentication

In Systems Network Architecture (SNA), a session level security protocol that enables two logical units (LUs) to authenticate each other while they are activating a session. Session level authentication is also known as LU-LU verification.

session-level cryptography

In Systems Network Architecture (SNA), a method of encrypting and decrypting data that flows on a session between two logical units (LUs).

shared inbound channel

In WebSphere MQ for z/OS, a channel that was started by a listener using the group port. The channel definition of a shared channel can be stored either on page set zero (private) or in the shared repository (global).

shared outbound channel

In WebSphere MQ for z/OS, a channel that moves messages from a shared transmission queue. The channel definition of a shared channel can be stored either on page set zero (private) or in the shared repository (global).

shared queue

In WebSphere MQ for z/OS, a type of local queue. The messages on the queue are stored in the coupling facility and can be accessed by one or more queue managers in a queue-sharing group. The definition of the queue is stored in the shared repository. See also [queue-sharing group](#).

shared repository

In WebSphere MQ for z/OS, a shared Db2 database that is used to hold object definitions that have been defined globally.

sharing conversations

The facility for more than one conversation to share a channel instance, or the conversations that share a channel instance.

shell

A software interface between users and an operating system. Shells generally fall into one of two categories: a command line shell, which provides a command line interface to the operating system; and a graphical shell, which provides a graphical user interface (GUI).

SID

See [security identifier](#).

signal

A mechanism by which a process can be notified of, or affected by, an event occurring in the system. Examples of such events include hardware exceptions and specific actions by processes.

signaling

In WebSphere MQ for z/OS and WebSphere MQ for Windows, a feature that allows the operating system to notify a program when an expected message arrives on a queue.

signature

The collection of types associated with a method. The signature includes the type of the return value, if any, as well as the number, order, and type of each of the method's arguments.

signer certificate

A certificate that is used for encipherment or signing.

single instance queue manager

A queue manager that does not have multiple instances. See also [multi-instance queue manager](#).

single logging

A method of recording WebSphere MQ for z/OS activity where each change is recorded on one data set only. See also [dual logging](#).

single-phase backout

A method in which an action in progress must not be allowed to finish, and all changes that are part of that action must be undone.

single-phase commit

A method in which a program can commit updates to a commitment resource without coordinating those updates with updates the program has made to resources controlled by another resource manager.

SIT

See [system initialization table](#).

SMF

See [System Management Facilities](#).

SNA

See [Systems Network Architecture](#).

software development kit (SDK)

A set of tools, APIs, and documentation to assist with the development of software in a specific computer language or for a particular operating environment.

source queue manager

See [local queue manager](#).

SP2

See [Scalable Parallel 2](#).

SPX

See [Sequenced Packet Exchange protocol](#).

SSI

See [Security Support Provider Interface](#).

SSL

See [Secure Sockets Layer](#).

SSLPeer

The value in the issuer represents the distinguished name of the remote personal certificate.

SSL or TLS client

The initiating end of the connection. One outbound channel from a queue manager is also an SSL or TLS client.

standby queue manager instance

An instance of a running multi-instance queue manager ready to take over from the active instance. There are one or more standby instances of a multi-instance queue manager.

stanza

A group of lines in a file that together have a common function or define a part of the system. Stanzas are usually separated by blank lines or colons, and each stanza has a name.

star-connected communications network

A network in which all nodes are connected to a central node.

storage class

In WebSphere MQ for z/OS, the page set that is to hold the messages for a particular queue. The storage class is specified when the queue is defined.

store and forward

The temporary storing of packets, messages, or frames in a data network before they are retransmitted toward their destination.

streaming

In object-oriented programming, the serialization of class information and object instance data.

subscribe

To request information about a topic.

subsystem

In z/OS, a service provider that performs one or many functions but does nothing until a request is made. For example, each WebSphere MQ for z/OS queue manager or instance of a Db2 for z/OS database management system is a z/OS subsystem.

supervisor call (SVC)

An instruction that interrupts the program being run and passes control to the supervisor so that it can perform the specific service indicated by the instruction.

SVC

See [supervisor call](#).

switchover

The change from the active multi-instance queue manager instance to a standby instance. A switchover results from an operator intentionally stopping the active multi-instance queue manager instance.

switch profile

In WebSphere MQ for z/OS, a RACF profile used when WebSphere MQ starts up or when a refresh security command is issued. Each switch profile that WebSphere MQ detects turns off checking for the specified resource.

symmetric key cryptography

A system of cryptography in which the sender and receiver of a message share a single, common, secret key that is used to encrypt and decrypt the message. This system does not offer any authentication. See also [asymmetric key cryptography](#).

symptom string

Diagnostic information displayed in a structured format designed for searching the IBM software support database.

synchronous messaging

A method of communication between programs in which a program places a message on a message queue and then waits for a reply to its message before resuming its own processing. See also [asynchronous messaging](#).

sync point

A point during the processing of a transaction at which protected resources are consistent.

sysplex

A set of z/OS systems that communicate with each other through certain multisystem hardware components and software services.

system bag

A type of data bag that is created by the MQAI.

system control commands

Commands used to manipulate platform-specific entities such as buffer pools, storage classes, and page sets.

system diagnostic work area (SDWA)

In a z/OS environment, the data that is recorded in a SYS1.LOGREC entry that describes a program or hardware error.

system initialization table (SIT)

A table containing parameters used by CICS on start up.

system item

A type of data item that is created by the MQAI.

System Management Facilities (SMF)

A component of z/OS that collects and records a variety of system and job-related information.

system selector

In the WebSphere MQ Administration Interface (MQAI), a system item identifier that is included in the data bag when it is created.

Systems Network Architecture (SNA)

The description of the logical structure, formats, protocols, and operational sequences for transmitting information through and controlling the configuration and operation of networks.

T

tampering

A breach of communication security in which information in transit is changed or replaced and then sent on to the recipient. See also [eavesdropping](#), [impersonation](#).

target library high-level qualifier (thlqual)

A high-level qualifier for z/OS target data set names.

target queue manager

See [remote queue manager](#).

task control block (TCB)

A z/OS control block that is used to communicate information about tasks within an address space that is connected to a subsystem.

task switching

The overlapping of I/O operations and processing between several tasks.

TCB

See [task control block](#).

TCP

See [Transmission Control Protocol](#).

TCP/IP

See [Transmission Control Protocol/Internet Protocol](#).

technote

A short document about a single topic.

telemetry channel

A Telemetry channel is a communication link between a queue manager on WebSphere MQ, and MQTT clients. Each channel might have one or more telemetry devices connected to it.

telemetry advance client

The Advanced telemetry client is installed in the mqxr subfolder of the main WebSphere MQ installation. They are small footprint, MQTT servers allowing multiple MQTT clients to connect to it and provide an uplink or bridge to WebSphere MQ. Advanced clients can start messages on behalf of the clients when the uplink connection is broken.

telemetry client

Telemetry clients are MQTT clients installed within the mqxr subfolder of the main WebSphere MQ installation. The telemetry clients use the MQTT protocol to connect to MQ.

telemetry (MQXR) service

An MQ service that handles the server half of the MQTT protocol (see MQTT Server). The telemetry (MQXR) service hosts telemetry channels.

temporary dynamic queue

A dynamic queue that is deleted when it is closed. Temporary dynamic queues are not recovered if the queue manager fails, so they can contain nonpersistent messages only. See also [permanent dynamic queue](#).

teraspace

A one terabyte temporary storage area that provides storage that is private to a process.

termination notification

A pending event that is activated when a CICS subsystem successfully connects to WebSphere MQ for z/OS.

thlqual

See [target library high-level qualifier](#).

thread

A stream of computer instructions that is in control of a process. In some operating systems, a thread is the smallest unit of operation in a process. Several threads can run concurrently, performing different jobs.

TID

See [transaction identifier](#).

time-independent messaging

See [asynchronous messaging](#).

TLS

Transport Layer Security - successor to SSL.

TMF

See [Transaction Manager Facility](#).

TMI

See [trigger monitor interface](#).

TP

See [transaction program](#).

trace

A record of the processing of a computer program or transaction. The information collected from a trace can be used to assess problems and performance.

transaction ID

See [transaction identifier](#).

transaction identifier (TID, transaction ID, XID)

A unique name that is assigned to a transaction and is used to identify the actions associated with that transaction.

transaction manager

A software unit that coordinates the activities of resource managers by managing global transactions and coordinating the decision to commit them or roll them back.

Transaction Manager Facility (TMF)

In MQSeries for HP NonStop Server, a subsystem to protect your business transactions and the integrity of your databases. Often used synonymously with NonStop Transaction Manager/MP.

transaction program (TP)

A program that processes transactions in an SNA network.

Transmission Control Protocol (TCP)

A communication protocol used in the Internet and in any network that follows the Internet Engineering Task Force (IETF) standards for internetwork protocol. TCP provides a reliable host-to-host protocol in packet-switched communication networks and in interconnected systems of such networks. See also [Internet Protocol](#).

Transmission Control Protocol/Internet Protocol (TCP/IP)

An industry-standard, nonproprietary set of communication protocols that provides reliable end-to-end connections between applications over interconnected networks of different types.

transmission program

See [message channel agent](#).

transmission queue

A local queue on which prepared messages destined for a remote queue manager are temporarily stored.

triggered queue

A local queue which, when it has triggering set on and when the triggering conditions are met, requires that trigger messages are written.

trigger event

An event, such as a message arriving on a queue, that causes a queue manager to create a trigger message on an initiation queue.

triggering

In WebSphere MQ, a facility that allows a queue manager to start an application automatically when predetermined conditions on a queue are satisfied.

trigger message

A message that contains information about the program that a trigger monitor is to start.

trigger monitor

A continuously running application that serves one or more initiation queues. When a trigger message arrives on an initiation queue, the trigger monitor retrieves the message. It uses the information in the trigger message to start a process that serves the queue on which a trigger event occurred.

trigger monitor interface (TMI)

The WebSphere MQ interface to which customer- or vendor-written trigger monitor programs must conform. A part of the WebSphere MQ Framework.

trust store

The place where CA certificates are put to validate certificates from a remote system. See also [key store](#)

two way authentication

In this method of authentication, the queue manager and the client, present the certificate to each other. Also known as mutual authentication.

two-phase commit

A two-step process by which recoverable resources and an external subsystem are committed. During the first step, the database manager subsystems are polled to ensure that they are ready to commit. If all subsystems respond positively, the database manager instructs them to commit.

type

A characteristic that specifies the internal format of data and determines how the data can be used.

U

UDP

See [User Datagram Protocol](#).

unauthorized access

Gaining access to resources within a computer system without permission.

undelivered message queue

See [dead-letter queue](#).

undo/redo record

A log record used in recovery. The redo part of the record describes a change to be made to a WebSphere MQ object. The undo part describes how to back out the change if the work is not committed.

unit of recovery

A recoverable sequence of operations within a single resource manager, such as an instance of Db2 for z/OS. See also [unit of work](#).

unit of work (UOW)

A recoverable sequence of operations performed by an application between two points of consistency. A unit of work begins when a transaction starts or at a user-requested syncpoint. It ends either at a user-requested syncpoint or at the end of a transaction.

UOW

See [unit of work](#).

user bag

In the MQAI, a type of data bag that is created by the user.

User Datagram Protocol (UDP)

An Internet protocol that provides unreliable, connectionless datagram service. It enables an application program on one machine or process to send a datagram to an application program on another machine or process.

user item

In the MQAI, a type of data item that is created by the user.

user selector

In the WebSphere MQ Administration Interface (MQAI), the identifier that is placed with a data item into a data bag to identify the data item. WebSphere MQ provides predefined user selectors for WebSphere MQ objects.

user token (UTOKEN)

The RACF security token that encapsulates or represents the security characteristics of a user. RACF assigns a UTOKEN to each user in the system.

utility

In WebSphere MQ, a supplied set of programs that provide the system operator or system administrator with facilities in addition to those provided by the WebSphere MQ commands.

UTOKEN

See [user token](#).

V

value

The content of a data item. This can be an integer, a string, or the handle of another data bag.

virtual method

In object-oriented programming, a method that exhibits polymorphism.

W

WebSphere MQ

A family of IBM licensed programs that provides message queuing services.

WebSphere MQ Administration Interface (MQAI)

A programming interface that performs administration tasks on a WebSphere MQ queue manager through the use of data bags. Data bags allow the user to handle properties (or parameters) of WebSphere MQ objects.

WebSphere MQ classes for .NET

A set of classes that allow a program written in the .NET programming framework to connect to WebSphere MQ as a WebSphere MQ client or to connect directly to a WebSphere MQ server.

WebSphere MQ classes for C++

A set of classes that encapsulate the WebSphere MQ Message Queue Interface (MQI) in the C++ programming language.

WebSphere MQ classes for Java

A set of classes that encapsulate the WebSphere MQ Message Queue Interface (MQI) in the Java programming language.

WebSphere MQ fully-managed .NET client

Part of a WebSphere MQ product that can be installed on a system without installing the full queue manager. The WebSphere MQ .NET client is used by fully-managed .NET applications and communicates with a queue manager on a server system. A .NET application that is not fully managed uses the WebSphere MQ MQI client. See also [client](#), [WebSphere MQ MQI client](#), [WebSphere MQ Java client](#).

WebSphere MQ Java client

Part of a WebSphere MQ product that can be installed on a system without installing the full queue manager. The WebSphere MQ Java client is used by Java applications (both WebSphere MQ classes for Java and WebSphere MQ classes for JMS) and communicates with a queue manager on a server system. See also [client](#), [WebSphere MQ MQI client](#), [WebSphere MQ fully-managed .NET client](#).

WebSphere MQ MQI client

Part of a WebSphere MQ product that can be installed on a system without installing the full queue manager. The WebSphere MQ MQI client accepts MQI calls from applications and communicates with

a queue manager on a server system. See also [client](#), [WebSphere MQ Java client](#), [WebSphere MQ fully-managed .NET client](#).

WebSphere MQ script commands (MQSC)

Human readable commands, uniform across all platforms, that are used to manipulate WebSphere MQ objects. See also [programmable command format](#).

WebSphere MQ server

A queue manager that provides queuing services to one or more clients. All the WebSphere MQ objects, for example queues, exist only on the queue manager system, that is, on the MQI server machine. A server can support normal local MQI applications as well.

WebSphere MQ Telemetry

WebSphere MQ Telemetry provides small client libraries that can be embedded into smart devices running on a number of different device platforms. Applications built with the clients use MQ Telemetry Transport (MQTT) and the WebSphere MQ Telemetry (MQXR) service to publish and subscribe messages reliably with WebSphere MQ. When the WebSphere MQ custom installation option to install Telemetry is selected, it installs: 1) Telemetry (MQXR) service 2) Telemetry clients and 3) Telemetry Advanced clients.

WebSphere MQ Telemetry daemon for devices

The WebSphere MQ Telemetry daemon for devices is an advanced MQTT V3 client. It is a very small footprint MQTT server designed for embedded systems.

Windows NT Challenge/Response

The authentication protocol that is used on networks that include Windows NT systems and on standalone systems.

wiretapping

The act of gaining access to information that is flowing along a wire or any other type of conductor used in communications. The objective of wiretapping is to gain unauthorized access to information without being detected.

X

X509

International Telecommunications Union standard for PKI. Specifies the format of the public key certificate and the public key cryptography.

XCF

See [cross-system coupling facility](#).

XID

See [transaction identifier](#).

X/Open XA

The X/Open Distributed Transaction Processing XA interface. A proposed standard for distributed transaction communication. The standard specifies a bidirectional interface between resource managers that provide access to shared resources within transactions, and between a transaction service that monitors and resolves transactions.

Accessibility features for IBM WebSphere MQ

Accessibility features help users who have a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

The following list includes the major accessibility features in IBM WebSphere MQ. You can use screen-reader software to hear what is displayed on the screen.

- Supports keyboard-only operation
- Supports interfaces commonly used by screen readers

Tip: This product documentation, and its related publications, are accessibility-enabled for the IBM Home Page Reader. You can operate all features using the keyboard instead of the mouse. For more information, see [Accessibility features](#).

Keyboard navigation

This product uses standard Linux and Microsoft Windows navigation keys.

For more information, see [Accessibility features](#).

Visit the <https://www.ibm.com/able> for more information about the commitments that IBM makes towards accessibility.

Accessibility on Windows

The IBM WebSphere MQ user interfaces do not use any special keys, but instead follow the Windows user interface guidelines for accelerator keys on items such as context menus, dialogs, and dialog controls such as buttons. Access the accelerator keys in the usual way. See the Windows help for more information (look in the Windows help index for *keyboard*; for accessibility features look for *Accessibility*).

Special features for accessibility

Some of the user interfaces in IBM WebSphere MQ are normally visual, but they behave differently when accessibility features are activated, as follows:

- High Contrast Mode

In this mode Launchpad, Prepare IBM WebSphere MQ Wizard, Postcard, and Default Configuration all hide their background bitmaps and ensure that they use the system text colors so that they are easily visible and readable.

- Screen Reader Mode

When a screen reader is active, Prepare IBM WebSphere MQ Wizard, Default Configuration, and Postcard, simplify their appearance by hiding background bitmaps, raised effects, shadow boxes, and other effects that can otherwise confuse the screen reader.

- Explorer Object Status

The Explorer component of IBM WebSphere MQ uses icons to indicate the status of objects, such as queue managers. Screen readers cannot interpret these icons, so there is an option to show a textual description of the icon. To select this option, from within the Explorer click **Window > Preferences > WebSphere MQ Explorer** and select **Show status of objects after object name**.

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:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, 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.

This book contains information on intended programming interfaces that allow the customer to write programs to obtain the services of IBM WebSphere MQ.

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, ibm.com[®], are trademarks of IBM Corporation, registered in many jurisdictions worldwide. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" www.ibm.com/legal/copytrade.shtml. Other product and service names might be trademarks of IBM or other companies.

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

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

This product includes software developed by the Eclipse Project (<http://www.eclipse.org/>).

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.



Part Number:

(1P) P/N: