WebSphere MQ for Linux

Quick Beginnings

Version 6.0

GC34-6480-00

WebSphere MQ for Linux

Quick Beginnings

Version 6.0

GC34-6480-00

Note!:

Before using this information and the product it supports, be sure to read the general information under notices at the back of this book.

First Edition (May 2005)

This edition of the book applies to the following product:

- IBM WebSphere MQ for Linux Version 6.0

and to any subsequent releases and modifications until otherwise indicated in new editions.

Contents

Tables

Welcome to WebSphere MQ for Linux

This book describes WebSphere® MQ for Linux®, Version 6.0, and explains how to plan for the product, install it, and verify that the installation has worked.

See the:

- *WebSphere MQ Bibliography and Glossary* for an explanation of terms used in this book
- *WebSphere MQ System Administration Guide* for further information on using the control commands crtmqm, strmqm and endmqm.

Road map

Use Table 1 to find the information that you need to get started with WebSphere MQ for Linux.

Table 1. Getting started road map

If you want to...	Refer to...
Learn about system requirements for WebSphere MQ for Linux	"Preparing for installation" on page 3
Install WebSphere MQ for Linux server	"Installing a WebSphere MQ server" on page 15
Install the WebSphere MQ for WebSphere MQ for Linux client	"Installing WebSphere MQ" on page 41 and "Verifying the client installation" on page 42
Read more about WebSphere MQ	Chapter 7, "WebSphere MQ Documentation," on page 53
Apply maintenance to WebSphere MQ for Linux	Chapter 7, "WebSphere MQ Documentation," on page 53
Uninstall a WebSphere MQ for Linux server or client	Chapter 6, "Uninstalling WebSphere MQ," on page 51

Conventions

Knowing the conventions used in this book will help you to use it more efficiently.

- Where appropriate, WebSphere MQ for Linux, Version 6.0 (x86 platform), WebSphere MQ for Linux, Version 6.0 (zSeries® platform), WebSphere MQ for Linux, Version 6.0 (POWER™ platform), are together referred to as WebSphere MQ for Linux.
- The terms click, double-click, and right-click are used to describe item selection with the mouse.
- The term enter means type the relevant text or command, then press the Enter key.
- **Boldface type** indicates the name of an item that you need to select or the name of a command.
- *Italics type* indicates new terms, book titles, or variable information that must be replaced by an actual value.
- `Monospace type` indicates an example (such as a fictitious path or file name) or text that is displayed on the screen.

What's new in WebSphere MQ for Linux, Version 6.0

WebSphere MQ for Linux, Version 6.0 provides the following new and changed functions:

- WebSphere MQ for Linux, Version 6.0 (x86 platform) now supports the WebSphere MQ Explorer. The Eclipse-based WebSphere MQ Explorer is a graphical user interface and allows you to administer and monitor WebSphere MQ objects on both Linux and other platforms. For further information about using the WebSphere MQ Explorer see the *System Administration Guide*.
- WebSphere MQ for Linux, Version 6.0 introduces the ability for queue managers to communicate using the IPv6 protocol in addition to the existing, IPv4, protocol. For further information for migrating customers see the *Migration* book.
- A new form of license management is implemented for this release of the product. WebSphere MQ for Linux, Version 6.0 supports IBM Tivoli License Manager.
- Support for DCE exits, the DCE name service and DCE thread support features has been removed. For further information for migrating customers see the *Migration* book.
- Recent security improvements in WebSphere MQ Version 6.0 have affected the behavior of the AMQ6183 message. Users who are not in the mqm group cannot have message AMQ6183 written to the System error log file. Message AMQ6183 indicates that a FDC record has been written due to an FFST being generated. As a result, users can no longer rely on these messages for information regarding processes run by users who are not members of group mqm. See the *Security* book for further information on security changes.
- WebSphere MQ for Linux, Version 6.0 now runs on the POWER platform. The queue manager for this platform is 64-bit. Queue manager processes are 64-bit only. For information on the implications of migrating to use the new 64-bit capabilities see the *Migration* book.
- WebSphere MQ for Linux, Version 6.0 (x86 platform) supports the WebSphere MQ File Transfer Application, new to this product release. The File Transfer Application allows you to send and receive ordinary files in the form of WebSphere MQ messages and has both a graphical user interface and a command line interface.
- The SSL runtime cryptography on WebSphere MQ for Linux, Version 6.0 provides a cryptography package called IBM Crypto for C (ICC). On Linux the ICC software complies with the National Institute of Standards and Technology (NIST) Federal Information Processing Standards (FIPS) Cryptomodule Validation Program (CMVP).

Release notes

Before starting to install WebSphere MQ, review the release notes file, which you will find on the product CD-ROM in the `\Readmes` folder for each national language. This file contains any additional information about the WebSphere MQ for Linux, Version 6.0 product and might update information in this book.

During installation, the release notes file is copied to the WebSphere MQ program files folder (default `c:\Program Files\IBM\WebSphere MQ`).

Chapter 1. Migrating from an earlier version

See the following topics for information about how to migrate to WebSphere MQ Version 6.0.

- "Migrating to WebSphere MQ Version 6.0 and upgrading your operating system"
- "Migrating to WebSphere MQ Version 6.0 when using Red Hat Enterprise Linux V3.0" on page 2

Migrating to WebSphere MQ Version 6.0 and upgrading your operating system

If you want to migrate to WebSphere MQ Version 6.0, and also need to upgrade your operating system to one supported by WebSphere MQ Version 6.0, complete this task. For a list of supported Operating Systems, see "Checking the operating environment" on page 4.

Follow these steps to upgrade your operating system and migrate to WebSphere MQ Version 6.0 whilst keeping your WebSphere MQ data.

1. End all WebSphere MQ activity.
2. Function supplied by the SupportPacs in the table below has been superseded by function in WebSphere MQ Version 6.0. Remove these SupportPacs before installing WebSphere MQ Version 6.0.

Table 2. SupportPacs superseded by WebSphere MQ Version 6.0

SupportPac™ Number	**Description**
MA0C	MQSeries® Publish/subscribe
MA0R	WebSphere MQ transport for SOAP
MA88	MQSeries Classes for Java™ and WebSphere MQ classes for Java Message Service

   Please review any other installed SupportPacs for their applicability to WebSphere MQ Version 6.0.
3. Uninstall WebSphere MQ.
4. Copy `/var/mqm` directory and save it to a location which will not be erased by the operating system upgrade.
5. Upgrade your operating system by following the manufacturer's instructions.
6. Copy the `/var/mqm` directory that you saved in step 4 to back into your file system.
7. Check your system has the required WebSphere MQ Version 6.0 prerequisites and install WebSphere MQ Version 6.0 as described in the following chapters.
8. Restart WebSphere MQ.

Migrating to WebSphere MQ Version 6.0 when using Red Hat Enterprise Linux V3.0

If you want to migrate to WebSphere MQ Version 6.0 and are already using the RedHat Enterprise Linux 3.0 operating system with WebSphere MQ Version 5.3, you do not need to upgrade your operating system. Follow these steps to migrate to WebSphere MQ Version 6.0 whilst keeping your WebSphere MQ data:

1. End all WebSphere MQ activity.
2. Check that the environment variable LD_ASSUME_KERNEL is set to 2.4.19 by typing the following command:

   ```
   echo $LD_ASSUME_KERNEL
   ```

3. If it is not correctly set, type the following command:

   ```
   export LD_ASSUME_KERNEL=2.4.19
   ```

4. Setting LD_ASSUME_KERNEL to 2.4.19 enables Linux Threads. Accessing the RPM database whilst using Linux Threads can cause the database to corrupt due to an error in RPM function on Red Hat Enterprise Linux 3.0. To force RPM to run under the NPTL threading model, set the environment variable RPM_FORCE_NPTL by typing the following command:

   ```
   export RPM_FORCE_NPTL=1
   ```

5. Uninstall WebSphere MQ Version 5.3.
6. Unset the LD_ASSUME_KERNEL and RPM_FORCE_NPTL environment variables by typing the following commands:

   ```
   unset LD_ASSUME_KERNEL
   unset RPM_FORCE_NPTL
   ```

7. Check your system has the required WebSphere MQ prerequisites and install WebSphere MQ Version 6.0 as described in the following chapters.
8. Restart WebSphere MQ.

Chapter 2. Installing a WebSphere MQ server

This chapter describes how to install a WebSphere MQ Version 6.0 server. The information covers topics such as preparing for installation and verifying your installation, as well as installation itself. If you already have an installation of WebSphere MQ, and are migrating to WebSphere MQ Version 6.0 see "Migrating to WebSphere MQ Version 6.0 and upgrading your operating system" on page 1 before installing WebSphere MQ Version 6.0.

WebSphere MQ for Linux can be installed as a server or a client.

A WebSphere MQ server is an installation of one or more queue managers that provide queueing services to one or more clients. All the WebSphere MQ objects, for example queues, exist only on the queue manager machine (the WebSphere MQ server machine), and not the client. A WebSphere MQ server can also support local WebSphere MQ applications.

A WebSphere MQ client is a component that allows an application running on one system to communicate with a queue manager running on another system. The output from the call is sent back to the client, which passes it back to the application. To install a WebSphere MQ client see, Chapter 3, "Installing a WebSphere MQ client," on page 31.

It is possible to have both a server and a client installation on the same machine, for instructions on how to do this see, Chapter 4, "Installing a client on the same machine as a server," on page 47.

See the *WebSphere MQ System Administration Guide* for an introduction to WebSphere MQ concepts and objects.

For information on the components that can be included in the server and client installations see, "WMQ Components" on page 16.

The installation process is divided into the following set of tasks, complete all of these tasks in sequence:

- "Preparing for installation"
- "Installing a WebSphere MQ server" on page 15
- "Verifying your installation" on page 19

Preparing for installation

Before you install WebSphere MQ, complete the following tasks.

- "Checking prerequisite hardware and software" on page 4
- "Creating WebSphere MQ file systems" on page 11
- "Setting up the user ID and group ID" on page 13

Additionally, if you require messages in a language other than U.S. English see, "Displaying messages in your national language" on page 13.

Checking prerequisite hardware and software

This section details the operating system requirements, the prerequisite software and optional software required for using WebSphere MQ Version 6.0. These requirements differ in some cases depending on the system hardware on which you will be running WebSphere MQ. Please see the appropriate section for your installation.

Checking the operating environment

Before installing WebSphere MQ Version 6.0, you must check that your system meets the hardware and software requirements set out in this topic.

Note: WebSphere MQ does not support host names that contain spaces. If you install WebSphere MQ on a computer with a host name that contains spaces, you will be unable to create any queue managers.

Hardware

WebSphere MQ for Linux, Version 6.0 (x86 platform) runs on any machine that supports the x86 machine architecture. WebSphere MQ for Linux, Version 6.0 (POWER platform) runs on any machine that supports the POWER machine architecture. WebSphere MQ for Linux, Version 6.0 (zSeries platform) runs on any machine that supports the zSeries machine architecture.

Operating System, WebSphere MQ for Linux, Version 6.0 (x86 platform)

WebSphere MQ for Linux, Version 6.0 (x86 platform) has been tested with the following distributions:

- Red Hat Enterprise Linux AS V3.0 plus Update 2
- SuSE Linux Enterprise Server (SLES) V8 plus Service Pack 3

  Note: The MQ Explorer graphical interface is not supported on this operating system.
- SuSE Linux Enterprise Server (SLES) V9

Operating System, WebSphere MQ for Linux, Version 6.0 (POWER platform)

WebSphere MQ for Linux, Version 6.0 (POWER platform)) has been tested with the following distributions:

- Red Hat Enterprise Linux AS V3.0 plus Update 2
- SuSE Linux Enterprise Server (SLES) V9

Operating System, WebSphere MQ for Linux, Version 6.0 (zSeries platform)

WebSphere MQ for Linux, Version 6.0 (zSeries platform) has been tested with the following distributions:

- Red Hat Enterprise Linux AS V3.0 plus Update 2
- SuSE Linux Enterprise Server (SLES) V8 plus Service Pack 3

- SuSE Linux Enterprise Server (SLES) V9

Threading Models

The support for threads on Linux is provided by the pthreads library.

- Most distributions based upon the 2.4 series kernels, such as those based upon UnitedLinux 1.0, include the LinuxThreads pthreads library. This threading library differs from most others in that the 'getpid()' function returns a different value for each thread in a process and is identified as a separate process in the output of the 'ps' command.
- Most distributions based upon the 2.6 series kernels, such as those based upon UnitedLinux 2.0, include the Native Posix Threading Library (NPTL). NPTL was written as the replacement to LinuxThreads and relies on updates made to recent kernels to overcome some of the shortfalls of LinuxThreads. Notably, threads within a process return the same pid from calls to 'getpid()' and closer compliance to the POSIX Threads specification.

Both of these threading models are supported for use with the WebSphere MQ Queue Manager and applications which connect to WebSphere MQ.

Other threading libraries exist as replacements for these threading libraries, however, none of these replacements are supported for use with the WebSphere MQ Queue Manager or applications which connect to WebSphere MQ.

Connectivity Requirements

The network protocols supported by WebSphere MQ for Linux, Version 6.0 (x86 platform) and WebSphere MQ for Linux, Version 6.0 (zSeries platform) are TCP/IP and LU6.2 . TCP/IP is part of the WebSphere MQ for Linux, Version 6.0 (x86 platform) and WebSphere MQ for Linux, Version 6.0 (zSeries platform) operating systems. If you want to use the SNA LU6.2 support on WebSphere MQ for Linux, Version 6.0 (x86 platform) or WebSphere MQ for Linux, Version 6.0 (zSeries platform) you need the IBM® Communications Server for Linux Version 6.2. The Communications Server is available as a PRPQ product from IBM. For more details, see: http://www.ibm.com/software/network/commserver/about.

The network protocol supported by WebSphere MQ for Linux, Version 6.0 (POWER platform) is TCP/IP. You can use any communications hardware supporting TCP/IP.

Storage Requirements

The storage requirements for the WebSphere MQ for Linux depend on which components you install, and how much working space you need. This, in turn, depends on the number of queues that you use, the number and size of the messages on the queues, and whether the messages are persistent. You also require archiving capacity on disk, tape or other media. The approximate amount of storage required for each component is detailed in the table below.

Table 3. Storage requirements for WebSphere MQ filesets

Component	Location of storage	Storage Requirement in MB (x86 platform)	Storage Requirement in MB (z/Series platform)	Storage Requirement in MB (POWER platform)
Runtime	/usr	25.2	25.1	30.8

Table 3. Storage requirements for WebSphere MQ filesets (continued)

Client	/opt	2.8	2.9	5.0
Server	/opt	14.3	14.5	32.8
SDK	/opt	1.7	1.7	1.7
Sample programs	/opt	4.8	4.8	5.3
Java messaging	/opt	6.9	6.9	6.9
SSL support	/opt	58.6	50.0	60.8
Extended Transactional Client	/opt	0.1	0.1	0.2
Man pages	/usr	0.7	0.8	0.8
German Message catalogs	/usr	0.5	0.6	0.7
Spanish Message catalogs	/usr	0.6	0.6	0.6
French Message catalogs	/usr	0.6	0.6	0.7
Italian Message catalogs	/usr	0.6	0.6	0.6
Japanese Message catalogs	/usr	0.5	0.5	0.6
Korean Message catalogs	/usr	0.5	0.5	0.5
Brazilian Portuguese Message catalogs	/usr	0.6	0.6	0.6
Simplified Chinese Message catalogs	/usr	0.4	0.4	0.4
Traditional Chinese Message catalogs	/usr	0.4	0.4	0.4
WebSphere Eclipse Platform (x86 platform only)	/opt	224.8	not applicable	not applicable
WebSphere MQ Explorer (x86 platform only	/opt	40.5	not applicable	not applicable
WebSphere MQ File Transfer Application (x86 platform only)	/opt	0.8	not applicable	not applicable

Table 4. Storage requirements for prerequisite filesets

Component	**Location of storage**	**Storage Requirement in MB (x86 platform)**	**Storage Requirement in MB (z/Series platform)**	**Storage Requirement in MB (POWER platform)**
gsk7bas-7.0-x.x	**/usr**	**10.4**	**13.3**	**11.3**
gsk7bas64-7.0-x.x	**/usr**	**not applicable**	**not applicable**	**12.5**
IBMJava2-SDK-1.4.2-0.0	**/opt**	**80.5**	**71.3**	**82.5**
IBMJava2-SDK-ppc64-1.4.2-0.0	/opt	not applicable	not applicable	82.1

You can use the df command to determine the amount of free space on your system.

Disk storage is also required for

- Prerequisite software
- Optional software
- Your application programs

If you want to use the WebSphere MQ Explorer (available for use with WebSphere MQ for Linux, Version 6.0 (x86 platform) only), you will need a minimum

- 512 MB RAM
- 1 GHz processor
- 1.5 GB for WebSphere Eclipse Platform code and data
- A suitable monitor for the operating system with a screen size of at least 1024x768

After checking the requirements in this topic, proceed to "Prerequisite software for using SSL."

Prerequisite software for using SSL

This section describes the prerequisites for using SSL, minimum supported software levels are given. Later levels, if any, are supported unless otherwise stated. These prerequisites apply to both client and server installations of WebSphere MQ.

Installing the g++ version runtime support

If you intend to run SSL channels then you must have the g++ version 3.2 runtime libraries installed. The GNU g++ libraries are called libgcc_s.so and `libstdc++.so.5` and, on a Red Hat sytem with `libgcc` and `libstdc++5` RPMs installed, are found in the directory `/usr/lib`.

If you have a distribution that does not install `libgcc_s.so` and `libstdc++.so.5` into `/usr/lib`, then you will have to do one of the following:

- Locate and install the packages from your distribution vendor that contain these libraries.
- Install the GNU gcc and g++ version 5 compilers from another location, for example the home page for the GNU compiler collection at URL:.

  `http://gcc.gnu.org/`

  Ensure that the libraries listed above are included in `/usr/lib` after installation.

IBM Global Security Kit V7

If you want to use the SSL support, you need IBM Global Security Kit V7. This is supplied with WebSphere MQ as one of the components available for installation. If you are migrating from WebSphere MQ Version 5.3 and have no other requirement for the IBM Global Security Kit V6, you can uninstall it using the process described in "Installing a WebSphere MQ server" on page 15, the package name is `gskbas-6.0-n.nn.<arch>.rpm`.

Checking optional software - x86 platform

Compilers

The following compilers are supported for WebSphere MQ for Linux, Version 6.0.

C applications/C++ applications

- GNU C Compiler (gcc) and g++ Version 2.9.5
- GNU C Compiler (gcc) and g++ Version 3.2
- GNU C Compiler (gcc) and g++ Version 3.3
- GNU C Compiler (gcc) and g++ Version 3.4

Note: The C++ support libraries are installed in directories whose names match the compiler version, `/opt/mqm/lib/<version>`, and links are placed from `/opt/mqm/lib` to the default version, 3.2.

COBOL applications

- Micro Focus Server Express, V4.0

Java applications

- IBM 32-bit SDK for Linux on Intel® architecture, Java 2 Technology Edition, Version 1.4.2

Transaction Monitors

- BEA Tuxedo 8.1
- WebLogic V8.1 is supported when used with the Red Hat Enterprise Linux 3.0 operating system.
- IBM WebSphere Application Server (WAS) V5.1 is supported when used with the SUSE Linux Enterprise Server (SLES) 8 operating system.

WebSphere MQ for Linux, Version 6.0 supports WebSphere Application Server as an XA coordinator. For more information about the WebSphere MQ application adaptor, see the WebSphere Application Server Enterprise Edition WebSphere MQ Application Adaptor Development Guide, SC09-4444.

Databases

The following databases are supported:

- DB2 Universal Database™ V8.2
- Oracle9*i*Release 2 with Patch Set 4 (9.2.0.5)
- Oracle 10g
- Sybase Adaptive Server Enterprise (ASE) 12.5.1
- Informix® Dynamic Server (IDS) V9.40 plus Client SDK V2.90

Java Messaging and SOAP transport

If you want to use Java Messaging and SOAP (Simple Object Access Protocol) Support, you need a Java Runtime Environment Version 1.4.2 or later. An appropriate runtime environment is provided in the JDK provided on the WebSphere MQ Server CD-ROM. See “WMQ Components” on page 16 for package information.

The JDK available on the server CD-ROM is:

- IBM 32-bit SDK for Linux on Intel architecture, Java 2 Technology Edition, Version 1.4.2

This JDK provides support for SOAP and are FIPS 140-2 compliant. For a list of alternative JDKs, see

www.ibm.com/software/integration/websphere/mqplatforms/supported.html

For further information about using Java with WebSphere MQ see, *Using Java*.

For further information about SOAP see, the *WebSphere MQ Transport For SOAP* book.

If you use a JDK other than the one supplied by WebSphere MQ Version 6.0 you should be aware that:

- The JDK may not be FIPS level 140-2 compliant and by using it with WebSphere MQ, will not comply the FIPS 140-2 standards.
- SOAP is not supported.

Checking optional software - POWER platform

Compilers

The following compilers are supported for WebSphere MQ for Linux, Version 6.0.

C applications/C++ applications

- GNU C Compiler (gcc) and g++ Version 3.2
- GNU C Compiler (gcc) and g++ Version 3.3
- GNU C Compiler (gcc) and g++ Version 3.4

Note: The C++ support libraries are installed in directories whose names match the compiler version, `/opt/mqm/lib/<version>`, and links are placed from `/opt/mqm/lib` to the default version, 3.2.

COBOL applications

- Micro Focus Server Express, V4.0

Java applications

- IBM Software Developer's Kit (SDK), Java 2 Technology Edition for Linux on POWER Version 1.4.2

Databases

The following database are supported:

- DB2 Universal Database V8.2

Java Messaging and SOAP transport

If you want to use Java Messaging and SOAP (Simple Object Access Protocol) Support, you need a Java Runtime Environment Version 1.4.2 or later. An appropriate runtime environment is provided in the JDKs provided on the WebSphere MQ Server CD- ROM. See "WMQ Components" on page 16 for package information.

The JDKs available on the server CD-ROM are:

- IBM 32-bit Software Developer's Kit (SDK) for Linux for iSeries™ and pSeries®, Java 2 Technology Edition, Version 1.4
- IBM 64-bit Software Developer's Kit (SDK) for Linux for iSeries and pSeries, Java 2 Technology Edition, Version 1.4

These JDKs provide support for SOAP and are FIPS 140-2 compliant. For a list of alternative JDKs, see

www.ibm.com/software/integration/websphere/mqplatforms/supported.html

For further information about using Java with WebSphere MQ see, *Using Java*.

For further information about SOAP see, *WebSphere MQ Transport For SOAP* book.

If you use a JDK other than the one supplied by WebSphere MQ Version 6.0 you should be aware that:

- The JDK may not be FIPS level 140-2 compliant and by using it with WebSphere MQ, will not comply the FIPS 140-2 standards.
- SOAP is not supported.

On the POWER platform, the 32-bit and 64-bit JDKs are typically installed to different locations, for example, the 32-bit JDK is located in `/opt/IBMJava2-ppc-142` and the 64-bit JDK is located in `/opt/IBMJava2-ppc64-142`. Ensure that the PATH variable is correctly set for your applications that use Java. To use the JMS Postcard application described in "Verifying the installation using the JMS Postcard application" on page 24 you need to use a 32-bit JDK. You can check the version installed using the following command:

```
java -version
```

Checking optional software - z/Series platform

Compilers

The following compilers are supported for WebSphere MQ for Linux, Version 6.0.

C applications/C++ applications

- GNU C Compiler (gcc) and g++ Version 2.9.5
- GNU C Compiler (gcc) and g++ Version 3.2
- GNU C Compiler (gcc) and g++ Version 3.3
- GNU C Compiler (gcc) and g++ Version 3.4

Note: The C++ support libraries are installed in directories whose names match the compiler version, `/opt/mqm/lib/<version>`, and links are placed from `/opt/mqm/lib` to the default version, 3.2.

COBOL applications

- Micro Focus Server Express, V4.0

Java applications

- IBM 31-bit SDK for Linux on zSeries, Java 2 Technology Edition, Version 1.4

Transaction Monitors

- BEA Tuxedo 8.1 is supported when used with the SUSE Linux Enterprise Server (SLES) 8 operating system.

- IBM WebSphere Application Server (WAS) V5.1 is supported when used with the SUSE Linux Enterprise Server (SLES) 8 operating system.

WebSphere MQ for Linux, Version 6.0 supports WebSphere Application Server as an XA coordinator. For more information about the WebSphere MQ application adaptor, see the WebSphere Application Server Enterprise Edition WebSphere MQ Application Adaptor Development Guide, SC09-4444.

Databases

The following databases are supported:

- DB2 Universal Database V8.2
- Informix Dynamic Server (IDS) V9.40 plus Client SDK V2.90

Java Messaging and SOAP transport

If you want to use Java Messaging and SOAP (Simple Object Access Protocol) Support, you need a Java Runtime Environment Version 1.4.2 or later. An appropriate runtime environment is provided in the JDK provided on the WebSphere MQ Server CD-ROM. See "WMQ Components" on page 16 for package information.

The JDK available on the server CD-ROM is:

- IBM 31-bit SDK for Linux on zSeries, Java 2 Technology Edition, Version 1.4

This JDK provides support for SOAP and are FIPS 140-2 compliant. For a list of alternative JDKs, see

www.ibm.com/software/integration/websphere/mqplatforms/supported.html

For further information about using Java with WebSphere MQ see, *Using Java*.

For further information about SOAP see, *WebSphere MQ Transport for SOAP* book.

If you use a JDK other than the one supplied by WebSphere MQ Version 6.0 you should be aware that:

- The JDK may not be FIPS level 140-2 compliant and by using it with WebSphere MQ, will not comply the FIPS 140-2 standards.
- SOAP is not supported.

Creating WebSphere MQ file systems

The installation directory for the WebSphere MQ product code is `/opt/mqm`. Working data is stored in `/var/mqm`. You cannot change these locations. The GSKit must also be installed into its default location.

The topics contained in this section describe how to prepare your file system for installing WebSphere MQ. Perform these tasks before installation.

Creating a file system for the product code

This topic describes how to prepare the `/opt/mqm` file system into which the WebSphere MQ code will be installed and what to do should you not have enough storage space available in the file system.

The WebSphere MQ product code is installed in /opt/mqm. If you cannot install the product code in the /opt/mqm file system because the file system is too small to contain the product, you can do one of the following:

1. Create a new file system and mount it as /opt/mqm. If you choose this option, the new file system must be created and mounted before installing the product code.
2. Create a new directory anywhere on your machine, and create a symbolic link from /opt/mqm to this new directory. For example:

   ```
   mkdir /bigdisk/mqm
   ln -s /bigdisk/mqm /opt/mqm
   ```

   If you choose this option, the new directory must be created, and the link created, before installing the product code.

The file system into which the code is installed can be a remote network device, for example, NFS. However, you must define the mount options defined on that device to allow **setuid** programs, including those which are **setuid** root, to run.

Creating a file system for the working data

Before you install WebSphere MQ for Linux, create and mount a file system called /var/mqm. If possible, use a partition strategy with a separate volume for the WebSphere MQ data. This means that other system activity is not affected if a large amount of WebSphere MQ work builds up.

To determine the size of the /var/mqm file system for a server installation, consider:

- The maximum number of messages in the system at one time
- Contingency for message buildups, if there is a system problem
- The average size of the message data, plus 500 bytes for the message header
- The number of queues
- The size of log files and error messages
- The amount of SSL trace that is written to the /var/mqm/trace directory

Allow 50 MB as a minimum for a WebSphere MQ server and 15 MB as a minimum for a WebSphere MQ client.

Creating separate file systems for working data

You can also create separate file systems for your log data (/var/mqm/log) and error files (/var/mqm/errors). If possible, store log files on a different physical disk from the WebSphere MQ queues (/var/mqm).

If you create separate file systems:

- The /var/mqm and /var/mqm/log directories ***must*** be on a local file system.
- The /var/mqm/errors directory can be NFS mounted. However, if you choose to NFS-mount /var/mqm/errors, the error logs might be lost if the network fails.

If you are creating separate file systems, allow a minimum of 30 MB of storage for /var/mqm for a server installation and 15 MB of storage for /var/mqm for a client installation. Also allow 20 MB of storage for /var/mqm/log, and 4 MB of storage for /var/mqm/errors for both client and sever installations.

If you want to use individual queues that will hold more than 2 GB of data, you must enable /var/mqm to use large files.

The size of the log file depends on the log settings that you use. The minimum sizes above are for circular logging using the default settings. For further information on log sizes see the *WebSphere MQ System Administration Guide*.

Setting up the user ID and group ID

WebSphere MQ requires a user ID of the name mqm, with a primary group of mqm. The mqm user ID owns the directories and files that contain the resources associated with the product. Create the user ID and group IDs as described in the following sections.

- "Creating the user ID and group"
- "Adding existing user IDs to the group"

Creating the user ID and group

Create the required user ID and group ID ***before*** you install WebSphere MQ. Both user ID and group ID must be set to mqm. For stand-alone machines, you can create the new user ID and group IDs locally; for machines administered in a network information services (NIS) domain, an administrator must create the IDs on the NIS master server machine.

It is also suggested that you set the mqm user's home directory to /var/mqm.

Adding existing user IDs to the group

If you want to run administration commands, for example **crtmqm** (create queue manager) or **strmqm** (start queue manager), your user ID must be a member of the mqm group.

Users do not need mqm group authority to run applications that use the queue manager; it is needed only for the administration commands.

Displaying messages in your national language

Messages in U.S. English are automatically installed with WebSphere MQ.

If you require messages in a different language, ensure that you:

1. Install the appropriate message catalog (see "WMQ Components" on page 16).
2. To select messages in a different language, use the following command with the identifier for the language you want to install:

   ```
   export LANG=message identifier
   ```

   The message identifiers for the message catalogs are as follows:
   - de_DE (German)
   - es_ES (Spanish)
   - fr_FR (French)
   - it_IT (Italian)
   - ja_JP (Japanese)
   - ko_KR (Korean)
   - pt_BR (Brazilian Portuguese)

- zh_CN (Simplified Chinese)
- zh_TW (Traditional Chinese)

Implications of a 64-bit queue manager

Note: The 64-bit queue manager is only available on the POWER platform.

When using the new 64-bit queue manager, the use of the LIBPATH and LD_LIBRARY_PATH environment variable is not advised. Setting these environment variables might result in you not being able to run any WebSphere MQ commands. By default, the installation will operate as in previous versions of WebSphere MQ, with symbolic links being created from `/usr/lib`, `/usr/bin` and `/usr/include` to the appropriate files within the WebSphere MQ tree structure. In the case of `/usr/lib` the symbolic links will be to the 32-bit WebSphere MQ libraries provided for customer 32-bit applications.

Note: No symbolic links are required for the 64-bit WebSphere MQ libraries required by WebSphere MQ commands.

All WebSphere MQ commands are 64-bit and have a built in path to the WebSphere MQ 64-bit libraries, however, this can be overridden by the use of LIBPATH and thus can cause WebSphere MQ commands to fail to run. The recommended way of using WebSphere MQ commands and your applications is as follows:

- Unset LIBPATH and LD_LIBRARY_PATH and build your applications with a built in path to the appropriate WebSphere MQ libraries, this is detailed in the appropriate WebSphere MQ book for your type of WebSphere MQ application.
- If you need to set LIBPATH or LD_LIBRARY_PATH, consider not including `/usr/lib` in the path you specify in the variable. If you need to include `/usr/lib` in your LIBPATH or LD_LIBRARY_PATH then in order to avoid errors running 64-bit WebSphere MQ applications or WebSphere MQ commands, consider removing the symbolic links from `/usr/lib` to the 32-bit WebSphere MQ libraries using the **dltmqlnk** command documented in the *System Administration Guide*. The symbolic links can be restored with the **crtmqlnk** command. You also need to build your applications with a built in path to the appropriate WebSphere MQ libraries.
- If you cannot use either of the first two options, run your applications in a different environment to the one which issues any WebSphere MQ commands.

  Note: WebSphere MQ libraries are in the following locations: `/opt/mqm/lib` (32-bit libraries) and `/opt/mqm/lib64` (64-bit libraries).

Kernel configuration

WebSphere MQ makes use of System V IPC resources, in particular shared memory and semaphores. The default configuration of these resources, supplied with your installation, is probably adequate for WebSphere MQ but if you have a large number of queues or connected applications, you might need to increase this configuration.

If you are using linear logging, increase `msgmnb` to 65535 and `msgtql` to the maximum number of log files that may be used on the system, this is the number of primary and secondary logs added together for all queue managers.

The amount of System V IPC resources available may be determined by looking at the contents of the following files:

```
/proc/sys/kernel/shmmax - The maximum size of a shared memory segment.
/proc/sys/kernel/shmmni - The maximum number of shared memory segments.
/proc/sys/kernel/shmall - The maximum amount of shared memory
                          that may be allocated.
/proc/sys/kernel/sem    - The maximum number and size of semaphore sets
                          that may be allocated.
/proc/sys/kernel/msgmnb - The maximum number of bytes on a message queue.
```

For example, to view the maximum size of a shared memory segment that can be created enter:

```
cat /proc/sys/kernel/shmmax
```

To change the maximum size of a shared memory segment to 256 MB enter:

```
echo 268435456 > /proc/sys/kernel/shmmax
```

To configure these values every time the machine is restarted we recommend that you add these commands to a startup script in /etc/rc.d/...

Maximum open files

If the system is heavily loaded, you might need to increase the maximum possible number of open files. If your distribution supports the proc filesystem you can do this by issuing the following command:

```
echo 32768 > /proc/sys/fs/file-max
```

If you are using a pluggable security module such as PAM (Pluggable Authentication Module), ensure that this does not unduly restrict the number of open files for the 'mqm' user. For a standard WebSphere MQ queue manager, set the 'nofile' value to 10240 or more for the 'mqm' user. We suggest you add this command to a startup script in `/etc/rc.d/...`

Maximum processes

A running WebSphere MQ queue manager consists of a number of thread programs, and each connected application will increase the number of threads running in the queue manager processes. By default, the Linux 2.4 kernel is capable of running a large number of processes; however, you should ensure that the maximum number of processes which the `mqm` user is allowed to run is not unduly restricted by one of the pluggable security modules such as PAM. Set `nproc` for the `mqm` user to 4090 or more.

Installing a WebSphere MQ server

Before you start the installation procedure, make sure you have prepared your system as described in "Preparing for installation" on page 3.

This section describes the installation of the server, using the RPM Package Manager installer, enabling you to choose which components you want to install. The components and filesets are listed in "WMQ Components" on page 16; you must install at least the Runtime and Server components.

1. Log in as root.
2. If you are installing from the Server CD-ROM, mount the CD-ROM file system.

3. Run the `mqicense.sh` script. If you want to view a text-only version of license, which can be read by a screen-reader, type:

   ```
   ./mqlicense.sh -text_only
   ```

   The license is displayed.

   If want to accept the license without it being displayed, you can run the `mqlicense.sh` script with the `-accept` option.

   ```
   ./mqlicense.sh -accept
   ```

   You must accept the license agreement before you can proceed with the installation.
4. Use the **rpm -ivh** command to install each component that you require. This example shows a minimum installation,

   ```
   rpm -ivh MQSeriesRuntime-6.0.0-0.i386.rpm MQSeriesServer-6.0.0-0.i386.rpm
   ```

Note: Ensure that `/bin/sh` is a valid shell interpreter compatible with the Bourne shell, otherwise the post-installation configuration of will not complete successfully.

If your Linux machine was not installed with RPM, you might see a prerequisites failure of `/bin/sh` when you try to install . This is because the RPM tables do not recognize that a valid shell interpreter is installed.

If this occurs, you can reinstall the `/bin/sh` shell using RPM, or specify the RPM option `--nodeps` to disable dependency checking during installation of WebSphere MQ.

WMQ Components

When you install WebSphere MQ for Linux, you can choose which components to install.

Table 5. WebSphere MQ components and packages

Component	Description	Package	Server	Client (with SSL)
Runtime	Mandatory component. Needed for application development and provides support for external applications.	MQSeriesRuntime	X	X
SDK	Required for compiling applications.	MQSeriesSDK	X	X
Server	The server feature allows you to run queue managers on your computer and connect to other computers over a network. Provides messaging and queuing services to applications, and support for WebSphere MQ client connections.	MQSeriesServer	X	

Table 5. WebSphere MQ components and packages (continued)

Client	The WebSphere MQ client is a small subset of WebSphere MQ, without a queue manager. Provides remote access to WebSphere MQ. Must be connected to a server. To install a client on the same machine as a server, use the Server CD-ROM; otherwise use the Clients CD-ROM.	MQSeriesClient	X	X
Sample programs	Sample application programs. Needed if you want to check your WebSphere MQ installation using the verification procedures described in "Verifying the installation using the JMS Postcard application" on page 24.	MQSeriesSamples	X	X
Java messaging	The files needed for messaging using Java (includes Java Messaging Service).	MQSeriesJava	X	X
SSL support	Support for SSL key management	MQSeriesKeyman	X	X
Brazilian Portuguese Message catalogs	Brazilian Portuguese message catalogs	MQSeriesMsg_pt	X	X
French Message catalogs	French message catalogs	MQSeriesMsg_fr	X	X
German Message catalogs	German message catalogs	MQSeriesMsg_de	X	X
Italian Message catalogs	Italian message catalogs	MQSeriesMsg_it	X	X
Japanese Message catalogs	Japanese message catalogs	MQSeriesMsg_ja	X	X
Korean Message catalogs	Korean message catalogs	MQSeriesMsg_ko	X	X
Spanish Message catalogs	Spanish message catalogs	MQSeriesMsg_es	X	X
Simplified Chinese Message catalogs	Simplified Chinese message catalogs	MQSeriesMsg_Zh_CN	X	X
Traditional Chinese Message catalogs	Traditional Chinese message catalogs	MQSeriesMsg_Zh_TW	X	X

Table 5. WebSphere MQ components and packages (continued)

Man pages	UNIX man pages, in U.S. English, for the following: • Control commands • Message Queue Interface (MQI) commands • MQSC commands	MQSeriesMan	X	
Extended Transactional Client	WebSphere MQ component that allows a client application, within the same unit of work: • To put messages to, and get messages from, queues that are owned by the queue manager to which it is connected. • To update the resources of a resource manager other than a WebSphere MQ queue manager.	MQSeriesTXClient	X	
WebSphere MQ Explorer (x86 platform only)	The WebSphere MQ Explorer allows you to administer and monitor WebSphere MQ resources.	MQSeriesConfig	X	
WebSphere Eclipse Platform (x86 platform only)	The WebSphere Eclipse Platform is a prerequisite for the WebSphere MQ Explorer and WebSphere MQ File Transfer Application components.	MQSeriesIES30	X	
WebSphere MQ File Transfer Application (x86 platform only)	The File Transfer Application allows you to send and receive ordinary files in the form of WebSphere MQ messages. You can use the File Transfer Application to send and receive any type of file in any format, for example: ASCII Linux format (with line feed characters), ASCII file Windows format (with carriage return/line feed characters), binary (for example, image files, wordprocessor files, spreadsheet files, or zip files), also reports, letters, memos and charts. The File Transfer Application has both a graphical user interface and a command line interface.	MQSeriesFTA	X	X

Table 6. Other products supplied with WebSphere MQ

Component	Description	Fileset	Server	Client
IBM Global Security Kit V7	Certificate and SSL Base Runtime - 32 bit	gsk7bas	X	X
IBM Global Security Kit V7 (POWER platform only)	Certificate and SSL Base Runtime - 64 bit. Currently available only on WebSphere MQ for Linux, Version 6.0 (POWER platform).	gsk7bas64	X	X

Table 6. Other products supplied with WebSphere MQ (continued)

IBM Java SDK (32-bit)	IBM 32-bit SDK for Linux on Intel architecture, Java 2 Technology Edition, Version 1.4.2 (for WebSphere MQ for Linux, Version 6.0 (x86 platform)).IBM 32-bit Software Developer's Kit (SDK) for Linux for iSeries and pSeries, Java 2 Technology Edition, Version 1.4 (for WebSphere MQ for Linux, Version 6.0 (POWER platform)).IBM 31-bit SDK for Linux on zSeries, Java 2 Technology Edition, Version 1.4 (for WebSphere MQ for Linux, Version 6.0 (zSeries platform)).	IBMJava2-SDK	X	X
IBM Java SDK (64-bit) (POWER platform only)	IBM 64-bit Software Developer's Kit (SDK) for Linux for iSeries and pSeries, Java 2 Technology Edition, Version 1.4	IBMJava2-SDK-ppc64	X	X

Verifying your installation

The following set of tasks describes how to verify that the WebSphere MQ for Linux server has been correctly installed and configured. You can verify a WebSphere MQ server installation at different levels:

- To verify a local (stand-alone) installation that has no communication links with other WebSphere MQ installations, see "Verifying a local installation."
- To verify a server-to-server installation that includes communication links to other WebSphere MQ installations, see "Verifying a server-to-server installation" on page 21.

See "Verifying the client installation" on page 42 if you have a client/server installation that includes communication links between a server machine and a WebSphere MQ client.

Verifying a local installation

To verify a local installation using a simple configuration of one queue manager and one queue, complete the following tasks.

- "Setting up the installation"
- "Testing the installation" on page 20

Note: WebSphere MQ object definitions are case-sensitive. Any text entered as an MQSC command in lowercase is converted automatically to uppercase unless you enclose it in single quotation marks. Make sure that you type the examples exactly as shown.

The procedures outlined in this section describe how to configure your default queue manager from the command line.

Setting up the installation

To verify your installation you must first perform this task. From a shell window, use these steps to create a queue manager and a queue:

1. Log in as a user in the `mqm` group

2. Create a default queue manager called `venus.queue.manager` by entering the following command:

   ```
   crtmqm -q venus.queue.manager
   ```

   You will see messages telling you that the queue manager has been created, and that the default WebSphere MQ objects have been created.
3. To start the queue manager, type:

   ```
   strmqm
   ```

   A message tells you when the queue manager has started.
4. Enable MQSC commands by typing:

   ```
   runmqsc
   ```

   A message tells you that an MQSC session has started. MQSC has no command prompt.
5. Define a local queue called `ORANGE.QUEUE` by entering the following command:

   ```
   define qlocal (orange.queue)
   ```

   A message tells you when the queue has been created.
6. Stop MQSC by typing:

   ```
   end
   ```

   You will see some messages, followed by the command prompt.

You have now defined:

- A default queue manager called `venus.queue.manager`
- A queue called `ORANGE.QUEUE`

Now proceed to "Testing the installation" to verify your installation.

Testing the installation

Before completing this task you must have created a queue manager called `venus.queue.manager` and a local queue called `ORANGE.QUEUE`. For instructions on how to do this see Setting up the installation.

To test the queue manager and queue, use the **amqsput** sample program to put a message on the queue, and the **amqsget** sample program to get the message back from the queue:

1. Log on as a user in group `mqm`, if you are not already.
2. Change into the `/usr/mqm/samp/bin` directory, which contains the sample programs.
3. Put a message on the queue using the following command:

   ```
   ./amqsput ORANGE.QUEUE
   ```

   The following messages are displayed:

   ```
   Sample AMQSPUT0 start
   target queue is ORANGE.QUEUE
   ```
4. Type some message text, on one or more lines, followed by a blank line. The following message is displayed:

   ```
   Sample AMQSPUT0 end
   ```

Your message is now on the queue and the command prompt is displayed again.

5. To get the message from the queue, use the following command:

   ```
   ./amqsget ORANGE.QUEUE
   ```

   The sample program starts, and your message is displayed. After a pause, the sample ends and the command prompt is displayed again.

You have now successfully verified your local installation.

Verifying a server-to-server installation

To verify a server-to-server installation using two servers, one as a sender and one as a receiver, complete the following tasks.

- "Setting up the sender server"
- "Setting up the receiver server" on page 22
- "Testing communication between the servers" on page 23

To verify a server-to-server installation you need to check the communications link between the two machines. Before you can do this, you must ensure that the communications protocol has been installed and configured on both systems. WebSphere MQ for Linux, Version 6.0 (x86 platform) and WebSphere MQ for Linux, Version 6.0 (zSeries platform) support both TCP and LU6.2. The network protocol supported by WebSphere MQ for Linux, Version 6.0 (POWER platform) is TCP/IP. The tasks in this section explain how to verify your installation and use TCP in the examples; if you are using an alternative protocol, refer to the *WebSphere MQ Intercommunication* manual.

The verification procedure assumes that both systems are Linux or UNIX® machines; if this is not the case, some of the commands are different (for details, refer to the documentation for that system).

Note: WebSphere MQ object definitions are case-sensitive. Any text entered as an MQSC command in lowercase is converted automatically to uppercase unless you enclose it in single quotation marks. Make sure that you type the examples exactly as shown.

Setting up the sender server

In order to verify a server-to-server installation you must first set up a sender server. From a shell window, follow these steps to set up the sender server.

1. Log in as a user in the `mqm` group.
2. Create a default queue manager called `saturn.queue.manager` with the following command:

   ```
   crtmqm -q saturn.queue.manager
   ```

   Messages tell you that the queue manager has been created, and that the default WebSphere MQ objects have been created.
3. To start the queue manager, type:

   ```
   strmqm
   ```

   A message tells you when the queue manager has started.
4. Start MQSC commands by typing:

   ```
   runmqsc
   ```

A message tells you that an MQSC session has started. MQSC had no command prompt.

5. Define a local queue called TRANSMIT1.QUEUE (to be used as a transmission queue) by entering the following command:

   ```
   define qlocal (transmit1.queue) usage (xmitq)
   ```

   A message tells you when the queue has been created.

6. Define a local definition of the remote queue with the following command:

   ```
   define qremote (local.def.of.remote.queue) rname (orange.queue)
   rqmname ('venus.queue.manager') xmitq (transmit1.queue)
   ```

   The name specified by the **rname** parameter must be the same as the name of the queue to which you are sending the message (ORANGE.QUEUE on the receiver workstation).

7. Define a sender channel with the following command:

   ```
   define channel (first.channel) chltype (sdr)
   conname ('con-name(port)') xmitq (transmit1.queue) trptype (tcp)
   ```

   The value *con-name* is the TCP address of the receiver workstation, and *port* is the port number, port 1414 is the default port number.

8. End MQSC by typing:

   ```
   end
   ```

   Some messages are displayed, followed by the command prompt.

You have now defined the following objects:

- A default queue manager called saturn.queue.manager
- A transmission queue called TRANSMIT1.QUEUE
- A local definition of a remote queue called LOCAL.DEF.OF.REMOTE.QUEUE
- A sender channel called FIRST.CHANNEL

Now to set up the receiver server so that you can verify your server-to-server installation, see "Setting up the receiver server."

Setting up the receiver server

After you have completed the task, "Setting up the sender server" on page 21, follow these steps to set up the receiver server:

1. Log in as a user in the mqm group.
2. Create a default queue manager called venus.queue.manager by entering the following command:

   ```
   crtmqm -q venus.queue.manager
   ```

   Messages tell you that the queue manager has been created, and that the default WebSphere MQ objects have been created.

3. To start the queue manager, type:

   ```
   strmqm
   ```

   A message tells you when the queue manager has started.

4. Enable MQSC commands by typing:

   ```
   runmqsc
   ```

A message tells you that an MQSC session has started. MQSC has no command prompt.

5. Define a local queue called ORANGE.QUEUE by entering the following command:

   ```
   define qlocal (orange.queue)
   ```

   A message tells you when the queue has been created.

6. Define a listener by entering the following command:

   Note: If you do not specify the port that the listener should listen on, the default of 1414 is used. If you specified a port other than 1414 in step 7 of "Setting up the sender server" on page 21, you must include the port parameter in the command, as shown below.

   ```
   define listener (listener1) trptype (tcp) control (qmgr) port (port_number)
   ```

   Where

   port_number
   is the name of the port the listener should run on. This must be the same as the number used when defining your sender channel.

7. Start the listener by entering the following command:

   ```
   start listener (listener1)
   ```

   Note: It is not recommended to start the listener in the background from any shell that automatically lowers the priority of background processes.

8. Define a receiver channel with the following command:

   ```
   define channel (first.channel) chltype (rcvr) trptype (tcp)
   ```

   A message tells you when the channel has been created.

9. End MQSC by typing:

   ```
   end
   ```

   Some messages are displayed, followed by the command prompt.

You have now defined the following objects:

- A default queue manager called venus.queue.manager
- A queue called ORANGE.QUEUE
- A receiver channel called FIRST.CHANNEL

Now to test communications between your sender and receiver workstations, see "Testing communication between the servers."

Testing communication between the servers

After completing, "Setting up the sender server" on page 21, and "Setting up the receiver server" on page 22, use this topic to test communications between sender and receiver workstations using sample programs. Use the **amqsput** sample program to put a message from the sender server to a queue at the receiver server, and the **amqsget** sample program on the receiver server to get the message from the queue:

1. Log in to both servers as a user in the mqm group.
2. If the queue managers on the two servers have stopped, restart them now by typing the following on both servers:

   ```
   strmqm
   ```

Also, if the queue managers had stopped, on the **receiver** server, start the listener program using the following command:

```
runmqlsr
```

3. On the **sender** server, start the sender channel by entering the following command:

   ```
   runmqchl -c FIRST.CHANNEL -m saturn.queue.manager
   ```

   The receiver channel on the receiver server starts automatically when the sender channel starts.
4. On the **sender** server, change into the `/opt/mqm/samp/bin` directory, which contains the sample programs.
5. To put a message on the local definition of the remote queue (which in turn specifies the name of the remote queue), use the following command:

   ```
   ./amqsput LOCAL.DEF.OF.REMOTE.QUEUE
   ```

   You will see the following messages:

   ```
   Sample amqsput0 start
   target queue is LOCAL.DEF.OF.REMOTE.QUEUE
   ```
6. Type some message text on one or more lines, followed by a blank line. You will see the following message:

   ```
   Sample amqsput0 end
   ```

   Your message is now on the queue and the command prompt is displayed again.
7. On the **receiver** server, change into the `/opt/mqm/samp/bin` directory, which contains the sample programs.
8. To get the message from the queue at the receiver, enter the following command:

   ```
   ./amqsget ORANGE.QUEUE
   ```

   The sample program starts, and your message is displayed. After a pause, the sample ends and the command prompt is displayed again.

You have now successfully verified the server-to-server installation.

Verifying the installation using the JMS Postcard application

Use the **JMS Postcard** application to verify that WebSphere MQ is successfully installed, the associated communication links are working properly, and that WebSphere MQ Java Messaging Support is successfully installed.

To set up your system to use the **JMS Postcard**, see "Setting up your system to run the JMS postcard" on page 25.

To use the **JMS Postcard** application to verify a *local* installation (which does not have any communication links with other WebSphere MQ installations), see "Using the JMS postcard application to verify a local installation" on page 26.

To use the **JMS Postcard** application to verify communication between your machine and the machine of another named user, where that machine is running WebSphere MQ and using TCP/IP, see "Using the JMS postcard application to verify a server-to-server installation" on page 27.

Setting up your system to run the JMS postcard

Before you can run the **JMS Postcard** application, you must ensure that:

- You must install the optional WebSphere MQ Java component.
- You must have a working JRE (Java Runtime Environment).
- You are a member of the WebSphere MQ administrators group (mqm).

Setting the PATH variable:

To use any Java language application described in this book, including the JMS postcard application, you must have a Java Runtime Environment (JRE) installed. For further information about supported JREs, see "Checking prerequisite hardware and software" on page 4 in this book. To ensure that WebSphere MQ installation can use the installed JRE, ensure that the location of the JRE is set in your PATH environment variable.

See the documentation accompanying your JRE or JDK to find out where the JRE or JDK will be installed to.

On Linux POWER platform, the 32-bit and 64-bit JDKs are typically installed to different locations, for example, the 32-bit JDK is located in `/opt/IBMJava2-ppc-142` and the 64-bit JDK is located in `/opt/IBMJava2-ppc64-142`. Ensure that the PATH variable is correctly set, to use the JMS Postcard application you need to use a 32-bit JDK. You can check the version installed using the following command:

```
java -version
```

For further information about using a JDK see "Checking prerequisite hardware and software" on page 4.

Setting environment variables using setjmsenv:

Before performing this task ensure you have removed any hardcoded links to the Java libraries as described in "Setting the PATH variable."

For WebSphere MQ Version 6.0 Java scripts to function properly a number of environment variables must be set. The **setjmsenv** script can be used to set these variables, and is located in `/opt/mqm/java/bin`. The environment variables that **setjmsenv** sets are as follows:

CLASSPATH	`/opt/mqm/java/lib/com.ibm.mq.jar:`
	`/opt/mqm/java/lib/com.ibm.mqjms.jar:`
	`/opt/mqm/samp/java/base:`
	`/opt/mqm/samp/java/jms:`

MQ_JAVA_INSTALL_PATH	`/opt/mqm/java`
MQ_JAVA_DATA_PATH	`/var/mqm`
MQ_JAVA_LIB_PATH	`/opt/mqm/java/lib (32-bit libraries)`

Use either the 32-bit libraries or the 64-bit libraries. Use the 64-bit libraries only if you are running your application in a 64-bit Java virtual machine (JVM) on a 64-bit platform. Otherwise, use the 32-bit libraries.

The setjmsenv script sets MQ_JAVA_LIB_PATH to the location of the 32-bit libraries so that you can run the postcard application. If you use setjmsenv to set your environment variables you need to set your PATH to use a 32-bit JVM as described in "Setting the PATH variable" on page 25.

You can choose to use this script in a variety of ways:

- You can use the **setjmsenv** script as a basis for setting the required environment variables, as shown in the table, or add them to the .profile using a text editor. If you have a non-typical setup, edit the script contents as necessary.
- Alternatively, you can run **setjmsenv** in every session from which JMS startup scripts are to be run. If you choose this option you need to run the **setjmsenv** script in every shell window you start, during the JMS verification process by typing:

  ```
  . ./setjmsenv
  ```

For further information about using Java with WebSphere MQ, see the *Using Java* book.

When you have configured your system you are able to verify that WebSphere MQ Version 6.0 has installed correctly as described in "Verifying your installation" on page 19.

Using the JMS postcard application to verify a local installation

To verify that the local installation is working, you can run two instances of the JMS Postcard application on the same machine and send messages between them. This shows that WebSphere MQ messaging is working correctly on the machine, and that WebSphere MQ Java Messaging support is successfully installed.

Note: The JMS Postcard application has a graphical interface, to view this interface, your system requires the ability to view a graphical display. If you want the JMS Postcard application to use font and color settings different from the Java Virtual Machine defaults, change the Postcard.ini file. For more information see *WebSphere MQ Using Java*.

1. Log on as a user in group mqm.
2. Change directory to `/opt/mqm/java/bin`
3. If you have not already run setjmsenv as described in "Setting environment variables using setjmsenv" on page 25 do so now.
4. Run the postcard shell script.

   ```
   ./postcard
   ```

   If there are no queue managers on your machine, you are invited to run the Default Configuration wizard to create a queue manager to use with the JMS Postcard application before signing on to the JMS postcard application.

   If you already have a queue manager on your machine you will go straight to the JMS sign on window.
5. At the JMS Postcard - Sign On window , type in a nickname to use to send messages within the postcard application (for example, `user1`).
6. Select the queue manager to use as the mailbox:
   - If the only queue manager on your machine is the default queue manager that you created by running the Default Configuration wizard, this queue manager is used automatically as your mailbox for postcards.

- If you have created one or more of your own queue managers, but you have not run the Default Configuration wizard, select the appropriate queue manager from the list displayed.
- If you have run the Default Configuration wizard and you want to use the default queue manager, but there is more than one queue manager on your machine, select the **Advanced** checkbox, then select **Use Default Configuration as mailbox**.
- If you have run the Default Configuration wizard and also created one or more of your own queue managers, and you do not want to use the default queue manager, select the **Advanced** checkbox, select **Choose queue manager as mailbox**, then select the appropriate queue manager from the list displayed.

When your selection is complete, click **OK** to display your first postcard window.

7. Run the Postcard shell script again in a different shell window. This opens a second postcard window.
8. The JMS Postcard - Sign On panel is displayed again. Type in a second nickname to use to send messages within the Postcard application (for example, `user2`).
9. Repeat the selection of the queue manager that you want to use as the mailbox (as described in step 5). The queue manager you select for this second postcard must either be the same queue manager, be in the same cluster as the queue manager for the first postcard, or communication links must have been set up between them. You now have two postcards, one with the nickname `user1` and one with the nickname `user2`.
10. In one of the postcards (for example, `user1`), enter the nickname for the other postcard application in the **To:** field and the queue manager it is using in the **On:** field.
11. Type a message in the **Message:** field and click **Send**.
12. The **Postcards sent and received** area of the postcard shows details of the message. In the sending postcard, the message is displayed as *sent*. In the receiving postcard, the message is displayed as *received*.
13. From the receiving postcard, double-click the message in the **Postcards sent and received** area to view it.

Depending on your situation, you might want to do the following:

- Install WebSphere MQ on other machines. Follow the same installation procedure that you used for the first machine. Ensure that you use the Join Default Cluster window in the Default Configuration wizard to add the other machines to your first machine's cluster.
- Install the WebSphere MQ client on other machines. See the Chapter 3, "Installing a WebSphere MQ client," on page 31.
- Continue with further administration tasks. See the *WebSphere MQ System Administration Guide*.

Using the JMS postcard application to verify a server-to-server installation

To verify that the communication between two machines, the sender of the message and the receiver, are working correctly, and that the WebSphere MQ Java messaging support is successfully installed, you can use the JMS Postcard application. Both machines must use TCP/IP.

To use the **JMS Postcard** application for this type of verification one of the following must be true;

- Both queue managers must be in the same cluster, this is the simplest method. To ensure that both queue managers are in the same cluster you can run the **JMS Postcard** application before creating any local queue managers on each machine. The **JMS Postcard** application detects that there are no local queue managers defined for that machine, and displays the Default Configuration wizard so that you can create the default queue managers and link them to the default cluster. This topic describes how to use the Default Configuration wizard.

  You can use the **JMS Postcard** application with existing queue managers, as long as both queue managers belong to the same cluster. If you have already completed the Default Configuration wizard but did not put the two queue managers into the same cluster you can create your own new queue managers on both machines, create a cluster, and ensure that the queue managers that you create on each machine belong to the same cluster.
- Alternatively if the queue managers are not in the same cluster you can configure channels to communicate between the two machines. For instructions on how to see up the channels see, "Setting up the sender server" on page 21 and "Setting up the receiver server" on page 22. Once you have set up communication you can use the postcard application, starting at step 6.

On the sender machine:

1. Log on as a user in group mqm.
2. Change directory to `/opt/mqm/java/bin`
3. If you have not already run setjmsenv as described in "Setting environment variables using setjmsenv" on page 25 do so now.
4. Run the postcard shell script.

   ```
   ./postcard
   ```

   If there are no queue managers on your machine, you are invited to run the Default Configuration wizard to create a queue manager to use with the JMS Postcard application before signing on to the JMS postcard application.

   If you already have a queue manager on your machine you will go straight to the JMS sign on window (step 6). You can use the **JMS Postcard** application with existing queue managers, as long as both queue managers belong to the same cluster. If your existing queue manager does not belong to the appropriate cluster refer to the introduction of this topic for information on how to proceed.
5. Work through the Default Configuration wizard. When you get to the option to join the queue manager to the default cluster, tick the checkbox. On the next screen select **yes, make it the repository for the cluster**. Once you have completed the wizard you are taken back to the JMS Postcard - Sign On window.
6. At the JMS Postcard - Sign On window , type in a nickname to use to send messages within the postcard application (for example, `user1`).
7. Select the queue manager to use as the mailbox:
   - If the only queue manager on your machine is the default queue manager that you created by running the Default Configuration wizard, this queue manager is used automatically as your mailbox for postcards.
   - If you have created one or more of your own queue managers, but you have not run the Default Configuration wizard, select the appropriate queue manager from the list displayed.

- If you have run the Default Configuration wizard and you want to use the default queue manager, but there is more than one queue manager on your machine, select the **Advanced** checkbox, then select **Use Default Configuration as mailbox**.
- If you have run the Default Configuration wizard and also created one or more of your own queue managers, and you do not want to use the default queue manager, select the **Advanced** checkbox, select **Choose queue manager as mailbox**, then select the appropriate queue manager from the list displayed.

When your selection is complete, click **OK** to display your second postcard window.

On the receiver machine:

1. Log on as a user in group mqm.
2. Change directory to `/opt/mqm/java/bin`
3. If you have not already run setjmsenv as described in "Setting environment variables using setjmsenv" on page 25, do so now.
4. Run the postcard shell script.

   ```
   ./postcard
   ```

   If there are no queue managers on your machine, you are invited to run the Default Configuration wizard to create a queue manager to use with the JMS Postcard application before signing on to the JMS postcard application.

   If you already have a queue manager on your machine you will go straight to the JMS Sign On window (step 6). You can use the **JMS Postcard** application with existing queue managers, as long as both queue managers belong to the same cluster. If your existing queue manager does not belong to the appropriate cluster refer to the introduction of this topic for information on how to proceed.
5. Work through the Default Configuration wizard:
   a. When you get the option to join the queue manager to the default cluster, tick the checkbox.
   b. In the next window click **No another computer has already joined the cluster as a repository**. Click Next.
   c. When requested, enter the location of the repository, by typing the machine name of the sender machine. Click Next.
   d. Complete the Default Configuration wizard. Once you have completed the wizard you are taken back to the JMS Postcard Application- Sign On window.
6. At the JMS Postcard - Sign On window, type in a nickname to use to send messages within the postcard application (for example, `user2`).
7. Select the queue manager to use as the mailbox:
   - If the only queue manager on your machine is the default queue manager that you created by running the Default Configuration wizard, this queue manager is used automatically as your mailbox for postcards.
   - If you have created one or more of your own queue managers, but you have not run the Default Configuration wizard, select the appropriate queue manager from the list displayed.

- If you have run the Default Configuration wizard and you want to use the default queue manager, but there is more than one queue manager on your machine, select the **Advanced** checkbox, then select **Use Default Configuration as mailbox**.
- If you have run the Default Configuration wizard and also created one or more of your own queue managers, and you do not want to use the default queue manager, select the **Advanced** checkbox, select **Choose queue manager as mailbox**, then select the appropriate queue manager from the list displayed.

When your selection is complete, click **OK** to display your first postcard window.

8. In one of the postcards (for example, `user1`), enter the nickname for the other postcard application the **To:** field and the queue manager it is using in the **On:** field.
9. Type a message in the **Message:** field and click **Send**.
10. The **Postcards sent and received** area of the postcard shows details of the message. In the sending postcard, the message is displayed as *sent*. In the receiving postcard, the message is displayed as *received*.
11. In the sent and received area of the postcard, details of the new message are displayed. The message is displayed as *received*. When this message arrives, this verifies that WebSphere MQ and the Java messaging support are correctly installed and that your communication link between the two machines is working correctly.

When all installation and verification is complete, you are ready to start using WebSphere MQ (see the *WebSphere MQ System Administration Guide*).

Chapter 3. Installing a WebSphere MQ client

This chapter describes how to install a WebSphere MQ Version 6.0 client. The information covers topics such as preparing for installation and verifying your installation, as well as installation itself. If you already have an installation of WebSphere MQ, and are migrating to WebSphere MQ Version 6.0 see "Migrating to WebSphere MQ Version 6.0 and upgrading your operating system" on page 1 before installing WebSphere MQ Version 6.0.

WebSphere MQ for Linux can be installed as a server or a client.

A WebSphere MQ client is a component that allows an application running on one system to communicate with a queue manager running on another system. The output from the call is sent back to the client, which passes it back to the application.

A WebSphere MQ server is an installation of one or more queue managers that provide queueing services to one or more clients. All the WebSphere MQ objects, for example queues, exist only on the queue manager machine (the WebSphere MQ server machine), and not the client. A WebSphere MQ server can also support local WebSphere MQ applications. To install a WebSphere MQ server see, Chapter 2, "Installing a WebSphere MQ server," on page 3.

It is possible to have both a server and a client installation on the same machine, for instructions on how to do this see, Chapter 4, "Installing a client on the same machine as a server," on page 47.

See the *WebSphere MQ System Administration Guide* for an introduction to WebSphere MQ concepts and objects.

For information on the components that can be included in the server and client installations see, "WMQ Components" on page 16.

The following set of tasks take you through the process of installing a WebSphere MQ client, complete all of these tasks in sequence.

- "Checking prerequisite hardware and software" on page 4
- "Installing WebSphere MQ" on page 41
- "Verifying the client installation" on page 42

Preparing to install

Before you install WebSphere MQ, complete the following tasks.

- "Checking hardware and software requirements" on page 32
- "Creating WebSphere MQ file systems" on page 11
- "Setting up the user ID and group ID" on page 13

Additionally, if you require messages in a language other than U.S. English see, "Displaying messages in your national language" on page 13.

Checking hardware and software requirements

This section details the operating system requirements, the prerequisite software and optional software required for using WebSphere MQ Version 6.0. These requirements differ in some cases depending on the system hardware on which you will be running WebSphere MQ. Please see the appropriate section for your installation.

Checking the operating environment

Before installing WebSphere MQ Version 6.0, you must check that your system meets the hardware and software requirements set out in this topic.

Note: WebSphere MQ does not support host names that contain spaces. If you install WebSphere MQ on a computer with a host name that contains spaces, you will be unable to create any queue managers.

Hardware

WebSphere MQ for Linux, Version 6.0 (x86 platform) runs on any machine that supports the x86 machine architecture. WebSphere MQ for Linux, Version 6.0 (POWER platform) runs on any machine that supports the POWER machine architecture. WebSphere MQ for Linux, Version 6.0 (zSeries platform) runs on any machine that supports the zSeries machine architecture.

Operating System, WebSphere MQ for Linux, Version 6.0 (x86 platform)

WebSphere MQ for Linux, Version 6.0 (x86 platform) has been tested with the following distributions:

- Red Hat Enterprise Linux AS V3.0 plus Update 2
- SuSE Linux Enterprise Server (SLES) V8 plus Service Pack 3
- SuSE Linux Enterprise Server (SLES) V9

Operating System, WebSphere MQ for Linux, Version 6.0 (POWER platform)

WebSphere MQ for Linux, Version 6.0 (POWER platform)) has been tested with the following distributions:

- Red Hat Enterprise Linux AS V3.0 plus Update 2
- SuSE Linux Enterprise Server (SLES) V9

Operating System, WebSphere MQ for Linux, Version 6.0 (zSeries platform)

WebSphere MQ for Linux, Version 6.0 (zSeries platform) has been tested with the following distributions:

- Red Hat Enterprise Linux AS V3.0 plus Update 2
- SuSE Linux Enterprise Server (SLES) V8 plus Service Pack 3
- SuSE Linux Enterprise Server (SLES) V9

Threading Models

The support for threads on Linux is provided by the pthreads library.

- Most distributions based upon the 2.4 series kernels, such as those based upon UnitedLinux 1.0, include the LinuxThreads pthreads library. This threading library differs from most others in that the 'getpid()' function returns a different value for each thread in a process and is identified as a separate process in the output of the 'ps' command.
- Most distributions based upon the 2.6 series kernels, such as those based upon UnitedLinux 2.0, include the Native Posix Threading Library (NPTL). NPTL was written as the replacement to LinuxThreads and relies on updates made to recent kernels to overcome some of the shortfalls of LinuxThreads. Notably, threads within a process return the same pid from calls to 'getpid()' and closer compliance to the POSIX Threads specification.

Both of these threading models are supported for use with the WebSphere MQ Queue Manager and applications which connect to WebSphere MQ.

Other threading libraries exist as replacements for these threading libraries, however, none of these replacements are supported for use with the WebSphere MQ Queue Manager or applications which connect to WebSphere MQ.

Connectivity Requirements

The network protocols supported by WebSphere MQ for Linux, Version 6.0 (x86 platform) and WebSphere MQ for Linux, Version 6.0 (zSeries platform) are TCP/IP and LU6.2 . TCP/IP is part of the WebSphere MQ for Linux, Version 6.0 (x86 platform) and WebSphere MQ for Linux, Version 6.0 (zSeries platform) operating systems. If you want to use the SNA LU6.2 support on WebSphere MQ for Linux, Version 6.0 (x86 platform) or WebSphere MQ for Linux, Version 6.0 (zSeries platform) you need the IBM Communications Server for Linux Version 6.2. The Communications Server is available as a PRPQ product from IBM. For more details, see: http://www.ibm.com/software/network/commserver/about.

The network protocol supported by WebSphere MQ for Linux, Version 6.0 (POWER platform) is TCP/IP. You can use any communications hardware supporting TCP/IP.

Storage Requirements

The storage requirements for the WebSphere MQ for Linux depend on which components you install, and how much working space you need. This, in turn, depends on the number of queues that you use, the number and size of the messages on the queues, and whether the messages are persistent. You also require archiving capacity on disk, tape or other media.

Component	Location of storage	Storage Requirement in MB (x86 platform)	Storage Requirement in MB (z/Series platform)	Storage Requirement in MB (POWER platform)
Runtime	/usr	25.13	25.10	30.79
Client	/opt	2.73	2.73	4.95
Sample programs	/opt	4.77	4.78	5.26
Java messaging	/opt	6.84	6.84	6.89

Extended Transactional Client	/opt	0.04	0.04	0.13
Man pages	/usr	0.71	0.71	0.72
German Message catalogs	/usr	0.47	0.57	0.61
Spanish Message catalogs	/usr	0.56	0.56	0.57
French Message catalogs	/usr	0.56	0.56	0.61
Italian Message catalogs	/usr	0.55	0.56	0.57
Japanese Message catalogs	/usr	0.50	0.50	0.51
Korean Message catalogs	/usr	0.44	0.44	0.44
Brazilian Portuguese Message catalogs	/usr	0.53	0.53	0.55
Simplified Chinese Message catalogs	/usr	0.33	0.33	0.33
Traditional Chinese Message catalogs	/usr	0.35	0.35	0.36

Table 7. Storage requirements for prerequisite filesets

Component	Location of storage	Storage Requirement in MB (x86 platform)	Storage Requirement in MB (z/Series platform)	Storage Requirement in MB (POWER platform)
gsk7bas-7.0-x.x	**/usr**	**10.4**	**13.3**	**11.3**
gsk7bas64-7.0-x.x	**/usr**	**not applicable**	**not applicable**	**12.5**
IBMJava2-SDK-1.4.2-0.0	**/opt**	**80.5**	**71.3**	**82.5**
IBMJava2-SDK-ppc64-1.4.2-0.0	/opt	not applicable	not applicable	82.1

You can use the df command to determine the amount of free space on your system.

Disk storage is also required for

- Prerequisite software
- Optional software
- Your application programs

After checking the requirements in this topic, proceed to “Prerequisite software for using SSL” on page 7.

Prerequisite software for using SSL

This section describes the prerequisites for using SSL, minimum supported software levels are given. Later levels, if any, are supported unless otherwise stated. These prerequisites apply to both client and server installations of WebSphere MQ.

Installing the g++ version runtime support

If you intend to run SSL channels then you must have the g++ version 3.2 runtime libraries installed. The GNU g++ libraries are called libgcc_s.so and `libstdc++.so.5` and, on a Red Hat sytem with `libgcc` and `libstdc++5` RPMs installed, are found in the directory `/usr/lib`.

If you have a distribution that does not install `libgcc_s.so` and `libstdc++.so.5` into `/usr/lib`, then you will have to do one of the following:

- Locate and install the packages from your distribution vendor that contain these libraries.
- Install the GNU gcc and g++ version 5 compilers from another location, for example the home page for the GNU compiler collection at URL:.

  `http://gcc.gnu.org/`

  Ensure that the libraries listed above are included in `/usr/lib` after installation.

IBM Global Security Kit V7

If you want to use the SSL support, you need IBM Global Security Kit V7. This is supplied with WebSphere MQ as one of the components available for installation. If you are migrating from WebSphere MQ Version 5.3 and have no other requirement for the IBM Global Security Kit V6, you can uninstall it using the process described in "Installing a WebSphere MQ server" on page 15, the package name is `gskbas-6.0-n.nn.<arch>.rpm`.

Checking optional software - x86 platform

Compilers

The following compilers are supported for WebSphere MQ for Linux, Version 6.0.

C applications/C++ applications

- GNU C Compiler (gcc) and g++ Version 2.9.5
- GNU C Compiler (gcc) and g++ Version 3.2
- GNU C Compiler (gcc) and g++ Version 3.3
- GNU C Compiler (gcc) and g++ Version 3.4

Note: The C++ support libraries are installed in directories whose names match the compiler version, `/opt/mqm/lib/<version>`, and links are placed from `/opt/mqm/lib` to the default version, 3.2.

COBOL applications

- Micro Focus Server Express, V4.0

Java applications

- IBM Software Developer's Kit (SDK), Java 2 Technology Edition for Linux Version 1.4.2

Transaction Monitors

The following transaction processing monitors (coordination through X/Open XA interface) are supported for use with the Extended Transactional Client:

- BEA Tuxedo 8.1

- WebLogic V8.1 is supported when used with the Red Hat Enterprise Linux 3.0 operating system.
- IBM WebSphere Application Server (WAS) V5.1 is supported when used with the SUSE Linux Enterprise Server (SLSE) 8 operating system.

WebSphere MQ for Linux, Version 6.0 supports WebSphere Application Server as an XA coordinator. For more information about the WebSphere MQ application adaptor, see the WebSphere Application Server Enterprise Edition WebSphere MQ Application Adaptor Development Guide, SC09-4444.

Java Messaging and SOAP transport

If you want to use Java Messaging and SOAP (Simple Object Access Protocol) Support, you need a Java Runtime Environment Version 1.4.2 or later. An appropriate runtime environment is provided in the JDK provided on the WebSphere MQ Server CD- ROM. See "WMQ Components" on page 16 for package information.

The JDK available on the server CD-ROM is:

- IBM 32-bit Software Developer's Kit (SDK) for Linux on Intel architecture, Java 2 Technology Edition, Version 1.4.2

This JDK provides support for SOAP and are FIPS 140-2 compliant. For a list of alternative JDKs, see

www.ibm.com/software/integration/websphere/mqplatforms/supported.html

For WebSphere MQ Version 6.0 Java scripts to function properly a number of environment variables must be set. For information on setting these environment variables and general information about using Java with WebSphere MQ see, *Using Java*.

For further information about SOAP see, the *WebSphere MQ Transport for SOAP* book.

If you use a JDK other than the one supplied by WebSphere MQ Version 6.0 you should be aware that:

- The JDK may not be FIPS level 140-2 compliant and by using it with WebSphere MQ, will not comply the FIPS 140-2 standards.
- SOAP is not supported.

Checking optional software - POWER platform

Compilers

The following compilers are supported for WebSphere MQ for Linux, Version 6.0.

C applications/C++ applications

- GNU C Compiler (gcc) and g++ Version 3.2
- GNU C Compiler (gcc) and g++ Version 3.3
- GNU C Compiler (gcc) and g++ Version 3.4

Note: The C++ support libraries are installed in directories whose names match the compiler version, `/opt/mqm/lib/<version>`, and links are placed from `/opt/mqm/lib` to the default version, 3.2.

COBOL applications

- Micro Focus Server Express, V4.0

Java applications

- IBM Software Developer's Kit (SDK), Java 2 Technology Edition for Linux on POWER Version 1.4.2

Java Messaging and SOAP transport

If you want to use Java Messaging and SOAP (Simple Object Access Protocol) Support, you need a Java Runtime Environment Version 1.4.2 or later. An appropriate runtime environment is provided in the JDKs provided on the WebSphere MQ Server CD- ROM. See "WMQ Components" on page 16 for package information.

The JDKs available on the server CD-ROM are:

- IBM 32-bit Software Developer's Kit (SDK) for Linux for iSeries and pSeries, Java 2 Technology Edition, Version 1.4
- IBM 64-bit Software Developer's Kit (SDK) for Linux for iSeries and pSeries, Java 2 Technology Edition, Version 1.4

These JDKs provide support for SOAP and are FIPS 140-2 compliant. For a list of alternative JDKs, see

www.ibm.com/software/integration/websphere/mqplatforms/supported.html

For WebSphere MQ Version 6.0 Java scripts to function properly a number of environment variables must be set. For information on setting these environment variables and general information about using Java with WebSphere MQ see, *Using Java*.

For further information about SOAP see, the *WebSphere MQ Transport for SOAP* book.

If you use a JDK other than the one supplied by WebSphere MQ Version 6.0 you should be aware that:

- The JDK may not be FIPS level 140-2 compliant and by using it with WebSphere MQ, will not comply the FIPS 140-2 standards.
- SOAP is not supported.

On the POWER platform, the 32-bit and 64-bit JDKs are typically installed to different locations, for example, the 32-bit JDK is located in `/opt/IBMJava2-ppc-142` and the 64-bit JDK is located in `/opt/IBMJava2-ppc64-142`. Ensure that the PATH variable is correctly set for your applications that use Java.

Checking optional software - z/Series platform

Compilers

The following compilers are supported for WebSphere MQ for Linux, Version 6.0.

C applications/C++ applications

- GNU C Compiler (gcc) and g++ Version 2.9.5
- GNU C Compiler (gcc) and g++ Version 3.2
- GNU C Compiler (gcc) and g++ Version 3.3

- GNU C Compiler (gcc) and g++ Version 3.4

Note: The C++ support libraries are installed in directories whose names match the compiler version, `/opt/mqm/lib/<version>`, and links are placed from `/opt/mqm/lib` to the default version, 3.2.

COBOL applications

- Micro Focus Server Express, V4.0

Java applications

- IBM Software Developer's Kit (SDK), Java 2 Technology Edition for Linux on zSeries Version 1.4.2

Transaction Monitors

The following transaction processing monitors (coordination through X/Open XA interface) are supported for use with the Extended Transactional Client:

- BEA Tuxedo 8.1 is supported when used with the SUSE Linux Enterprise Server (SLSE) 8 operating system.
- IBM WebSphere Application Server (WAS) V5.1 is supported when used with the SUSE Linux Enterprise Server (SLSE) 8 operating system.

WebSphere MQ for Linux, Version 6.0 supports WebSphere Application Server as an XA coordinator. For more information about the WebSphere MQ application adaptor, see the WebSphere Application Server Enterprise Edition WebSphere MQ Application Adaptor Development Guide, SC09-4444.

Java Messaging and SOAP transport

If you want to use Java Messaging and SOAP (Simple Object Access Protocol) Support, you need a Java Runtime Environment Version 1.4.2 or later. An appropriate runtime environment is provided in the JDK provided on the WebSphere MQ Server CD- ROM. See "WMQ Components" on page 16 for package information.

The JDK available on the server CD-ROM is:

- IBM 31-bit Software Developer's Kit (SDK) for Linux on zSeries, Java 2 Technology Edition, Version 1.4

This JDK provides support for SOAP and are FIPS 140-2 compliant. For a list of alternative JDKs, see

www.ibm.com/software/integration/websphere/mqplatforms/supported.html

For WebSphere MQ Version 6.0 Java scripts to function properly a number of environment variables must be set. For information on setting these environment variables and general information about using Java with WebSphere MQ see, *Using Java*.

For further information about SOAP see, the *WebSphere MQ Transport For SOAP* book.

If you use a JDK other than the one supplied by WebSphere MQ Version 6.0 you should be aware that:

- The JDK may not be FIPS level 140-2 compliant and by using it with WebSphere MQ, will not comply the FIPS 140-2 standards.
- SOAP is not supported.

Creating WebSphere MQ file systems

The installation directory for the WebSphere MQ product code is /opt/mqm. Working data is stored in /var/mqm. You cannot change these locations. The GSKit must also be installed into its default location.

The topics contained in this section describe how to prepare your file system for installing WebSphere MQ. Perform these tasks before installation.

Creating a file system for the product code

This topic describes how to prepare the /opt/mqm file system into which the WebSphere MQ code will be installed and what to do should you not have enough storage space available in the file system.

The WebSphere MQ product code is installed in /opt/mqm. If you cannot install the product code in the /opt/mqm file system because the file system is too small to contain the product, you can do one of the following:

1. Create a new file system and mount it as /opt/mqm. If you choose this option, the new file system must be created and mounted before installing the product code.
2. Create a new directory anywhere on your machine, and create a symbolic link from /opt/mqm to this new directory. For example:

   ```
   mkdir /bigdisk/mqm
   ln -s /bigdisk/mqm /opt/mqm
   ```

   If you choose this option, the new directory must be created, and the link created, before installing the product code.

The file system into which the code is installed can be a remote network device, for example, NFS. However, you must define the mount options defined on that device to allow **setuid** programs, including those which are **setuid** root, to run.

Creating a file system for the working data

Before you install WebSphere MQ for Linux, create and mount a file system called /var/mqm. If possible, use a partition strategy with a separate volume for the WebSphere MQ data. This means that other system activity is not affected if a large amount of WebSphere MQ work builds up.

To determine the size of the /var/mqm file system for a server installation, consider:

- The maximum number of messages in the system at one time
- Contingency for message buildups, if there is a system problem
- The average size of the message data, plus 500 bytes for the message header
- The number of queues
- The size of log files and error messages
- The amount of SSL trace that is written to the /var/mqm/trace directory

Allow 50 MB as a minimum for a WebSphere MQ server and 15 MB as a minimum for a WebSphere MQ client.

Creating separate file systems for working data

You can also create separate file systems for your log data (/var/mqm/log) and error files (/var/mqm/errors). If possible, store log files on a different physical disk from the WebSphere MQ queues (/var/mqm).

If you create separate file systems:

- The /var/mqm and /var/mqm/log directories ***must*** be on a local file system.
- The /var/mqm/errors directory can be NFS mounted. However, if you choose to NFS-mount /var/mqm/errors, the error logs might be lost if the network fails.

If you are creating separate file systems, allow a minimum of 30 MB of storage for /var/mqm for a server installation and 15 MB of storage for /var/mqm for a client installation. Also allow 20 MB of storage for /var/mqm/log, and 4 MB of storage for /var/mqm/errors for both client and sever installations.

If you want to use individual queues that will hold more than 2 GB of data, you must enable /var/mqm to use large files.

The size of the log file depends on the log settings that you use. The minimum sizes above are for circular logging using the default settings. For further information on log sizes see the *WebSphere MQ System Administration Guide*.

Setting up the user ID and group ID

WebSphere MQ requires a user ID of the name mqm, with a primary group of mqm. The mqm user ID owns the directories and files that contain the resources associated with the product. Create the user ID and group IDs as described in the following sections.

- "Creating the user ID and group" on page 13
- "Adding existing user IDs to the group" on page 13

Creating the user ID and group

Create the required user ID and group ID ***before*** you install WebSphere MQ. Both user ID and group ID must be set to mqm. For stand-alone machines, you can create the new user ID and group IDs locally; for machines administered in a network information services (NIS) domain, an administrator must create the IDs on the NIS master server machine.

It is also suggested that you set the mqm user's home directory to /var/mqm.

Adding existing user IDs to the group

If you want to run administration commands, for example **crtmqm** (create queue manager) or **strmqm** (start queue manager), your user ID must be a member of the mqm group.

Users do not need mqm group authority to run applications that use the queue manager; it is needed only for the administration commands.

Displaying messages in your national language

Messages in U.S. English are automatically installed with WebSphere MQ.

If you require messages in a different language, ensure that you:

1. Install the appropriate message catalog (see "WMQ Components" on page 16).
2. To select messages in a different language, use the following command with the identifier for the language you want to install:

   ```
   export LANG=message identifier
   ```

   The message identifiers for the message catalogs are as follows:

   - de_DE (German)
   - es_ES (Spanish)
   - fr_FR (French)
   - it_IT (Italian)
   - ja_JP (Japanese)
   - ko_KR (Korean)
   - pt_BR (Brazilian Portuguese)
   - zh_CN (Simplified Chinese)
   - zh_TW (Traditional Chinese)

Installing WebSphere MQ

This chapter tells you how to install the WebSphere MQ for Linux client. If you want to install the WebSphere MQ server see Chapter 2, "Installing a WebSphere MQ server," on page 3.

Before you start the installation procedure, make sure you have prepared your system as described in "Preparing for installation" on page 3.

There are three types of WebSphere MQ clients:

Standard client
: This is the standard WebSphere MQ client. Use this client if you do **not** require Secure Sockets Layer (SSL) support. You can install this client from the Client CD-ROM or the Server CD-ROM.

Client with SSL
: This is the standard WebSphere MQ client with additional code to allow you to use SSL support. You can install the client with SSL from either the client or the server CD.

Extended Transactional Client
: This is additional code to allow a client application within the same unit of work to:

  - To put messages to, and get messages from, queues that are owned by the queue manager to which it is connected.
  - To update the resources of a resource manager other than a WebSphere MQ queue manager.

  You can only install this from the server CD.

For more information about SSL, see the *WebSphere MQ Security* book.

To install a WebSphere MQ server, see “Installing a WebSphere MQ server” on page 15.

If you want to install the client on the same machine as a WebSphere MQ server, see Chapter 4, “Installing a client on the same machine as a server,” on page 47.

Client Installation procedure

Before you start the installation procedure, make sure that you have prepared your system as described in “Preparing for installation” on page 3.

This installation procedure uses the Red Hat Package Manager (RPM) installer, enabling you to choose which components you want to install. The components (or filesets) are listed in “WMQ Components” on page 16; you must install at least the Runtime and Client components.

1. Log in as root.
2. If you are installing from the CD-ROM, mount the CD-ROM file system, and change into the directory corresponding to the type of client you are installing, `MQClient` or `MQClientwithSSL`
3. Run the `mqlicense.sh` script. If you want to view a text-only version of license, which can be read by a screen-reader, type:

   ```
   ./mqlicense.sh -text_only
   ```

   The license is displayed.

   If want to accept the license without it being displayed, you can run the `mqlicense.sh` script with the `-accept` option.

   ```
   ./mqlicense.sh -accept
   ```

   You must accept the license agreement before you can proceed with the installation.
4. Use the **rpm -ivh** command to install each component that you require. For example: On x86 architecture, enter the following commands for a minimum installation:

   ```
   rpm -ivh MQSeriesRuntime-6.0.0-0.i386.rpm MQSeriesClient-6.0.0-0.i386.rpm
   ```

   Note: Ensure that `/bin/sh` is a valid shell interpreter compatible with the Bourne shell, otherwise the post-installation configuration of will not complete successfully.

   If your Linux machine was not installed with RPM, you might see a prerequisites failure of `/bin/sh` when you try to install . This is because the RPM tables do not recognize that a valid shell interpreter is installed.

   If this occurs, you can reinstall the `/bin/sh` shell using RPM, or specify the RPM option `--nodeps` to disable dependency checking during installation of WebSphere MQ.

Verifying the client installation

The following set of tasks describes how to verify that the WebSphere MQ for Linux client has been correctly installed and configured.

To verify your WebSphere MQ client installation, you need a WebSphere MQ server with communication links with your client workstation. You can then complete the following tasks in order:

- "Setting up the server workstation"
- "Setting up the client workstation" on page 44
- "Testing communication between workstations" on page 45

The verification procedure assumes that:

- TCP/IP is configured and initialized on both the server and the client machines. If you are using SNA, refer to the *WebSphere MQ Intercommunication* manual.
- The WebSphere MQ server is installed on a Linux or UNIX machine; if this is not the case, some of the commands will be different (for details, refer to the *WebSphere MQ Clients* book).

Note: WebSphere MQ object definitions are case-sensitive. Any text entered as an MQSC command in lowercase is converted automatically to uppercase unless you enclose it in single quotation marks. Make sure that you type the examples exactly as shown.

Setting up the server workstation

In order to verify your installation you must first perform this task. From a shell window, use these steps to install a queue manager and a queue on the server:

1. Create a default queue manager called `saturn.queue.manager` by entering the following command:

   ```
   crtmqm -q saturn.queue.manager
   ```

   You will see messages telling you that the queue manager has been created, and that the default WebSphere MQ objects have been created.
2. To start the queue manager, type:

   ```
   strmqm
   ```

   A message tells you when the queue manager has started.
3. Enable MQSC commands by typing:

   ```
   runmqsc
   ```

   A message tells you that an MQSC session has started. MQSC has no command prompt.
4. Define a local queue called `QUEUE1` by entering the following command:

   ```
   define qlocal (queue1)
   ```

   A message tells you when the queue has been created.
5. Define a server-connection channel by entering the following command on one line:

   ```
   define channel (channel1) chltype (svrconn) trptype (tcp) mcauser ('mqm')
   ```

   A message tells you when the channel has been created.
6. Define a listener by entering the following command:

   Note: If you do not specify the port that the listener should listen on, by omitting the port parameter from the command below, the default of `1414` is used. If you want to specify a port other than 1414, you must include the port parameter in the command, as shown.

   ```
   define listener (listener1) trptype (tcp) control (qmgr) port (port_number)
   ```

Where

port_number
is the name of the port the listener should run on. This must be the same as the number used when defining your client-connection channel in "Setting up the client workstation."

7. Start the listener by entering the following command:
```
start listener (listener1)
```
8. Stop MQSC by typing:
```
end
```

You will see some messages, followed by the command prompt.

You have now defined the following objects on the server:

- A default queue manager called `saturn.queue.manager`
- A local queue called `QUEUE1`
- A server-connection channel called `CHANNEL1`

To continue with the verification process, see "Setting up the client workstation."

Setting up the client workstation

Before you complete this task you must have completed, "Setting up the server workstation" on page 43.

When a WebSphere MQ application is run on the WebSphere MQ client, the following information is required:

- The name of the MQI channel that connects the client to the server
- The communications protocol
- The address of the server

You provide this information by defining a client-connection channel with the name used for the server-connection channel defined on the server. This example uses the `MQSERVER` environment variable to define the client-connection channel.

1. Before starting, use the **ping** command to check that your TCP/IP software is correctly configured, and that your WebSphere MQ client and server TCP/IP sessions have been initialized. From the client, enter:
```
  ping server-hostname
or
  ping n.n.n.n
```

where

server-hostname
Is the host name of the server

n.n.n.n
Is the network address of the server

2. Press `Ctrl-C` to stop the **ping** command.
3. To create a client-connection channel, set the `MQSERVER` environment variable as follows:
```
  export MQSERVER='CHANNEL1/TCP/server-hostname(port)'
```

where

CHANNEL1
Is the name of the server-connection channel already defined on the server

TCP Is the communications protocol.

server-address
Is the TCP/IP host name of the server.

port Is optional and is the port number that the server is listening on, you specified this in step 6 of "Setting up the server workstation" on page 43. If you do not give a port number, WebSphere MQ uses:

- The one specified in the `qm.ini` file.
- If no value is specified in the `qm.ini` file, WebSphere MQ uses the port number identified in the TCP/IP services file for the service name WebSphere MQ. If this entry in the services file does not exist, a default value of `1414` is used.

The client-connection channel and server listener program must use the same port number.

To continue with the verification process, see "Testing communication between workstations."

Testing communication between workstations

Before you complete this task you must have completed, "Setting up the client workstation" on page 44.

On the WebSphere MQ client workstation, use the **amqsputc** sample program to put a message on the queue at the server workstation, and the **amqsgetc** sample program to get the message from the queue back to the client:

1. Change into the `/opt/mqm/samp/bin` directory, which contains the sample programs.
2. Put a message on the queue at the server using the following command:

   ```
   ./amqsputc QUEUE1 saturn.queue.manager
   ```

   This displays the following messages:

   ```
   Sample amqsput0 start
   target queue is QUEUE1
   ```

3. Type some message text on one or more lines, followed by a blank line. You will see the following message:

   ```
   Sample amqsput0 end
   ```

   Your message is now on the queue and the command prompt is displayed again.
4. To get the message from the queue located on the server, enter the following command:

   ```
   ./amqsgetc QUEUE1 saturn.queue.manager
   ```

   The sample program starts and your message is displayed. After a pause, the sample ends and the command prompt is displayed again.

You have now successfully verified the client installation.

Chapter 4. Installing a client on the same machine as a server

To install a WebSphere MQ for Linux client on a server machine, use the WebSphere MQ Server CD-ROM. Choose the Client component on the Server CD-ROM to install the client code on the server machine, and use the installation procedure described in “Installing a WebSphere MQ server” on page 15.

If you install a WebSphere MQ client on the same machine as a WebSphere MQ server, the client is not connected to the server automatically. Configure the communication channel (an MQI channel) between the client and the server, as described in “Verifying the client installation” on page 42.

Chapter 5. Applying maintenance

This section describes how to maintain WebSphere MQ for Linux.

- "Applying service" describes the process of how to install service updates.
- "Restoring the previous service level" on page 50 details how to restore the previous service level.

This information applies to both server and client installations of WebSphere MQ Version 6.0.

The latest information about service updates and downloads can be found on the Internet, at:

`http://www.ibm.com/software/integration/mqfamily/support/`

Applying service

Service is delivered in the form of RPM update images, which are applied using the RPM installation tool. Additional disk space is required for the update images, as the updated files are kept in `/opt/mqm/maintenance` directory, to allow service to be removed and the previous level restored. Do not delete or move this directory or the files it contains.

Updates are cumulative. You can install your chosen update directly, without installing any previous updates first. The maintenance might contain updates for one or more package. You must install those parts of an update that correspond to the packages that are installed on your machine.

1. Before installing maintenance you must end all WebSphere MQ activity:
   a. Log in as root.
   b. Use the **dspmq** command to display the state of all the queue managers on the system.
   c. Use the **endmqm** command to stop all running queue managers.
   d. Stop any listeners associated with the queue managers, using the command:
      ```
      endmqlsr -m QMgrName
      ```
   e. To check that you have stopped all necessary WebSphere MQ activity, type:
      ```
      ps -ef | grep mq
      ```
      Check that there are no processes listed that are running command lines beginning `amq` or `runmq`. Ignore any that start with `amqi`.
2. Change into the directory containing the maintenance packages.
3. Use the `ls` command to list the available updates. For example, if there are level 1 service updates for the Runtime, SDK and Server packages, you will see the following:
   ```
   MQSeriesRuntime-Uxxxx-6.0.0-1.i386.rpm
   MQSeriesSDK-Uxxxx-6.0.0-1.i386.rpm
   MQSeriesServer-Uxxxx-6.0.0-1.i386.rpm
   ```
4. To find out which packages are installed on your machine, enter the following:
   ```
   rpm -q -a | grep MQSeries
   ```

For example, if you have a minimum WebSphere MQ installation and SDK component, at level 0, this will return:

```
MQSeriesRuntime-6.0.0-0
MQSeriesSDK-6.0.0-0
MQSeriesServer-6.0.0-0
```

5. Now install all available updates for the packages you have on your system:

```
rpm -ivh MQSeriesRuntime-Uxxxx-6.0.0-1.i386.rpm
  MQSeriesSDK-Uxxxx-6.0.0-1.i386.rpm
  MQSeriesServer-Uxxxx-6.0.0-1.i386.rpm
```

   Note: You must install all packages in a service update that apply to those packages that are currently installed on your system.

6. Repeat step 4, and you will see that the Runtime, SDK and Server packages are now at level 1:

```
MQSeriesRuntime-6.0.0-0
MQSeriesSDK-6.0.0-0
MQSeriesServer-6.0.0-0
MQSeriesRuntime-Uxxxx-6.0.0-1
MQSeriesSDK-Uxxxx-6.0.0-1
MQSeriesServer-Uxxxx-6.0.0-1
```

For further information on using RPM to install software packages, see your Linux documentation.

Restoring the previous service level

When service is installed, the original versions of replaced files are saved to allow the updates to be removed if necessary. To restore the previous service level, issue an RPM uninstall for all the packages that were updated by the maintenance package as follows:

1. Log in as root.
2. To find out which packages are installed on your machine, enter the following:

```
rpm -q -a | grep MQSeries
```

   Using the example given in "Applying service" on page 49, returns:

```
MQSeriesRuntime-6.0.0-0
MQSeriesSDK-6.0.0-0
MQSeriesServer-6.0.0-0
MQSeriesRuntime-Uxxxx-6.0.0-1
MQSeriesSDK-Uxxxx-6.0.0-1
MQSeriesServer-Uxxxx-6.0.0-1
```

3. Remove all the updates applied at level 1, with the following commands:

```
rpm -ev MQSeriesRuntime-Uxxxx-6.0.0-1 MQSeriesSDK-Uxxxx-6.0.0-1
  MQSeriesServer-Uxxxx-6.0.0-1
```

4. Repeat step 2 to check that the packages have been returned to their original levels:

```
MQSeriesRuntime-6.0.0-0
MQSeriesSDK-6.0.0-0
MQSeriesServer-6.0.0-0
```

For further information on using RPM to install software packages, see your Linux documentation.

Chapter 6. Uninstalling WebSphere MQ

This chapter tells you how to remove WebSphere MQ for Linux from your system.

1. Before starting to uninstall, end all WebSphere MQ activity.
   a. Log in as root.
   b. Use the **dspmq** command to display the state of all the queue managers on the system.
   c. Use the **endmqm** command to stop all running queue managers.
   d. Stop any listeners associated with the queue managers, using the command:
      ```
      endmqlsr -m QMgrName
      ```
   e. To check that you have stopped all of the necessary WebSphere MQ processes, type:
      ```
      ps -ef | grep mq
      ```
   f. Check that there are no processes listed that are running command lines beginning `amq` or `runmq`. Ignore any that start with `amqi`.
2. Before you uninstall WebSphere MQ for Linux you must find out the names of the packages (components) currently installed on your system. To list all the packages with their version information, enter the following:
   ```
   rpm -q -a | grep MQSeries
   rpm -q -a | grep gsk
   ```

   Note: If you have applied maintenance to the install, remove these maintenance packages before removing the base install.
3. Some of the installed packages are dependent on others. The **rpm** command will not remove a package if others are dependent on it. For this reason you must uninstall the packages in such an order that each one you uninstall has no dependencies from other packages. To list all of the packages on which a named package (for example `MQSeriesServer`) depends, enter the following:
   ```
   rpm -q --requires MQSeriesServer
   ```

   Alternatively, remove all the components at the same time by appending all the package names to the rpm command arguments. For example, to remove the runtime, Server and SDK components, enter the following:
   ```
   rpm -ev MQSeriesRuntime MQSeriesServer MQSeriesSDK
   ```
4. After uninstalling , you might want to delete the contents of `/var/mqm`, which is not removed by the uninstallation. **Only do this if you do not need your queue manager data.** For example, do not delete the content if you plan to migrate to a later version of WebSphere MQ.

You have now uninstalled WebSphere MQ.

Chapter 7. WebSphere MQ Documentation

This chapter describes the documentation and sources of information about WebSphere MQ. It starts with a list of the publications, including their PDF filenames, and then discusses:

- Publications supplied with the product
- Hardcopy books
- Online information

If there is similar information in this book and any of the books in the following list, the information in this book should take precedence.

WebSphere MQ is described in the following books:

Table 8. WebSphere MQ family books

PDF file name	Order Number	Title
AMQTAC05	GC34-6476	*WebSphere MQ For Windows Quick Beginnings*
AMQDAC07	GC34-6477	*WebSphere MQ For Solaris Quick Beginnings*
AMQAAC07	GC34-6478	*WebSphere MQ For AIX Quick Beginnings*
AMQCAC06	GC34-6479	*WebSphere MQ For HP-UX Quick Beginnings*
AMQ1AC04	GC34-6480	*WebSphere MQ For Linux Quick Beginnings*
AMQWAC03	GC34-6481	*WebSphere MQ For iSeries Quick Beginnings*
CSQZAE10	SC34-6587	*WebSphere MQ Intercommunication*
CSQZAH07	SC34-6589	*WebSphere MQ Queue Manager Clusters*
CSQZAF08	GC34-6590	*WebSphere MQ Clients*
AMQZAG06	SC34-6584	*WebSphere MQ System Administration Guide*
CSQZAJ10	SC34-6597	*WebSphere MQ Script (MQSC) Command Reference*
CSQZAX05	SC34-6593	*Monitoring WebSphere MQ*
CSQZAC04	SC34-6598	*WebSphere MQ Programmable Command Formats and Administration Interface*
AMQZAO05	GC34-6601	*WebSphere MQ Messages*
CSQZAL10	SC34-6595	*WebSphere MQ Application Programming Guide*
CSQZAK10	SC34-6596	*WebSphere MQ Application Programming Reference*
AMQZAN09	SC34-6592	*WebSphere MQ Using C++*
CSQZAW13	SC34-6591	*WebSphere MQ Using Java*
AMTYAK08	SC34-6065	*WebSphere MQ Application Messaging Interface*
CSQZAS02	SC34-6588	*WebSphere MQ Security*
CSQSAT03	GC34-6582	*WebSphere MQ for z/OS Concepts and Planning Guide*
CSQSAV04	SC34-6583	*WebSphere MQ for z/OS System Setup Guide*
CSQSAW03	SC34-6585	*WebSphere MQ for z/OS System Administration*
AMQWAG02	SC34-6586	*WebSphere MQ for iSeries System Administration Guide*
AMQTAN03	SC34-659	*WebSphere MQ for Windows Using the Component Object Model Interface*

Table 8. WebSphere MQ family books (continued)

PDF file name	Order Number	Title
AMQWAK02	SC34-6599	*WebSphere MQ for iSeries Application Programming Reference (ILE RPG)*
CSQSAQ03	GC34-6600	*WebSphere MQ for z/OS Problem Determination Guide*
CSQSAO04	GC34-6602	*WebSphere MQ for z/OS Messages and Codes*
CSQZAO00	GC34-6604	*WebSphere MQ Migration Guide*
CSQZAV00	GC34-6605	*WebSphere MQ Using .Net*
CSQSAD03	GI10-2584	*Program Directory for WebSphere MQ for z/OS*
AMQNAR10	SC34-6606	*WebSphere MQ Publish/Subscribe User's Guide*
CSQZAQ00	SC34-6607	*WebSphere MQ Constants*
CSQZAY03	SC34-6603	*WebSphere MQ Bibliography and Glossary*

Publications supplied with the product

The WebSphere MQ documentation is supplied separately on a CD-ROM alongside the product. You can either view the documents directly from CD, or you can install them on your computer (either before or after installing the WebSphere MQ product).

The WebSphere MQ online documentation is delivered in HTML, Microsoft® Compiled HTML Help (.CHM), and PDF formats on CD-ROM.

HTML

You can view the WebSphere MQ online documentation in HTML format directly from the documentation CD-ROM. All books are available in U.S. English and also in some or all of the following national languages:

- Brazilian Portuguese
- French
- German
- Italian
- Japanese
- Korean
- Spanish
- Simplified Chinese
- Traditional Chinese

When you read the books in HTML, you can follow hypertext links from one book to another. If you are reading translated books and link to a book that is not available in your national language, the U.S. English version of the book is opened instead.

PDF

A PDF (Portable Document Format), corresponding to each hardcopy book, is available on the documentation CD-ROM. You can read PDFs using Adobe Acrobat Reader. Additionally, you can download them to your own file system, or print them.

The PDFs are available in U.S. English in the *en_US* directory, and also in some or all of the following national languages. To find out which ones are available in your language, look for the appropriate directory on the CD-ROM. The PDFs are in a subdirectory called *ll_LL*, where *ll_LL* is one of the following:

- de_DE (German)
- es_ES (Spanish)
- fr_FR (French)
- it_IT (Italian)
- ja_JP (Japanese)
- ko_KR (Korean)
- pt_BR (Brazilian Portuguese)
- zh_CN (Simplified Chinese)
- zh_TW (Traditional Chinese)

Within these directories, you can find the complete set of PDFs that are available. "Hardcopy books" shows the file names used for the PDF files.

Hardcopy books

This book, and all the books listed in Table 8 on page 53, are available for you to order or print.

You can order publications from the IBMLink™ Web site at:

```
http://www.ibm.com/ibmlink
```

In the United States, you can also order publications by dialing **1-800-879-2755**.

In Canada, you can order publications by dialing **1-800-IBM-4YOU (1-800-426-4968).**

For further information about ordering publications, contact your IBM authorized dealer or marketing representative.

For information about printing books, see "Publications supplied with the product" on page 54.

Online information

This section describes the sources of information available online about WebSphere MQ Version 6.0:

HTML and PDF books on the World Wide Web

The WebSphere MQ books are available on the World Wide Web as well as on the product CD-ROM. They are available in PDF and HTML format. The WebSphere MQ product family Web site is at:

```
http://www.ibm.com/software/integration/mqfamily
```

By following links from this Web site you can:

- Obtain latest information about the WebSphere MQ product family.

- Access the WebSphere MQ books in HTML and PDF formats.

Online help

Man pages are provided for all API calls, MQSC commands, and relevant control commands including **crtmqm**, **strmqm**, and **endmqm**.

SupportPacs

SupportPacs contain material that complements the WebSphere MQ family products, for example, there are a number of SupportPacs to help you with performance and capacity planning. Many SupportPacs are freely available for download, others can be purchased as a fee-based service. SupportPacs can be obtained from the following Web site:

```
http://www.ibm.com/software/integration/websphere/support
```

WebSphere MQ newsgroups

WebSphere MQ support provides a number of newsgroups where members share their knowledge and experience with others. A list of the newsgroups can be found at:

```
http://www.ibm.com/software/integration/mqfamily/support/newsgroups
```

Whitepapers and migration documents

IBM produces a number whitepapers that contain other useful information about WebSphere MQ. These can be found at:

```
http://www.ibm.com/software/integration/websphere/library
```

Service support summary (PTF readmes)

The service support summary gives a summary of the support information and end of service dates for in-service MQSeries products. This can be found at:

```
http://www.ibm.com/software/integration/mqfamily/support/summary
```

Notices

This information was developed for products and services offered in the United States. IBM may not offer the products, services, or features discussed in this information 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 information. The furnishing of this information does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, 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 information. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this information 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 United Kingdom Laboratories,
Mail Point 151,
Hursley Park,
Winchester,
Hampshire,
England,
SO21 2JN.

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 Programming License Agreement, or any equivalent agreement between us.

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.

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. 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 IBM's application programming interfaces.

Trademarks

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

AIX®	AS/400®	BookManager®
CICS®	DB2®	DB2 Universal Database
FFST™	First Failure Support Technology™	IBM
IBMLink	Informix	iSeries
Lotus®	Lotus Notes®	MQSeries
OpenEdition®	OS/2®	OS/390®
pSeries	POWER	SupportPac
Tivoli®	TXSeries®	VisualAge®

WebSphere z/OS® zSeries

Lotus and Lotus Notes are registered trademarks of Lotus Development Corporation in the United States, or other countries, or both.

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of the Open Group in the United States and other countries.

Microsoft, Windows®, Windows NT®, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel is a registered trademark of Intel Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Index

W

Sending your comments to IBM

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM.

Feel free to comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness of this book.

Please limit your comments to the information in this book and the way in which the information is presented.

To make comments about the functions of IBM products or systems, talk to your IBM representative or to your IBM authorized remarketer.

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

You can send your comments to IBM in any of the following ways:

- By mail, to this address:

  User Technologies Department (MP095)
  IBM United Kingdom Laboratories
  Hursley Park
  WINCHESTER,
  Hampshire
  SO21 2JN
  United Kingdom
- By fax:
  - From outside the U.K., after your international access code use 44–1962–816151
  - From within the U.K., use 01962–816151
- Electronically, use the appropriate network ID:
  - IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
  - IBMLink: HURSLEY(IDRCF)
  - Internet: idrcf@hursley.ibm.com

Whichever method you use, ensure that you include:

- The publication title and order number
- The topic to which your comment applies
- Your name and address/telephone number/fax number/network ID.

GC34-6480-00