Installation Guide

Sun™ ONE Message Queue

Version 3.0.1

816-6453-10
October, 2002

Contents

List of Tables 5

List of Procedures 7

Preface 9

Audience for This Guide 9
Organization of This Guide 9
Conventions 10
Text Conventions 10
Directory Variable Conventions 11
Other Documentation Resources 13
The MQ Documentation Set 13
Example Client Applications 13

Chapter 1 Introduction 15
Product Editions 15
Platform Edition 15
Enterprise Edition 16
Supported Platforms and Products 17
MQ Software Modules 18
Installing from Web and CD-ROM 19
Installed Directory Structure 19
Upgrading from Version 2.0 21
Uninstalling iMQ 2.0 21
Compatibilities and Incompatibilities 21
Broker Compatibility 22
Administered Object Compatibility 23
Administration Tool Compatibility 23
Client Compatibility 24
Where To Go Next 24

Chapter 2 Solaris Installation **25**
Hardware and Software Requirements **25**
Installing MQ on Solaris **27**
Installing from the Web 27
Installing from CD-ROM 32
Configuring MQ for Automatic Startup **33**
Configuring the Java Runtime for MQ **34**
Upgrading Editions **35**
Where To Go Next **36**
Uninstalling MQ on Solaris **37**

Chapter 3 Linux Installation **39**
Hardware and Software Requirements **39**
Installing MQ on Linux **40**
Installing from the Web 41
Installing from CD-ROM 42
Configuring the Java Runtime for MQ **43**
Upgrading Editions **43**
Where To Go Next **44**
Uninstalling MQ on Linux **45**

Chapter 4 Windows Installation **47**
Hardware and Software Requirements **47**
Installing MQ on Windows **48**
Installation Defaults **50**
Troubleshooting Installation Problems **50**
Upgrading Editions **51**
Where To Go Next **52**
Uninstalling MQ on Windows **52**

List of Tables

Table 1 Book Contents 9
Table 2 Document Conventions 10
Table 3 MQ Directory Variables 11
Table 4 MQ Documentation Set 13
Table 1-1 MQ 3.0.1 Product Support Matrix 17
Table 1-2 MQ Software Modules 18
Table 1-3 Installed Directory Structure 19
Table 1-4 Compatibility of MQ 3.0.1 with iMQ 2.0 Data 22
Table 2-1 Hardware and Software Requirements for Solaris 25
Table 2-2 Packages in Solaris Bundle 29
Table 2-3 Packages Required for Various Scenarios 30
Table 2-4 Broker Startup Configuration Properties 34
Table 3-1 Hardware and Software Requirements for Linux 39
Table 4-1 Hardware and Software Requirements for Windows 47
Table 4-2 Windows Installation Defaults 50

List of Procedures

To install MQ on Solaris from the Web 27
To install MQ on Solaris from CD-ROM 32
To upgrade to Enterprise Edition on Solaris 35
To remove MQ on Solaris 37
To install MQ on Linux from the Web 41
To install MQ on Linux from CD-ROM 42
To upgrade to Enterprise Edition on Linux 43
To remove MQ on Linux 45
To install MQ on Windows 48
To troubleshoot installation problems on Windows 50
To upgrade to Enterprise Edition on Windows 51
To remove MQ on Windows 52

Preface

This book explains how to install Sun™ ONE Message Queue (MQ) 3.0.1, and this preface contains the following sections:

- Audience for This Guide
- Organization of This Guide
- Conventions
- Other Documentation Resources

Audience for This Guide

The audience for this guide is MQ developers and administrators.

Organization of This Guide

All users should read the introduction and then the appropriate platform-specific chapter. The following table briefly describes the contents of each chapter:

Table 1 Book Contents

Chapter	Description
Chapter 1, "Introduction"	Explains the MQ editions, supported platforms and products, the MQ software modules, and the installed directory structure.
Chapter 2, "Solaris Installation"	Explains the hardware and software requirements, defaults, and installation procedure for Solaris, as well as how to uninstall the product on this platform.

Table 1 Book Contents *(Continued)*

Chapter	Description
Chapter 3, "Linux Installation"	Explains the hardware and software requirements, defaults, and installation procedure for Linux, as well as how to uninstall the product on this platform.
Chapter 4, "Windows Installation"	Explains the hardware and software requirements, defaults, and installation procedure for Windows (2000 and XP), as well as how to uninstall the product on this platform.

Conventions

This section provides information about the conventions used in this document.

Text Conventions

Table 2 Document Conventions

Format	Description
italics	Italicized text represents a placeholder. Substitute an appropriate clause or value where you see italic text. Italicized text is also used to designate a document title, for emphasis, or for a word or phrase being introduced.
`monospace`	Monospace text represents example code, commands that you enter on the command line, directory, file, or path names, error message text, class names, method names (including all elements in the signature), package names, reserved words, and URL's.
`[]`	Square brackets to indicate optional values in a command line syntax statement.
ALL CAPS	Text in all capitals represents file system types (GIF, TXT, HTML and so forth), environment variables (IMQ_HOME), or acronyms (MQ, JSP).
Key+Key	Simultaneous keystrokes are joined with a plus sign: Ctrl+A means press both keys simultaneously.
Key-Key	Consecutive keystrokes are joined with a hyphen: Esc-S means press the Esc key, release it, then press the S key.

Directory Variable Conventions

MQ makes use of three directory variables; how they are set varies from platform to platform. Table 3 describes these variables and summarizes how they are used on the Solaris, Windows, and Linux platforms.

Table 3 MQ Directory Variables

Variable	Description
IMQ_HOME	This is generally used in MQ documentation to refer to the root MQ installation directory: • On Solaris, there is no root MQ installation directory. Therefore, IMQ_HOME is not used in MQ documentation to refer to file locations on Solaris. • On Solaris, for Sun ONE Application Server, Evaluation Edition, the root MQ installation directory is: *root Application Server installation directory*/imq. • On Windows, the root MQ installation directory is set by the MQ installer (by default, as C:\Program Files\Sun Microsystems\Message Queue 3.0). • On Windows, for Sun ONE Application Server, the root MQ installation directory is: *root Application Server installation directory*/imq. • On Linux, the root MQ installation directory is, by default: /opt/imq.
IMQ_VARHOME	This is the /var directory in which MQ temporary or dynamically-created configuration and data files are stored. It can be set as an environment variable to point to any directory. • On Solaris, IMQ_VARHOME defaults to the /var/imq directory. • On Solaris, for Sun ONE Application Server, Evaluation Edition, IMQ_VARHOME defaults to IMQ_HOME/var. • On Windows IMQ_VARHOME defaults to IMQ_HOME/var. • On Windows, for Sun ONE Application Server, IMQ_VARHOME defaults to IMQ_HOME/var. • On Linux, IMQ_VARHOME defaults to IMQ_HOME/var.

Table 3 MQ Directory Variables *(Continued)*

Variable	Description
IMQ_JAVAHOME	This is an environment variable that points to the location of the Java runtime (JRE) required by MQ executables: • On Solaris, IMQ_JAVAHOME defaults to the /usr/j2se/jre directory, but a user can optionally set the value to wherever the required JRE resides. • On Windows, IMQ_JAVAHOME defaults to IMQ_HOME/jre, but a user can optionally set the value to wherever the required JRE resides. • On Linux, MQ first looks for the java runtime in the /usr/java/j2sdk*Version* directory, and then looks in the /usr/java/j2re*Version* directory, but a user can optionally set the value of IMQ_JAVAHOME to wherever the required JRE resides.

In this guide, IMQ_HOME, IMQ_VARHOME, and IMQ_JAVAHOME are shown *without* platform-specific environment variable notation or syntax (for example, $IMQ_HOME on UNIX). All path names use UNIX file separator notation (/).

Other Documentation Resources

In addition to this guide, MQ provides additional documentation resources.

The MQ Documentation Set

The documents that comprise the MQ documentation set are listed in Table 4 in the order in which you would normally use them.

Table 4 MQ Documentation Set

Document	Audience	Description
MQ Installation Guide	Developers and administrators	Explains how to install MQ software on Solaris, Linux, and Windows platforms.
MQ Release Notes	Developers and administrators	Includes descriptions of new features, limitations, and known bugs, as well as technical notes.
MQ Developer's Guide	Developers	Provides a quick-start tutorial and programming information relevant to the MQ implementation of JMS.
MQ Administrator's Guide	Administrators, also recommended for developers	Provides background and information needed to perform administration tasks using MQ administration tools.

Example Client Applications

A number of example applications that provide sample client application code are included in the following location:

`IMQ_HOME/demo` (`/usr/demo/imq` on Solaris)

See the README file located in that directory and in each of its subdirectories.

Chapter 1

Introduction

This chapter provides an overall introduction to installing the MQ product. The topics covered are:

- Product Editions
- Supported Platforms and Products
- MQ Software Modules
- Installing from Web and CD-ROM
- Installed Directory Structure
- Upgrading from Version 2.0
- Where To Go Next

Product Editions

The Sun ONE Message Queue product is available in two editions: Platform and Enterprise—each corresponding to a different licensed capacity, as described below. (To upgrade MQ from one edition to another, see the instructions in the MQ *Installation Guide*.)

Platform Edition

This edition can be downloaded free from the Sun website and is also bundled with the latest Sun ONE Application Server platform. The Platform Edition places no limit on the number of JMS client connections supported by each MQ message service. It comes with two licenses, as described below:

- a basic license. This license provides basic JMS support (it's a full JMS provider), but does not include such enterprise features as load balancing (multi-broker message service), HTTP/HTTPS connections, secure connection services, scalable connection capability, and multiple queue delivery policies. The license has an unlimited duration, and can therefore be used in less demanding production environments.
- a 90-day trial enterprise license. This license includes all enterprise features (such as support for multi-broker message services, HTTP/HTTPS connections, secure connection services, scalable connection capability, and multiple queue delivery policies) not included in the basic license. However, the license has a limited 90-day duration enforced by the software, making it suitable for evaluating the enterprise features available in the Enterprise Edition of the product (see "Enterprise Edition" on page 16).

NOTE The 90-day trial license can be enabled by starting the MQ message service—an MQ broker instance—with the `-license` command line option (see the MQ *Administrator's Guide*) and passing "`try`" as the license to use:

```
imqbrokerd -license try
```

You must use this option each time you start the broker instance, otherwise it defaults back to the basic Platform Edition license.

Enterprise Edition

This edition is for deploying and running messaging applications in a production environment. It includes support for multi-broker message services, HTTP/HTTPS connections, secure connection services, scalable connection capability, and multiple queue delivery policies. You can also use the Enterprise Edition for developing, debugging, and load testing messaging applications and components. The Enterprise Edition has an unlimited duration license that places no limit on the number of brokers in a multi-broker message service, but specifies the number of CPU's that are supported.

Supported Platforms and Products

MQ 3.0.1 is supported on Solaris, Linux, and Windows operating systems and platforms. It also depends upon other technologies, as indicated in the following table. Other versions or vendor implementations can also be used but they are untested by Sun Microsystems and therefore not supported.

Table 1-1 MQ 3.0.1 Product Support Matrix

Platform/Product	Used For	Supported Platform/Product Version[1]
Java Runtime Environment (JRE) (Sun Microsystems production versions only)	MQ broker (message server) and MQ administration tools	JDK/JRE 1.4.0_02: • Solaris 9—MQ depends on platform-bundled JDK/JRE 1.4.0_02 • Solaris 8—platform-bundled JRE 1.3 needs to be removed and replaced by JDK/JRE 1.4.0_02 • Windows—JRE 1.4.0_02 is bundled and installed with MQ JDK/JRE 1.4.1: • Linux— JRE 1.4.1 is bundled on MQ CD distribution, but not yet with Linux platform
Java Software Development Kit (JDK), Standard Edition (Sun Microsystems production versions only)	JMS client development (SOAP messaging clients are supported only on JDK 1.4._02)	Version 1.4.0_02[2]: • (SPARC only) Solaris 8 and Solaris 9 • Windows XP professional, 2000 Professional SP2, 2000 Server SP2, 2000 Advanced Server SP2 • Linux RedHat 7.2 Version 1.3.1_05[3]: • (SPARC only) Solaris 8 and Solaris 9 • Windows XP professional, 2000 Professional SP2, 2000 Server SP2, 2000 Advanced Server SP2 Version 1.2.2_08: Not supported, but should work (in case you cannot upgrade to a later version)
LDAP Directory Server	MQ user repository and administered object support	Sun ONE Directory Server version 5.1
Web Server	HTTP and HTTPS support	Sun ONE Web Server, Enterprise Edition 6.0 SP4

Table 1-1 MQ 3.0.1 Product Support Matrix *(Continued)*

Platform/Product	Used For	Supported Platform/Product Version[1]
Database	Plugged-in persistence support	Cloudscape (version 3.0) Oracle 8i, version 8.1.7 and Oracle 9i, version 9.0.1
JNDI	administered object support	• JNDI version 1.2.1 • LDAP Service Provider version 1.2.2 • File System Service Provider version 1.2 Beta 3 (supported for development and testing, but not for deployment in a production environment.)

1. Check the MQ *Release Notes* for any updates to supported versions
2. Download this JDK from: http://java.sun.com/j2se/1.4/index.html
3. Download this JDK from: http://java.sun.com/j2se/1.3/index.html

MQ Software Modules

The following table identifies the full set of software modules included with the MQ product (see Table 1-2 on page 18 for their installed location).

Table 1-2 MQ Software Modules

Module	Contents
Broker	The server-side software used to route and deliver messages. This module requires the Java runtime module.
Administration tools	The command-line utilities and GUI tools used to administer an MQ messaging system. This module requires the client runtime and Java runtime modules.
Client runtime	The client-side software needed to support client applications.
Documentation	API documentation needed by client application developers in JavaDoc format.
Example applications	Example client applications.
Java runtime	Java Runtime Environment (Windows only).
License	The software needed to enable the licensed capacity of an MQ messaging system.

Installing from Web and CD-ROM

You have the option of either downloading the MQ 3.0.1 product from the Sun ONE website or installing it from CD-ROM. For detailed instructions, see the platform-specific instructions in subsequent chapters.

Installed Directory Structure

The install image below reflects a full Solaris installation (all packages) or a full ("Typical") Windows installation. This image might vary if you perform a partial installation.

NOTE On Windows, the `COPYRIGHT`, `LICENSE`, and `README` files have a .txt filename extension.

Table 1-3 Installed Directory Structure

File and directory (Solaris)	File and directory (Windows and Linux)[1]	Contents
`COPYRIGHT` (not installed)	`./COPYRIGHT`	Copyright text file
`LICENSE` (not installed)	`./LICENSE`	License text file
`README` (not installed)	`./README`	README text file
`/usr/bin` directory	`./bin` directory	Contains the executables for the broker (`imqbrokerd`) and the following MQ administration tools: • Administration Console (`imqadmin`) • Command (`imqcmd`) • Object Manager (`imqobjmgr`) • User Manager (`imqusermgr`) • Database Manager (`imqdbmgr`) • Key Tool (`imqkeytool`) On Windows, the files named above have a .bat filename extension. This directory also includes the utility to install and uninstall the broker as a Windows Service (`imqsvcadmin`) as well other executables (`imqbrokersvc`).

Table 1-3 Installed Directory Structure *(Continued)*

File and directory (Solaris)	File and directory (Windows and Linux)[1]	Contents
`/usr/share/lib` directory	`./lib` directory	Contains files that support the MQ client runtime: `/*jar` contains jar files used to build and run JMS client applications
`/usr/share/lib/imq` directory	`./lib` directory	Contains files used to support MQ tools and processes: `/ext/*jar` location for placing jar files needed for plug-in persistence capabilitiy `/props` subdirectory contains the broker's default configuration file `/help` subdirectory contains MQ help files `/images`
`/etc/imq` directory	`./etc` directory	Contains license files, security-related files (such as passfile, access control files, and flat-file user repository), and (on Solaris only) rc script configuration files that can be used for automatic startup
`/var/imq` directory	`./var` directory	Working storage directory for MQ. `/instances` subdirectory which will contain configuration files, log files, and file-based persistent data stores for each broker instance
`/usr/share/javadoc/imq` directory	`./javadoc` directory	Contains the MQ and JMS API documentation distributed as Javadoc (HTML)
`/usr/demo/imq` directory	`./demo` directory	Source code for and instructions on how to run client example applications
	`./jre` directory	The JRE 1.4 files (on Windows only)

1. Paths are relative to `IMQ_HOME` (see "Directory Variable Conventions" on page 11).

Upgrading from Version 2.0

MQ 3.0.1 is fully compatible with MQ 3.0, and upgrading from MQ 3.0 to MQ 3.0.1 requires no changes in broker configuration, administered objects, administration tools, or client applications.

However, MQ 3.0.1 is generally *not* compatible with iMQ 2.0, largely because of changes in internal and external data used by MQ 3.0.1. For this reason it is strongly recommended that you un-install iMQ 2.0 before installing MQ 3.0.1, and *not* try to install MQ 3.0.1 over iMQ 2.0.

Uninstalling iMQ 2.0

If you are running iMQ 2.0, Service Pack 1, you should first uninstall the Service Pack, using the uninstall instructions in the *Service Pack Installation Guide*, and then uninstall iMQ 2.0, using the uninstall instructions in the iMQ 2.0 *Installation Guide*.

The uninstall operation does not remove the iMQ 2.0 `IMQ_VARHOME` directory. This directory (by default `/var/opt/SUNWjmq` on Solaris and Linux operating systems, and c:\Program files\iPlanetMessageQueue2.0\var on Windows systems) contains transient and security-related files (see Table 1-4). Some of this data is compatible with MQ 3.0.1 and can be preserved using the instructions in the following section.

Compatibilities and Incompatibilities

Due to changes made to improve features, MQ 3.0.1 is generally not compatible with iMQ 2.0. In particular, there are a number of issues that you might need to address when upgrading from iMQ 2.0 to MQ 3.0.1:

- Broker Compatibility
- Administered Object Compatibility
- Administration Tool Compatibility
- Client Compatibility

Broker Compatibility

An MQ 3.0.1 broker will not inter-operate with an iMQ 2.0 broker due to changes in broker properties and the persistent store schema. However, some iMQ 2.0 data is compatible with MQ 3.0.1, as shown in Table 1-4 on page 22, and can be preserved when upgrading to MQ 3.0.1. When upgrading from iMQ 2.0 to MQ 3.0.1, you should consider the following:

- You can copy iMQ 2.0 `config.properties` files to another location and, in most cases, consult the property settings they contain when you configure MQ 3.0.1 brokers.
- Any persistent iMQ 2.0 data—messages, destinations, durable subscriptions—cannot be re-used. In particular, you will need to re-create iMQ 2.0 destinations in your MQ 3.0.1 brokers.
- You can continue to use iMQ 2.0 user repository and access control properties files after installing MQ 3.0.1. The MQ 3.0.1 installer does not overwrite these files. you will have to move them to the appropriate MQ 3.0 location (see the MQ *Administrator's Guide*, Appendix C).

Table 1-4 Compatibility of MQ 3.0.1 with iMQ 2.0 Data

iMQ 2.0 Data Category	Location of iMQ 2.0 Data	Compatibility with MQ 3.0.1
Broker properties	`IMQ_VARHOME/stores/`*brokerName*`/props/config.properties`	Incompatible; do not use.
Persistent store (messages, destinations, durable subscriptions)	`IMQ_VARHOME/stores/`*brokerName*`/filestore/` or JDBC-accessible data store	Incompatible; do not use.
Administered objects	local directory or LDAP server	Compatible; can use and/or convert to 3.0.1.
Security: user repositories	`IMQ_VARHOME/security/passwd` or LDAP server	Compatible. Move to following location: `IMQ_HOME/etc/passwd` (`/etc/imq/passwd` on Solaris)
Security: access control file	`IMQ_VARHOME/security/accesscontrol.properties`	Compatible. Move to following location: `IMQ_HOME/etc/...` (`/etc/imq/...` on Solaris)

Administered Object Compatibility

MQ 3.0.1 administered objects have been enhanced with new attributes and iMQ 2.0 attributes have been renamed. Therefore, when upgrading from iMQ 2.0 to MQ 3.0.1, you should consider the following:

- You can use the same object store and administered objects that you created in iMQ 2.0; however, it is best to upgrade your administered objects after installing MQ 3.0.1. The Administration Console (`imqadmin`) and the ObjectManager command line utility (`imqobjmgr`), when performing an update operation, will convert iMQ 2.0 administered objects into MQ 3.0.1 administered objects.
- The MQ 3.0.1 client runtime will look up and instantiate iMQ 2.0 administered objects by converting them into local MQ 3.0.1 administered objects, but this will *not* convert iMQ 2.0 administered objects in the object store into MQ 3.0.1 administered objects.
- JMS clients (applications and/or components) that directly instantiate administered objects—that is, that are JMS provider-dependent—need to be rewritten to accommodate new administered object attribute names (see Chapter 4 and Appendix A of the MQ *Developer's Guide* for information on administered object attributes).
- Scripts that start JMS clients and which set administered object attribute values using command line options need to be rewritten to accommodate the new administered object attribute names (see Chapter 4 and Appendix A of the MQ *Developer's Guide* for information on administered object attributes).

Administration Tool Compatibility

Because of the renaming of many files and directories (specifically to replace the string "jmq" with "imq"), all MQ 3.0.1 command line utilities, broker properties, administered object attributes, and internal file names have changed. Therefore, when upgrading from iMQ 2.0 to MQ 3.0.1, you should consider the following:

- Any scripts that use command line utilities (`imqbrokerd`, `imqcmd`, `imqobjmgr`, and so forth) need to be edited to replace the old commands with the newly-named commands. Note, especially, that the `jmqbroker` command is now `imqbrokerd`.
- The Administration Console (`imqadmin`) allows you to manage several brokers and/or object stores concurrently, and saves the list of managed entities that are displayed in the navigational pane on the left side of the screen. Thus each time you launch the Console, the list of managed entities is redisplayed. The name of the directory in which user settings for the iMQ 2.0 Administration Console were stored has changed for MQ 3.0.1. If you wish to preserve the old

Console settings when upgrading from iMQ 2.0 to MQ 3.0.1, you need to change the name of the directory where the `brokerlist.properties` and `objstorelist.properties` files are stored from *user.home*`/.jmq/admin` to *user.home*`/.imq/admin`, where *user.home* is a java system property.

Client Compatibility

When upgrading from iMQ 2.0 to MQ 3.0.1, you should consider the following:

- An MQ 3.0.1 broker will support the iMQ 2.0 client runtime (but without additional MQ 3.0.1 capabilities), but an iMQ 2.0 broker will *not* support the MQ 3.0.1 client runtime.
- JMS clients built on JDK 1.2, 1.3, or 1.4 can inter-operate with a broker running JRE 1.4. However, clients that use a secure (SSL-based) connection to a broker will require additional JSSE and JNDI libraries if they are not built on JDK 1.4 (which includes these libraries).

Where To Go Next

When you are ready to install MQ on a specific platform, see the appropriate chapter for your platform (Solaris, Linux, or Windows). Each chapter contains hardware and software requirements, installation procedures, and other relevant instructions, such as how to upgrade editions and how to proceed after installation.

Chapter 2

Solaris Installation

This chapter explains the following topics as they apply to a Solaris installation:

- Hardware and Software Requirements
- Installing MQ on Solaris
- Configuring MQ for Automatic Startup
- Configuring the Java Runtime for MQ
- Upgrading Editions
- Where To Go Next
- Uninstalling MQ on Solaris

Hardware and Software Requirements

At a minimum, your Solaris™ development system (SPARC™ Platform Edition) should satisfy the minimum requirements indicated in the following table.

Table 2-1 Hardware and Software Requirements for Solaris

Component	Requirements
Operating system	Solaris 8 or Solaris 9 (SPARC platforms) **Note:** To ensure proper operation of MQ, you should install all required Solaris Patches for Java 2. For the latest information about the patches and to download the recommended and required patches, see: `http://java.sun.com/j2se/1.4/install-solaris-patches.html`

Table 2-1 Hardware and Software Requirements for Solaris *(Continued)*

Component	Requirements
CPU	Sun Ultra™ 1 (or compatible) workstation that is TCP/IP networked
RAM	128 Mbytes
Hard drive space	The compressed installation file is approximately 6 Mbytes. The temporary working directory used for extracting the installation files requires an additional 8 Mbytes. The installed product requires approximately 8 Mbytes of hard drive space. MQ, however, may need more space if the broker stores persistent messages locally.
Java 2 Standard Edition (J2SE)	See Table 1-1 on page 17 for the supported versions of the Java Runtime Environment (JRE) and Java Software Development Kit (JDK) that are supported on Solaris. The MQ software distribution CD includes the required JRE version at the time of release.

Installing MQ on Solaris

The MQ product can be downloaded from the Sun ONE website or installed from the product CD-ROM. See the appropriate section below for details.

NOTE If you are upgrading from MQ 3.0, it is recommended that you first uninstall MQ 3.0 software, as described in the MQ 3.0 *Installation Guide*, before installing MQ 3.0.1.

NOTE Because MQ is installed with other products (such as Solaris 9 Update 2, Sun ONE Application Server 7.0, and possibly others), you might wish to check whether MQ has already been installed on your system. To do so, enter the following command:

```
pkginfo | grep SUNWiq
```

If MQ packages are already installed, you can check the version by entering:

`pkginfo -l` *packageName*

where *packageName* is any of the MQ packages.

Installing from the Web

The following instructions explain how to download and install the MQ product on Solaris from the Sun ONE website.

➤ **To install MQ on Solaris from the Web**

1. Download the MQ product from the website into an empty, temporary working directory.
2. Run the command script:

   `sh imq3_0_1-`*edition*`-solsparc.sh`

   where *edition* takes one of the following values: `plt` or `ent`, depending on whether you are installing the Platform or Enterprise Edition, respectively.

   The command displays the first page of the license for the product.
3. Read the product license. Installation and use of the product is subject to acceptance of the license agreement.

4. To display the entire license, one page at a time, repeatedly press the space bar. When you reach the end of the license, the program prompts you to accept the license.
   - If you choose not to accept the license agreement, type **no** or **n** and the installation terminates.
   - If you choose to accept the license agreement, type **yes** or **y** and the installation continues. The following files are extracted:
     - `README`
     - `imq3_0_1-`*edition*`-solsparc.tar.Z`
     - `COPYRIGHT`
     - `LICENSE` (a copy of the license agreement)
5. Extract the archived files:

   `/bin/zcat imq3_0_1-`*edition*`-solsparc.tar.Z | tar xvfp -`

   A new directory, `imq3_0_1-pkgs`, is created.

   NOTE To avoid possible problems, do *not* use the GNU `tar` utility when installing MQ.

6. Change directories:

   `cd imq3_0_1-pkgs`
7. Become root:

   `su root`
8. Determine which, if any, of the shared packages included with MQ are already installed on your system.

   To see a list of such packages, type:

   `pkginfo SUNWaclg SUNWjaf SUNWjhrt SUNWjmail SUNWxsrt`

   The output shows the packages already installed and those which cannot be found.

9. Run the `pkgadd` command to install the packages:

```
pkgadd -d ./
```

The `pkgadd` utility lists the names of all packages in the directory available for installation (see Table 2-2). When prompted, indicate the packages you want to install. (Do not install any shared packages found in Step 8.)

Table 2-2 Packages in Solaris Bundle

#	Package	Description	Note
1	`SUNWaclg`	Apache Commons Logging Framework: API and runtime	Required for SOAP/JAXM client support.
2	`SUNWiqdoc`	MQ client API javadoc and example applications	Needed only for client development.
3	`SUNWiqfs`	MQ JNDI File System Service Provider	Required only for client development and administration tools that use the JNDI FIle System Service Provider. The JNDI Service Provider is not supported for deployment.
4	`SUNWiqjx`	MQ Java API for XML Messaging (JAXM): API and runtime	Required for SOAP/JAXM client support.
5	`SUNWiqlpl` or `SUNWiqlen`	MQ license files for Platform or Enterprise Edition message server	Depends on MQ edition.
6	`SUNWiqr`	MQ message server root package	Files needed for MQ executables.
7	`SUNWiqsup`	JNDI and JSSE jar files	Needed for client development and deployment with JDK 1.2 and 1.3.
8	`SUNWiqu`	MQ message server and administration tools.	
9	`SUNWiquc`	MQ JMS API and client runtime.	Required for JMS client support.
10	`SUNWiqum`	MQ JMS/SOAP Message Transformer API and runtime	Required to perform conversions between SOAP messages and JMS messages.

Table 2-2 Packages in Solaris Bundle *(Continued)*

#	Package	Description	Note
11	SUNWjaf	JavaBeans Activation Framework: API and runtime.	Required for SOAP/JAXM client support.
12	SUNWjhrt	JavaHelp API and runtime	Required if installing on Solaris 8. (Solaris 9 and above already have this package installed.) Will only install if a JVM 1.4 or greater has first been installed.
13	SUNWjmail	JavaMail: API and runtime	Required for SOAP/JAXM client support.
14	SUNWxsrt	SOAP with Attachments API for Java: API and runtime	Required for SOAP/JAXM client support.

The `pkgadd` utility installs the packages you specified, perhaps asking for additional information, and eventually returns to the original prompt, displaying the list of packages available for installation.

Table 2-3 provides a guide to the packages you need for different use scenarios:

Table 2-3 Packages Required for Various Scenarios

Scenario	Packages Needed	Notes
MQ message server and administration tools	SUNWiqr SUNWiqu SUNWiqlpl or SUNWiqlen SUNWiquc SUNWjhrt (optional) SUNWiqfs (optional)	Required for a MQ message server to run on a host.

Table 2-3 Packages Required for Various Scenarios *(Continued)*

Scenario	Packages Needed	Notes
Developing and/or deploying JMS clients	`SUNWiquc` `SUNWiqdoc` (optional) `SUNWiqsup` (optional)	Can be installed on a system without a MQ message server.

Developing and/or deploying SOAP/JAXM clients	`SUNWaclg`
	`SUNWjaf`
	`SUNWjmail`
	`SUNWiqjx`
	`SUNWxsrt`
	`SUNWiqdoc` (optional)
		Can be installed on a system without a MQ message server.
		Note: SOAP clients require JDK1.4

Developing and/or deploying clients using the JMS/SOAP Message Transformer	`SUNWiqum` Plus all packages needed to support JMS and SOAP/JAXM clients	Can be installed on a system without a MQ message server. The MQ Message Transformer API depends on both the JMS and SOAP APIs.

10. Type **q** to quit.
11. Exit the root shell.
12. Back up the `imq3_0_1-`*edition*`-solsparc.sh` file from your temporary working directory.

    This is your logical media. Treat this file as you would any other installation media. Place a copy in a safe location in case you encounter a situation (such as a system failure) that requires reinstallation of the product.

13. Clean up all remaining files in your temporary working directory.

NOTE Once installation is complete, to run the default broker instance (named `imqbroker`), you must be root—or, as root, change the privileges on the `/var/imq/instnces/imqbroker` directory (where configuration and persistent data are stored). However, if you run a non-default broker instance (using the `-name` *brokerName* option) then you automatically have privileges to the `/var/imq/instnces/`*brokerName* directory.

Installing from CD-ROM

The following instructions explain how to install the MQ product on Solaris from CD-ROM.

NOTE A compressed installation file (`tar.Z`) is provided on the CD if you do not want to install MQ directly from the packages on the CD.

➤ **To install MQ on Solaris from CD-ROM**

1. Log in as root or change to superuser.

For example, type the following at a command prompt:

```
su root
```

Then type your superuser password.

2. Insert the MQ CD into your CD-ROM drive.

If the Volume Manager™ software is running on your machine, the CD-ROM is automatically mounted to the `/cdrom/messagequeue3_0_1` directory.

If the Volume Manager is not running on your machine:

- Create a directory called `/cdrom/messagequeue3_0_1` by typing:

  ```
  mkdir -p /cdrom/messagequeue3_0_1
  ```

- Mount the CD-ROM manually:

```
mount -rF hsfs cdrom-device /cdrom/messagequeue3_0_1
```

An example of *cdrom-device* is `/dev/dsk/c0t0d0s0`.

NOTE Volume Manager is a tool provided on Solaris that allows you to perform administrative tasks, such as mounting CD-ROMs, more easily. Volume Manager mounts a CD-ROM as `/cdrom/`*name_of_media*, where *name_of_media* is determined from the CD-ROM itself.

Open and read (using your preferred text editor) the `LICENSE` file located in the `solaris/` directory of the CD.

- If you choose NOT to accept the license agreement, discontinue installation and contact the place where you purchased the product to determine the return policy.
- If you choose to accept the agreement, continue with the installation steps below.

3. Change to the directory on the CD containing the installation packages. For example, type:

   ```
   cd /cdrom/messagequeue3_0_1/solaris/imq3_0_1-pkgs
   ```

   The contents of this directory vary depending on the product edition.

4. Run the `pkgadd` command to install the packages:

   ```
   pkgadd -d ./
   ```

   The `pkgadd` utility lists the names of all packages in the directory available for installation. When prompted, indicate the packages you want to install (see Table 2-2 on page 29).

5. When the `pkgadd` prompt returns, type **`q`** to quit.

6. Exit the root shell.

Configuring MQ for Automatic Startup

If you wish to set the broker (the MQ message server) for automatic startup, you need to become root and edit the following configuration file:

```
/etc/imq/imqbrokerd.conf
```

The startup properties you can set in this configuration file are shown in Table 2-4:

Table 2-4 Broker Startup Configuration Properties

Property Name	Description
AUTOSTART	Specifies (YES/NO) if the broker is automatically started at boot time. Default: NO
ARGS	Specifies command line options and arguments to pass to the broker startup command. See the MQ *Administrator's Guide* for a listing and description of `imqbrokerd` command line options. (For example `-name` *brokerName*)
RESTART	Specifies (YES/NO) if the broker is automatically re-started if it abnormally exits. Default: YES

To check that startup changes are correct (without booting the system), you can, as root, explicitly run the MQ initialization script (S52imq) in "debug" mode:

```
env DEBUG=1 /etc/rc3.d/S52imq start
```

Configuring the Java Runtime for MQ

At startup time, a broker (the MQ message server) checks to make sure it has access to the required Java runtime version (JDK/JRE 1.4). There are a number of ways you can configure or set the JRE used by the broker. These are shown in the following list, in order of precedence:

1. Pass in the JDK or JRE using either the `imqbrokerd -javahome` or `-jrehome` command line options, respectively (if both are passed in, the last one on the command line will take precedence).
2. Set the JDK or JRE in the `IMQ_JAVAHOME` environment variable.
3. Let the broker use the installed JDK.

   This is the JDK located in `/usr/j2se`

To figure out why a broker is picking up a specific JDK/JRE, you can start the broker with the following command:

```
imqbrokerd -verbose
```

Upgrading Editions

MQ comes in two editions, as explained in "Product Editions" on page 15.

To upgrade from the Platform Edition to the Enterprise Edition, you need to install the Enterprise Edition license. This installation does not overwrite the MQ modules already installed and does not modify the configuration of your MQ messaging system.

To install the Enterprise Edition license you need only the `SUNWiqlen` package contained in the Enterprise Edition. The `SUNWiqlen` package is placed automatically in the `imq3_0_1-pkgs` directory (Web installation) or in the `solaris/imq3_0_1-pkgs` directory (CD-ROM installation) as a result of running the installation command script, uncompressing the file archive, and extracting the archived files.

➤ **To upgrade to Enterprise Edition on Solaris**

1. Stop any running brokers.

   `imqcmd shutdown bkr -u` *name* `-p` *password* `[-b` *hostName:port*`]`

2. Follow the installation procedure in "Installing from the Web" on page 27, Step 1 through Step 6 or in "Installing from CD-ROM" on page 32, Step 1 through Step 3.

3. When the installation is finished, add the `SUNWjqent` package:

   ```
   pkgadd -d . SUNWiqlen
   ```

4. Verify that the Enterprise Edition license is available by running the following command:

   ```
   imqbrokerd -license
   ```

Where To Go Next

Read the `README` and MQ *Release Notes* files.

- The `README` includes information on where to find documentation, news and updates, and how to send feedback.
- The MQ *Release Notes* contain information on code and documentation changes, open bugs, and important technical notes. This document is available on the Sun ONE website.

For an overview of Sun ONE Message Queue concepts, a brief introduction to writing and compiling a client application, see the MQ *Developer's Guide*.

For details on configuring brokers and managing an MQ messaging system, see the MQ *Administrator's Guide*.

For class and member information used when writing a client application, browse the API documentation in the `/usr/share/javadoc/imq` directory.

To uninstall the product, see the following section.

Uninstalling MQ on Solaris

The following instructions explain how to uninstall MQ.

➤ **To remove MQ on Solaris**

1. Stop any running client applications.
2. Stop any running brokers.

   `imqcmd shutdown bkr -u` *name* `-p` *password* `[-b` *hostName:port*`]`
3. Unless you want to retain dynamic broker data, remove all data files associated with each broker instance.

   `imqbrokerd -name` *brokerName* `-remove instance`
4. If you wish to preserve the MQ flat file user repository and the MQ access control file, copy the following files to some safe location before removing MQ packages (they can be restored after re-installing or upgrading MQ):

   `/etc/imq/passwd`

   `/etc/imq/accesscontrol.properties`
5. Determine which MQ packages are installed.

   To see a list of MQ packages installed on your system using `pkginfo`, type:

   ```
   pkginfo | grep SUNWiq
   ```

   The output does not show shared packages (`SUNWaclg`, `SUNWjaf`, `SUNWjhrt`, `SUNWjmail`, and `SUNWxsrt`) installed by MQ on your system. Unless you are updating to a later version of MQ, (and thereby replacing these shared packages with updated versions) it is recommended that you do not remove shared packages.
6. Become root by typing:

   ```
   su root
   ```

   When prompted, type your root password.

7. Remove the MQ packages that were installed with `pkgadd`.

   Issue the following command:

   `pkgrm` *packageName*

   where *packageName* is any of the MQ packages or shared packages that were installed with `pkgadd`. To remove multiple packages, separate the package names by a space.

   Because other products might be using MQ packages, be careful about removing them. The `pkgrm` command will warn you of any dependencies on a package before removing it.

8. When prompted, confirm your removal request by typing **`y`**.

Chapter 3

Linux Installation

This chapter explains the following topics as they apply to a Linux installation:

- Hardware and Software Requirements
- Installing MQ on Linux
- Configuring the Java Runtime for MQ
- Upgrading Editions
- Where To Go Next
- Uninstalling MQ on Linux

Hardware and Software Requirements

At a minimum, your Linux development system should satisfy the minimum requirements indicated in the following table.

Table 3-1 Hardware and Software Requirements for Linux

Component	Requirements
Operating system	Red Hat Linux 7.2
CPU	Intel Pentium 166 MHz (or compatible) PC that is TCP/IP networked
RAM	128 Mbytes

Table 3-1 Hardware and Software Requirements for Linux *(Continued)*

Component	Requirements
Hard drive space	The GNUZIP `tar` file containing the product is approximately 8 Mbytes. The installed product requires approximately 8 Mbytes of hard drive space. MQ, however, may need more space if the broker stores persistent messages locally.
Java 2 Standard Edition (J2SE)	See Table 1-1 on page 17 for the supported versions of the Java Runtime Environment (JRE) and Java Software Development Kit (JDK) that are supported on Linux. The MQ software distribution CD includes the required JRE version at the time of release.

Installing MQ on Linux

The Sun ONE Message Queue product can be downloaded from the Sun ONE website or installed from the product CD-ROM. See the appropriate section below for instructions.

NOTE If you are upgrading from MQ 3.0, it is recommended that you first uninstall MQ 3.0 software, as described in the MQ 3.0 *Installation Guide*, before installing MQ 3.0.1.

Installing from the Web

The following instructions explain how to download and install the MQ product on Linux from the Sun ONE website.

➤ **To install MQ on Linux from the Web**

1. Accept the license agreement.
2. Download the MQ product from the website into an empty, temporary download directory, *temp_directory*.

   The download tar file is: `imq3_0_1-`*edition*`-linx86.tar.gz`

   where *edition* takes one of the following values: `plt` or `ent`, depending on whether you are installing the Platform Edition or Enterprise Edition, respectively.
3. Uncompress the downloaded file.

   `gunzip imq3_0_1-`*edition*`-linx86.tar.gz`

   This creates a single tar file, `imq3_0_1-`*edition*`-linx86.tar`
4. Become root on your system:

   `su root`
5. Change to the /opt directory.

   `cd /opt`

   This is the recommended installation directory. If you install MQ into a *root_path* directory different from `/opt`, you will have to make the corresponding modifications to the steps that follow.
6. Untar the archive:

   `/bin/tar xvfp` *temp_directory*`/imq3_0_1-`*edition*`-linx86.tar`

   The `/opt/imq` directory and its contents are created.
7. If you plan to use the broker (see Table 1-2 on page 18) as a user other than root, change permissions to allow reading and writing to the `/opt/imq/var` directory:

   `chmod 777 /opt/imq/var`

8. Back up the `imq3_0_1-`*edition*`-linx86.tar.gz` file from your temporary working directory.

   This is your logical media. Treat this file as you would any other installation media. Place a copy in a safe location in case you encounter a situation (such as a system failure) that requires reinstallation of the product.

Installing from CD-ROM

The following instructions explain how to install the MQ product on Linux from CD-ROM.

➤ **To install MQ on Linux from CD-ROM**

1. Log in as root or change to superuser.

   For example, type the following at a command prompt:

   ```
   su root
   ```

   Then type your superuser password.

2. Insert the Sun ONE Message Queue CD into your CD-ROM drive and mount it.

   ```
   mount /mnt/cdrom
   ```

   This command may vary depending on the version of Linux being used. Check the `mount` man page on the system.

3. Change to the directory on the CD containing the installation tar file:

   ```
   cd /cdrom/linux
   ```

   The contents of this directory vary depending on the MQ product edition.

4. Copy the MQ tar file from the CD directory into an empty, temporary working directory, *temp_directory*.

   The tar file is: `imq3_0_1-`*edition*`-linx86.tar.gz`

   where *edition* takes one of the following values: `plt` or `ent`, depending on whether you are installing the Platform Edition or Enterprise Edition, respectively.

5. Continue with Step 3 under "Installing from the Web" on page 41.

Configuring the Java Runtime for MQ

At startup time, a broker (the MQ message server) checks to make sure it has access to the required Java runtime version (JDK/JRE 1.4). There are a number of ways you can configure or set the JRE used by the broker. These are shown in the following list, in order of precedence:

1. Pass in the JDK or JRE using either the `imqbrokerd -javahome` or `-jrehome` command line options, respectively (if both are passed in, the last one on the command line will take precedence).
2. Set the JDK or JRE in the `IMQ_JAVAHOME` environment variable.
3. Let the broker use the installed JDK/JRE. The broker will pick up the latest version JDK/JRE installed on the system (greater than 1.4 and less than 2.0).

   The JDK is located in `/usr/java/j2sdk1.`*x.x*, and

   the JRE is located in `/usr/java/j2re1.`*x.x*

To find out why a broker is picking up a specific JDK/JRE, you can start the broker with the following command:

```
imqbrokerd -verbose
```

Upgrading Editions

MQ comes in two editions, as explained in "Product Editions" on page 15.

To upgrade from the Platform Edition to the Enterprise Edition, you need to install the Enterprise Edition license. This upgrade does not overwrite the MQ modules already installed and does not modify the configuration of your MQ messaging system.

To install the Enterprise Edition license you need only extract the license file contained in the Enterprise Edition and place it in your MQ installation directory.

➤ **To upgrade to Enterprise Edition on Linux**

1. Stop any running brokers.

   `imqcmd shutdown bkr -u` *name* `-p` *password* `[-b` *hostName:port*`]`

2. Follow the installation procedure in "Installing from the Web" on page 41, Step 1 through Step 4 or in "Installing from CD-ROM" on page 42, Step 1 through Step 4.

3. Change to the *root_path* directory (where MQ is installed).

   For example, change to `/opt` if MQ is installed in `/opt/imq`.

4. Extract the MQ license file:

   ```
   /bin/tar xvf download_directory/imq3_0_1-ent-linx86.tar
       imq/etc/lic/imqbrokerunl.lic
   ```

   This installs the license into the appropriate MQ directory.

5. Verify that the Enterprise Edition license is available by running

   ```
   imqbrokerd -license
   ```

Where To Go Next

Read the `README` and MQ *Release Notes* files.

- The `README` includes information on where to find documentation, news and updates, and how to send feedback.
- The MQ *Release Notes* contain information on code and documentation changes, open bugs, and important technical notes. This document is available on the Sun ONE website.

For an overview of Sun ONE Message Queue concepts, a brief introduction to writing and compiling a client application, see the MQ *Developer's Guide*.

For details on configuring brokers and managing an MQ messaging system, see the MQ *Administrator's Guide*.

For class and member information used when writing a client application, browse the API documentation in the `IMQ_HOME/javadoc` directory.

To uninstall the product, see the following section.

Uninstalling MQ on Linux

The following instructions explain how to uninstall MQ on Linux.

➤ **To remove MQ on Linux**

1. Stop any running client applications.
2. Stop any running brokers.

   `imqcmd shutdown bkr -u` *name* `-p` *password* `[-b` *hostName:port*`]`
3. Unless you want to retain dynamic broker data, remove all data files associated with each broker instance.

   `imqbrokerd -name` *brokerName* `-remove instance`
4. If you wish to preserve the MQ flat file user repository and the MQ access control file, copy the following files to some safe location before removing MQ packages (they can be restored after re-installing or upgrading MQ):

   `IMQ_HOME/etc/passwd`

   `IMQ_HOME/etc/accesscontrol.properties`
5. Become root:

   `su root`
6. If you want to retain dynamic data for any reason, move `/opt/imq/var` to a temporary directory:

   `mv -rf /opt/imq/var /tmp`
7. Remove the MQ product.

   Issue the `/bin/rm` command. For example, if the product is installed in `/opt/imq` you could type:

   `rm -rf /opt/imq`

Chapter 4

Windows Installation

This chapter explains the following topics as they apply to a Windows installation:

- Hardware and Software Requirements
- Installing MQ on Windows
- Installation Defaults
- Troubleshooting Installation Problems
- Upgrading Editions
- Where To Go Next
- Uninstalling MQ on Windows

Hardware and Software Requirements

At a minimum, your Windows development system should satisfy the requirements indicated in the following table.

Table 4-1 Hardware and Software Requirements for Windows

Component	Requirements
Operating system	Windows 2000 (Professional, Server, Advanced Server) SP2 or Windows XP
CPU	Intel Pentium 166 MHz (or compatible) workstation that is TCP/IP networked
RAM	128 Mbytes

Table 4-1 Hardware and Software Requirements for Windows *(Continued)*

Component	Requirements
Hard drive space	The self-extracting installation file is approximately 25 Mbytes. The temporary directory used for extracting the installation files requires an additional 45 Mbytes. The installed product requires approximately 45 Mbytes. MQ, however, may need more space if the broker stores persistent messages locally.
Java 2 Standard Edition (J2SE)	See Table 1-1 on page 17 for the supported versions of the Java Runtime Environment (JRE) and Java Software Development Kit (JDK) that are supported on Windows. The MQ installation program will (optionally) install the required JRE version in `IMQ_HOME\jre`.

Installing MQ on Windows

The Sun ONE Message Queue product can be downloaded from the Sun ONE website or installed from the product CD-ROM.

The following instructions explain how to install the MQ product either by downloading from the Sun ONE website or using the CD-ROM.

NOTE If you are upgrading from MQ 3.0, it is recommended that you first uninstall MQ 3.0 software, as described in the MQ 3.0 *Installation Guide*, before installing MQ 3.0.1. If you are installing MQ after a previous uninstall, check that any references to the previous MQ installation have been removed from the system's `PATH` environment variable.

➤ **To install MQ on Windows**

1. Quit any other programs you are running.
2. Download the distribution into a temporary working directory.

   If you are installing from CD-ROM, insert the CD into the CD-ROM drive.

3. In the Windows Explorer, double-click the `imq3_0_1-`*edition*`-win.exe` file

   where *edition* takes one of the following values: `plt` or `ent`, depending on whether you are installing the Platform Edition or Enterprise Edition, respectively.

   The installation file is extracted and setup is launched. If you are installing from CD-ROM, the setup program is started automatically.

4. Read the license agreement. Installation and use of the product is subject to acceptance of the license agreement.

5. Follow the setup program's instruction to select installation options.

   a. Select the type of installation.

      - **Compact** Includes only the files necessary to run the broker, administration, and client applications. No documentation or example client applications are installed.
      - **Custom** Lets you specify which MQ modules you want to install. Your options include the broker, client runtime, administration tools, documentation, example client applications, and Java runtime.
      - **Typical** Installs all files, including broker, client runtime, administration tools, documentation, example client applications, and Java runtime.
      - **Feature License** Installs only the files necessary to enable the licensed capacity of an MQ messaging system. This option does not overwrite the MQ files already installed, and does not modify the configuration of your messaging system.

   b. Select a location for the installation.

   c. Select or create a folder for the Windows Start > Programs menu.

   d. Choose whether or not to install the broker as a Windows service. If you choose to install the broker as a Windows service, the broker will be started automatically at system startup and run in the background. This option requires Windows Administrator Group privileges.

      If you choose not to install the broker as a Windows service now but later change your mind, you can use the `imqsvcadmin` utility to do so. See the MQ *Administrator's Guide* for information on using the `imqsvcadmin` utility.

6. Verify installation by choosing Start > Programs > Sun ONE Message Queue 3.0 >Message Broker to run the broker.

   You can also compile and run example applications located in the `c:\Program Files\Sun Microsystems\Message Queue 3.0\demo` directory. See the MQ *Developer's Guide*.

Installation Defaults

The following table indicates the installation defaults for Windows.

Table 4-2 Windows Installation Defaults

Installation directory	The product is installed, by default, into the following directory: `C:\Program Files\Sun Microsystems\ Message Queue 3.0`
Environment Variables	**IMQ_HOME**: The installer automatically sets the `IMQ_HOME` environment variable to the installation directory. **IMQ_VARHOME**: The `IMQ_VARHOME` environment variable defaults to `IMQ_HOME\var`. **IMQ_JAVAHOME**: The `IMQ_JAVAHOME` environment variable defaults to `IMQ_HOME\jre`. **PATH**: The installer automatically sets `PATH` to `%PATH%;%IMQ_HOME%\bin`. This enables you to run utilities, such as `imqbrokerd`, `imqcmd`, and `imqobjmgr` without specifying the entire path.

Troubleshooting Installation Problems

If the MQ installation program does not complete successfully on your Windows platform, try the following remedies. These apply regardless of whether you installed the product from CD-ROM or downloaded it from the Web.

➤ **To troubleshoot installation problems on Windows**

1. Use the Windows Task Manager to end the MQ installation program.
2. Clear any temporary directories (for example, `temp` or `tmp`).

3. Start the MQ installation again, being certain to install to the same directory as the initial installation attempt.

If your Windows installation of the MQ product is still not successful, try this more comprehensive procedure:

1. Use the Windows Task Manager to end the MQ installation program.
2. Clear any temporary directories (for example, `temp` or `tmp`).
3. In the Services control panel, temporarily stop any non-Windows network-related services (for example, a Solstice NFS Server service).
4. Delete the Message Queue 3.0 directory and all its contents.
5. In your Systems control panel, remove all references to MQ 3.0 from your environment settings.
6. Restart your Windows operating system.
7. Reinstall MQ in the same directory location as the initial failed installation.
8. Reset any services in the Services (Control Panel) that you stopped in Step 3.

Upgrading Editions

MQ comes in two editions, as explained in "Product Editions" on page 15.

You can upgrade from the Platform Edition to the Enterprise Edition by installing the Enterprise Edition. The Enterprise Edition is available from the Sun ONE website.

➤ **To upgrade to Enterprise Edition on Windows**

1. Stop any running brokers.

   `imqcmd shutdown bkr -u` *name* `-p` *password* `[-b` *hostName:port*`]`

2. Follow the installation procedure explained in "Installing MQ on Windows." Choose the Feature License option when it appears on your screen.

   This setup does not overwrite the MQ modules installed, and does not modify the configuration of your MQ messaging system.

Where To Go Next

Read the `README` and MQ *Release Notes* files.

- The `README` includes information on where to find documentation, news and updates, and how to send feedback.
- The MQ *Release Notes* contain information on code and documentation changes, open bugs, and important technical notes. This document is available on the Sun ONE website.

For an overview of Sun ONE Message Queue concepts, a brief introduction to writing and compiling a client application, see the MQ *Developer's Guide*.

For details on configuring brokers and managing an MQ messaging system, see the MQ *Administrator's Guide*.

For class and member information used when writing a client application, browse the API documentation in the `IMQ_HOME/javadoc` directory.

To uninstall the product, see the following section.

Uninstalling MQ on Windows

The following instructions explain how to uninstall MQ on Windows.

➤ **To remove MQ on Windows**

1. Stop any running client applications.
2. Stop any running brokers.

   `imqcmd shutdown bkr -u` *name* `-p` *password* `[-b` *hostName:port*`]`

   If you had installed the broker as a Windows service, you can stop it as follows:

   - From the Windows Start menu, choose Settings, then Control Panel.
   - Double-click the Administrative Tools icon, then the Services icon.
   - In the Services panel, select the MQ Broker entry, then click Stop.
   - Close the Services panel.
3. Unless you want to retain dynamic broker data, remove all data files associated with each broker instance.

   `imqbrokerd -name` *brokerName* `-remove instance`

4. Remove the Sun ONE Message Queue product.
   - From the Windows Start menu, choose Programs.
   - From the Sun ONE Message Queue 3.0 program group, choose Uninstall.
   - After Uninstall completes, remove any leftover files.

   The Windows Uninstall utility does not remove any files or directories installed into the product directory that were later modified. It also does not remove any files or directories created by the user as a result of product operations.
5. Manually remove MQ references from your environment.
   - From the Windows Start menu, choose Settings.
   - In the Control Panel, double-click the System icon.
   - On the Environment page, select the `PATH` environment variable and remove the `Message Queue 3.0` reference from the value.
   - Click Set then OK.