MQSeries® for Windows NT® and Windows® 2000

Quick Beginnings

Version 5.2

GC34-5389-01

MQSeries® for Windows NT® and Windows® 2000

Quick Beginnings

Version 5.2

GC34-5389-01

Note!
Before using this information and the product it supports, be sure to read the general information under "Appendix C. Notices" on page 119.

Second edition (December 2000)

This edition applies to IBM MQSeries for Windows NT and Windows 2000, Version 5 Release 2 and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Figures

Tables

Welcome to MQSeries

This book introduces you to IBM MQSeries for Windows NT and Windows 2000, Version 5.2. It contains information about both the server and client components of MQSeries for Windows NT and Windows 2000, V5.2 and describes how to plan for, install, and begin to use the product.

How this book is organized

Part 1 of this book contains the information that you need to know before you install MQSeries:

- "Chapter 1. Planning to install MQSeries" on page 3

  Provides information on hardware and software requirements for the MQSeries for Windows NT and Windows 2000 server, and describes the components that can be installed.

- "Chapter 2. Planning to install the MQSeries for Windows NT and Windows 2000 client" on page 11

  Provides information on hardware and software requirements for the MQSeries for Windows NT and Windows 2000 client, and describes the components that can be installed.

Part 2 of this book contains information about the various methods of installing MQSeries for Windows NT and Windows 2000, and also about the things that you need to do to ensure that MQSeries is working correctly. It also provides information on applying maintenance and removing MQSeries:

- "Chapter 3. Installing MQSeries" on page 15

  Describes how to install MQSeries interactively. It provides both a brief outline of how to install the product, and also a more detailed step-by-step walkthrough of the whole installation process.

- "Chapter 4. Other methods of installing" on page 29

  Describes other ways in which you can install MQSeries, including unattended installation, installing across a LAN, and also how to install using the Microsoft® System Management Server.

- "Chapter 5. Verifying the installation" on page 37

  Describes how to verify that the installation was successful. Information is provided on verifying both a local and a server-to-server installation.

- "Chapter 6. Setting up communications" on page 43

  Explains how to set up communication between multiple MQSeries installations.

- "Chapter 7. Installing the MQSeries client for Windows NT and Windows 2000" on page 45

  Describes the various methods of installing an MQSeries for Windows NT and Windows 2000 client, including interactive and unattended installations, and also installing from a LAN and using the Microsoft System Management Server.
- "Chapter 9. Verifying a client installation" on page 55

  Explains how to verify that the MQSeries for Windows NT and Windows 2000 client has installed successfully.
- "Chapter 8. Configuring communications for the MQSeries client" on page 53

  Describes how to configure communication for the MQSeries for Windows NT and Windows 2000 client.
- "Chapter 10. Applying maintenance" on page 59

  Explains how to apply maintenance information and restore previous backup versions of the product.
- "Chapter 11. Removing MQSeries" on page 63

  Explains how to remove MQSeries in both attended and unattended mode.

Part 3 of this book contains guidance information on getting started with MQSeries:

- "Chapter 12. About MQSeries" on page 67

  Introduces you to the capabilities of MQSeries.
- "Chapter 13. Using MQSeries" on page 77

  Describes how to manage and monitor MQSeries resources using the Microsoft Management Console (MMC) snap-ins (MQSeries Explorer and MQSeries Services), and the MQSeries Web Administration server.
- "Chapter 14. Using MQSeries command sets" on page 85

  Introduces the command sets that can be used to perform system administration tasks on MQSeries objects.
- "Chapter 15. Using the MQSeries Internet Gateway" on page 93

  Introduces the MQSeries Internet Gateway, and explains how to get more information about using it.
- "Chapter 16. Obtaining additional information" on page 95

  Identifies some sources of information that can be useful when you are using MQSeries products.

The **Appendixes** describe the sample programs that are provided with MQSeries for Windows NT and Windows 2000, and provide information on using response files for installation and uninstallation of MQSeries.

Conventions

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

- The term ″MQSeries″ is used to mean the MQSeries for Windows NT and Windows® 2000 product
- The terms ″click″, ″double-click″, and ″right-click″ are used to describe item selection with the mouse. For keyboard alternatives refer to the Windows NT or Windows 2000 help.
- **Boldface** type indicates the name of an item you need to select, the name of a command, or an example in running text.
- *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.

Conventions

What's new in MQSeries for Windows NT and Windows 2000, Version 5 Release 2

MQSeries for Windows NT and Windows 2000, Version 5 Release 2 provides the following new and changed functions:

- Enhancements have been made to the performance of MQI function, channels, message logging, and application initialization and termination.
- Microsoft Transaction Server (MTS) is supported.
- You can now request immediate update of Object Authority Manager (OAM) data, rather than having to stop and restart the queue manager before authorization changes take effect.
- Changes have been made to the way in which OAM data is held, to improve performance.
- Support for Java™ on MQSeries is separately installable from the CD-ROM included in the MQSeries V5.2 product package. Alternatively, you can download the latest version of support for Java on MQSeries from the MQSeries Web site at:

  `http://www.ibm.com/software/mqseries/`
- Support is included for *pipelining*, which is the ability of the Message Channel Agent (MCA) to transfer messages using multiple threads.
- Channel send-exit programs can reserve space in the transmission buffer for their own use. Typically, this would be used by an exit that wanted to encrypt data and add a security key.
- Dynamic Host Configuration Protocol (DHCP) can now be used in queue manager clusters.
- Management of log files for recovery and restart has been improved.
- The area of main storage used to store information relating to a queue manager cluster can be increased dynamically. A new cluster workload-exit call (MQXCLWLN) is provided to support navigation of MQWDR, MQWQR, and MQWCR records held in dynamically increased storage.
- Minor changes to the MQSeries application programming functions have been made, including: support for MQRFH2 (the version-2 rules and formatting header); improvements to the processing of the *CodedCharSetId* field in MQSeries headers; the addition of a command-level value MQCMD_LEVEL_520; and C++ support for MQCNO Version 2 and Version 3.
- Improvements have been made to the installation, default configuration, and Postcard applications.

What's new

- A new folder called Custom has been added to the MQSeries MMC snap-ins that you can use to nominate applications that are to be started automatically when MQSeries starts.
- A new page has been added to the Message Properties sheet that allows you to view the dead-letter header (MQDLH).

For a complete description of new and changed function in this product, see the *MQSeries V5.2 Release Guide.*

Part 1. Planning

Chapter 1. Planning to install MQSeries

This chapter is a summary of the prerequisite hardware and software for running MQSeries, the network protocols and the compilers supported, the delivery media, and the various components of the product.

The chapters in Part 2 of this book that deal with installation, verification, and communication setup, assume that you are using TCP/IP as your communication protocol. You can use other protocols (for example, SNA, SPX and NetBIOS), and, where specific procedures for these protocols are not covered in this book, references are made to other books in the MQSeries library that do contain the relevant information. However, you should note that the following functions of MQSeries are available only under TCP/IP:

- MQSeries Postcard
- MQSeries Explorer

Note: Although machines that use Dynamic Host Configuration (DHCP) can be members of a cluster, it is recommended that the queue manager that hosts the repository for a cluster should be on a machine that has a static IP address.

Release notes

Before starting to install MQSeries, 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 MQSeries Version 5.2 product and may update information in this book.

During installation, the Release Notes file is copied to the Program Files directory and an icon is added to the MQSeries program folder.

Server hardware

This information applies to the server environment only. For information about the MQSeries for Windows NT and Windows 2000 client environment, see “Client hardware” on page 11.

The following are the hardware requirements for the MQSeries server:

- Any IBM® PC machine (or compatible) capable of running Windows NT or Windows 2000
- Any communications hardware supporting SNA LU 6.2, TCP/IP, NetBIOS, or SPX

Server hardware

MQSeries requires a minimum of approximately 40 megabytes (MB) of disk space for product code and data, using NTFS. In addition, allow a minimum of 20 MB for working space.

Server software

This information applies to the server environment only. For information about the MQSeries for Windows NT and Windows 2000 client environment, see "Client software" on page 11.

The following are the prerequisites for running MQSeries; minimum supported levels are shown. Later compatible levels, if any, are supported unless otherwise stated.

- Microsoft Windows NT Version 4.0 (including TCP/IP, NetBIOS, and SPX) and Microsoft Windows NT Service Pack 6a.

  Service Pack 6a is available from the Microsoft Web Site at:

  `http://www.microsoft.com`

  or
- Microsoft Windows 2000 Release 1.

  Any of the following products:
  - Microsoft Windows 2000 Professional
  - Microsoft Windows 2000 Server
  - Microsoft Windows 2000 Advanced Server.
- Microsoft Internet Explorer 4.0.1 with Service Pack 1.

  Available from the Microsoft Web site.

  or
- Netscape Navigator 4.0.4 with the Java AWT patch (containing Java 1.1 enhancements) or later.

  Can be used in place of Microsoft Internet Explorer.
- Microsoft HTML Help 1.22.

  Provided on the MQSeries Server CD-ROM.
- Microsoft MMC 1.1.

  Provided on the MQSeries Server CD-ROM.
- Microsoft ADSI 2.0.

  Provided on the MQSeries Server CD-ROM.

To install the prerequisite software provided on the MQSeries Server CD-ROM (which does not include service packs or Web browsers):

1. Using Windows NT Explorer, select the folder **\Prereqs** on the MQSeries Server CD-ROM.
2. Select the folder for the software item to be installed.
3. Select the folder for the required installation language.
4. Start the installation program.

For information on prerequisites for individual components, see "Prerequisites for MQSeries components" on page 7.

Connectivity

- IBM Communications Server for Windows NT, Version 5.0 and Version 6.0
- IBM Personal Communications for Windows NT, Version 5.0
- Attachmate Extra! Personal Client, Version 6.5
- Microsoft SNA Server, Version 4.0
- TCP/IP, NetBIOS, and SPX are part of the base operating system.

Software supported

The following are options, not prerequisites. Minimum supported levels are shown. Later levels, if any, will be supported unless otherwise stated.

- Transaction processing monitors
  - BEA Tuxedo, Version 6.4 and Version 6.5
  - WebSphere™ Version 3.0x
- Lotus® Notes™, Version 4.5x or later
- Databases
  - IBM DB2 Universal Database®, Version 6.1 and Version 7
  - Sybase Adaptive Server Enterprise, Version 12 (includes open Client, SQL/C) and Version 11.1.1
  - Oracle, Version 8.1.5 and Version 8.1.6
- Software servers
  - Microsoft COM+

Compilers for MQSeries applications

The following software compilers are supported:

- IBM VisualAge® e-business for Windows, Version 1.0.1
- IBM C and C++:
  - IBM VisualAge for C++ for Windows, Version 3.5
  - Microsoft Visual C++ for Windows 95 and Windows NT, Version 6.0 or Version 6.1

Server software

- COBOL:
  - IBM VisualAge COBOL for Windows NT, Version 2.1 and Version 3.0
  - IBM VisualAge COBOL Enterprise, Version 2.2
  - Merant Net Express Version 3.0 and Version 3.1
- PL/I:
  - IBM VisualAge for PL/I for Windows
  - IBM VisualAge PL/I Enterprise, Version 2.1
- Visual Basic:
  - Microsoft Visual Basic for Windows, Version 5.0 and Version 6.0

For latest details, see the MQSeries product family Web site at:

```
http://www.ibm.com/software/mqseries/
```

MQSeries server components

You can select the components you require when you install MQSeries. The components shown below are available when you install the MQSeries server; for information on MQSeries client components, see "MQSeries Client components" on page 12.

Server The MQSeries for Windows NT and Windows 2000 server code.

MQSeries for Windows NT and Windows 2000 client
The client enables Windows applications to run as clients of remote queue managers. The client can be installed on the server machine, enabling you to have the MQSeries server and client on the same machine.

Development Toolkit
Sample MQSeries program source code, including header files, link libraries, and source files for sample applications. These are useful when starting MQSeries application program development. The MQSeries Information Center shows the sample programs that are provided.

Internet Gateway
Provides access to MQSeries applications via HTML and CGI.

Web Administration Server
Provides access to the Web-based administration server.

Documentation in English

Online versions of the books for MQSeries. Included are:

- MQSeries manuals in compiled HTML format
- MQSeries books in PDF format
- MQSeries Internet Gateway Documentation in HTML format

Documentation in Other Languages

Documentation that you have selected (above) in other national languages.

Prerequisites for MQSeries components

The following table shows the prerequisite software for running each of the MQSeries components, and also the installation options used to install the components:

Table 1. Prerequisites and installation options for MQSeries components

MQSeries component	Installation option	Prerequisites
Client	• Custom	• None
Server	• Typical • Compact • Custom	• Internet Explorer 4.01 with SP1 • HTML Help 1.22 • MMC 1.1 • ADSI 2.0
Development Toolkit	• Typical	• None
Internet Gateway	• Custom	• None
Web Administration Server	• Custom	• Internet Explorer 4.01 with SP1, or • Netscape Navigator 4.0.4 with the Java AWT patch (containing Java 1.1 enhancements) or later • HTML Help 1.22
Documentation	• Typical • Custom	• Internet Explorer 4.01 with SP1 • HTML Help 1.22

Components

Component dependencies

Some of the MQSeries installable components have a dependency on other components, as follows:

If you want to install the following components:	You must also install:
Web Administration Server	Server
Internet Gateway	Client
Documentation in Other Languages	Documentation in installation language

You cannot install a component without also installing the component on which it depends. Similarly, if you uninstall a component on which other components depend, you must also remove those (dependent) components.

In the case of an interactive installation or uninstallation, a message is displayed if you make a selection that does not follow these rules. Unattended (silent) installation writes a message (in English) to the installation log and terminates the process.

Migration

Your migration path to MQSeries for Windows NT and Windows 2000 depends on:

- The level of MQSeries for Windows NT you have been using
- Your MQSeries working environment

If you plan to migrate from MQSeries Version 5.0 or MQSeries Version 5.1, you will **not** be able to revert to your previous level. You should back up your system **before** installing MQSeries for Windows NT and Windows 2000. This will enable you to back off the upgrade if necessary. If you do back off the upgrade, however, you will not be able to recover any work performed by MQSeries Version 5.2.

With MQSeries for Windows NT and Windows 2000 the installation process detects whether this is a new installation or an update from a previous level of MQSeries for Windows NT. If you are moving from an earlier level, all the objects you previously created (for example, your queue managers) can be maintained. Your previous component options will be preselected when you install the new level. These components are uninstalled if you then deselect them.

AMQSCOMA.TST

MQSeries system default objects are automatically created when you create a queue manager with this release of MQSeries, so the sample MQSC definition file, AMQSCOMA.TST, is no longer provided. If you used AMQSCOMA.TST to customize your settings for MQSeries Version 5.0, and you want to use the same settings with Version 5.2 of the product:

1. Save your copy of AMQSCOMA.TST
2. Install MQSeries V5.2
3. Load your copy of AMQSCOMA.TST and use the file to recreate your default objects.

Alternatively, you can generate a new MQSC definition file, if required.

.INI file configuration information

MQSeries for Windows NT and Windows 2000 automatically migrates configuration information (from Version 5.0) from your .INI files into the Windows NT registry. Configuration information is then updated in the registry when you define or change details through the user interface.

Default configuration

If you migrate the default configuration of a machine hosting the repository queue manager for a cluster, the other machines in the cluster will no longer have access to the repository and will therefore encounter errors until they also migrate.

Although the default configuration can be set up and used on machines with either DHCP or static IP addresses, it is recommended that the queue manager selected to host the repository for a cluster should be on a machine that has a static IP address.

Migration of scmmqm

This release of MQSeries does not contain the **scmmqm** program (available in Version 5.0). When migrating, the MQSeries installation procedure processes the commands in any startup files that have been registered by using the `scmmqm` command. MQSeries Services properties are set for each command, as follows:

runmqchi
A channel initiator processing the specified initiation queue is automatically started when the queue manager starts.

runmqchl
The specified channel is automatically started when the queue manager starts.

runmqlsr
A listener with the specified parameters is automatically started when the queue manager starts.

strmqcsv
This command is ignored. A command server is automatically started when each queue manager starts.

strmqm
The specified queue manager is automatically started when the MQSeries Service starts.

tpstart This command is ignored. After MQSeries is installed, you should either set your system to run tpstart automatically on system startup, or configure an MQSeries listener for the queue manager with the required TPNAME property.

Java support for MQSeries

If you have applications that require Java support for MQSeries, you will need to install it separately after installing MQSeries. Support for Java on MQSeries is separately installable from the CD-ROM included in this product package. Alternatively, you can download support for Java on MQSeries from the IBM MQSeries Web site, at:

```
http://www.ibm.com/software/mqseries/
```

where the latest version of this support is always available.

If the MQSeries for Windows NT Version 5.1 Java support has been previously installed on the machine, the installation program for Version 5.2 will delete the old Version 5.1 Java files after displaying a warning message.

Chapter 2. Planning to install the MQSeries for Windows NT and Windows 2000 client

This chapter outlines the hardware and software required for use with the MQSeries for Windows NT and Windows 2000 client, and lists the compilers that are supported for applications running on MQSeries clients.

Client hardware

This section outlines the hardware requirements for an MQSeries client for Windows NT and Windows 2000.

An MQSeries client can run on Windows NT or Windows 2000 on any Intel 486 processor based IBM PC machine or equivalent (certified as Windows NT or Windows 2000 compatible). There must be enough random access memory (RAM) and disk storage for the programming prerequisites (below), the MQSeries client, the access methods, and the application programs.

Client software

The following are prerequisites for MQSeries applications running on a Windows NT and Windows 2000 client.

Minimum supported software levels are shown. Later levels, if any, will be supported unless otherwise stated.

- Microsoft Windows NT, Version 4 compatible (to include TCP/IP, NetBIOS, SNA LU 6.2, and SPX) and Microsoft Windows NT Service Pack 6a

  Service Pack 6a is available from the Microsoft Web Site at:

  `http://www.microsoft.com`
- Microsoft Windows 2000 Release 1

Connectivity

- IBM Communications Server for Windows NT, Version 5.0 and Version 6.0
- IBM Personal Communications for Windows NT, Version 5.0
- Attachmate Extra! Personal Client, Version 6.5
- Microsoft SNA Server, Version 4
- TCP/IP, NetBIOS, and SPX and SNA LU 6.2 are part of the base operating system

Client software

Software supported

The following are options, not prerequisites.

- IBM DCE, Version 1.1

Compilers for MQSeries applications (clients)

The following compilers are supported:

- IBM VisualAge C++ for Windows, Version 3.5
- Microsoft Visual C++ for Windows 95 and Windows NT, Version 6.0 and Version 6.1
- IBM VisualAge for PL/I for Windows
- IBM VisualAge PL/I Enterprise, Version 2.1
- IBM Visual Basic for Windows, Version 5.0 and Version 6.0
- IBM VisualAge COBOL for Windows NT, Version 2.1 and Version 3.0
- IBM VisualAge COBOL Enterprise, Version 2.2
- Microsoft Visual Basic for Windows, Version 5.0 and Version 6.0
- Merant Net Express Version 3.0 and Version 3.1

MQSeries Client components

The following components can be installed from the MQSeries Clients CD-ROM:

Client The client enables Windows applications to run as clients of remote queue managers.

MQSeries Development Toolkit
Sample MQSeries program source code, including header files, link libraries, and source files for sample applications. These are useful when starting MQSeries application program development.

MQSeries Internet Gateway
Provides access to MQSeries applications via HTML and CGI.

MQSeries Internet Gateway documentation
MQSeries Internet Gateway documentation supplied in HTML format.

For information on MQSeries Server components, see “MQSeries server components” on page 6, and for information on component dependencies, see “Component dependencies” on page 8.

Part 2. Installing MQSeries for Windows NT and Windows 2000

Chapter 3. Installing MQSeries

This chapter describes how to install MQSeries directly from the CD-ROM in attended mode (interactively).

For information on different methods of installation, see:
- "Installing from a LAN" on page 29
- "Performing an unattended installation" on page 31
- "Using Microsoft System Management Server" on page 34

The installation procedure described is for a TCP/IP environment. The procedure works for other environments (for example, SNA, SPX, and NetBIOS) but you must note that not all of the functions and facilities of MQSeries Version 5.2 are available in these environments. The items that are **not** available are:
- MQSeries Postcard
- MQSeries Explorer

Preparing for installation

Before starting to install MQSeries, 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 MQSeries Version 5.2 product and may update information in this book.

During installation, the Release Notes file is copied to the Program Files directory and an icon is added to the MQSeries program folder.

There are several points to consider before you start installing MQSeries:
- You **must** have administrator authority in order to install MQSeries. Define this authority through the Windows NT facilities.
- You ***cannot*** install MQSeries in a root directory.
- Ensure that you close all Windows programs before installing MQSeries.
- Your user ID ***must*** belong to the 'local' `mqm` or `Administrators` group in order to administer any queue manager on that system, or to run any of the MQSeries control commands. If the local `mqm` group does not already exist on the local computer, it is created automatically when MQSeries is installed. The user ID can belong either directly, or indirectly, by the inclusion of global groups in the local `mqm` group.

- If you intend to administer queue managers on a remote system, your user ID ***must*** be authorized on the target system. The information on protecting MQSeries resources in the *MQSeries System Administration* book includes more information on this topic.
- MQSeries works with the `Security` accounts database. It is important that you give the appropriate definitions in the accounts to the user IDs that are going to use MQSeries.
- A user account that is used to run the IBM MQSeries Services COM server is set up by default during the installation process with the user ID MUSR_MQADMIN. This account is reserved for use by MQSeries.
- For MQSeries authorizations, names of user IDs and groups must be no longer than 20 characters (spaces are not allowed).
- MQSeries does not support machine names that include spaces. If you install MQSeries on a computer with a machine name that contains spaces, you will be unable to create any queue managers.
- The system-defined user ID `System` can also administer any queue manager.
- If MQSeries has been previously installed on the machine, ensure that no queue managers are running and that the IBM MQSeries Service is stopped. (To do this, right-click on the MQSeries icon in the task bar and click **Stop IBM MQSeries**.)
- MQSeries checks for any existing MQSeries configuration files (MQS.INI or QM.INI). If it finds any, it automatically migrates configuration information to the Windows NT registry. Otherwise, MQSeries automatically puts its configuration information directly into the Windows NT registry.

For further information about MQSeries user IDs on Windows NT, and the MQSeries Object Authority Manager (OAM), see the *MQSeries System Administration* book.

Installing MQSeries - quick guide

The installation procedure leads you through a number of windows in which you choose what you want to install. Extensive online help information is available. The installation procedure should take about 15 minutes to complete.

1. Insert the MQSeries for Windows NT and Windows 2000 CD-ROM into the CD-ROM drive.

   If autorun is enabled, the installation process starts. If it is not, double click on `setup.exe` in the root folder on the CD-ROM to start the process.
2. Select the national language you want to use.
3. Follow the actions described in the windows that are presented to you.
4. The Setup Complete window is displayed when you have been through all the relevant windows.

5. Select from the Setup Complete window which of the applications you would like to launch at the end of the installation process, and then click **Finish**.

   Note: If you intend to use the Postcard application to verify the installation (described in "Verifying a local installation" on page 37), make sure that you click **Launch MQSeries Default Configuration Wizard** at this point (this option is presented only if there are no existing queue managers on the system). As a result, the Default Configuration wizard is launched and you can use it to create the default queue manager used by the Postcard application. For more information on using the Default Configuration wizard, see "Using the Default Configuration wizard" on page 21.

6. Verify the installation:
   - To verify that MQSeries is working correctly on a single machine, you can run a local verification using the Postcard application (see "Verifying a local installation" on page 37).
   - If you want to verify that MQSeries is communicating correctly between machines, you can do this using the Postcard application (see "Verifying a server-to-server installation" on page 41).
7. Installation and verification are now complete.

Notes:

1. If the installation process is interrupted, you should go back to the beginning and start the installation again.
2. Following installation, some of the MQSeries icons may not appear in the Start menu until the machine has been restarted. If you need to use these icons without restarting your machine, you can access them by opening the MQSeries folder.
3. If you have applications that require Java support for MQSeries, you will need to install it separately after installing MQSeries. Support for Java on MQSeries is separately installable from the CD-ROM included in this product package. Alternatively, you can download support for Java on MQSeries from the IBM MQSeries Web site, at:

   `http://www.ibm.com/software/mqseries/`

   where the latest version of this support is always available.

   If the MQSeries for Windows NT Version 5.1 Java support has been previously installed on the machine, the installation program for Version 5.2 will delete the old Version 5.1 Java files after displaying a warning message.

Installing MQSeries - step-by-step

This section provides step-by-step guidance on the process involved in installing MQSeries for Windows NT and Windows 2000. In addition to these instructions, extensive online help is available and can be displayed by selecting the **Help** button. The installation procedure should take approximately 15 minutes

Note: The following steps show how to perform a typical installation. The steps involved in a compact and custom installation are given in "Compact installation" on page 21 and"Custom installation" on page 21.

1. Insert the MQSeries for Windows NT and Windows 2000 server CD-ROM into the CD-ROM drive
2. If autorun is installed, the installation process starts. If not, double-click on the Setup icon in the root folder on the CD-ROM to start the installation process.
3. Select the set up language from the pull-down list.
4. Read the information on the MQSeries Welcome panel and click **Next** to continue.
5. Read the information on the Read License panel and accept the license terms.
6. If MQSeries detects that prerequisite software is not already installed on your machine, the Software Requirements panel is displayed, listing the software that is required. From this panel, you can cancel the installation and install the necessary software.
7. Choose the installation folders for program, data, and log files. When you specify installation folders, the data files top-level folder is used only if you install the server or the Windows NT client. The log files folder is used only if you install the server.
8. Choose the type of installation you want: typical (the default), compact, or custom. The components installed with each type of installation are shown in Table 2 on page 19.
9. Click **Program Folder** to choose where you want the MQSeries program icons to be placed. The Select Program Folder window is displayed. This window contains a list of all existing program folders and a field into which you can enter the name of the program folder you want to use (the default is **IBM MQSeries**).
10. On the **Ready to Copy Files** panel, check that you have selected and specified everything that you want before you continue.

Table 2. Components installed with each type of installation

Installation type	Components installed
Typical	This is the default. • Server • Development Toolkit • MQSeries Manuals (in compiled HTML format)
Compact	Server only
Custom	You can select which components you want to install from all of the available components. By default, the following components are selected: • Server • Development Toolkit • MQSeries Manuals (in compiled HTML format)

11. When the installation is complete, the **Setup Complete** window is displayed.

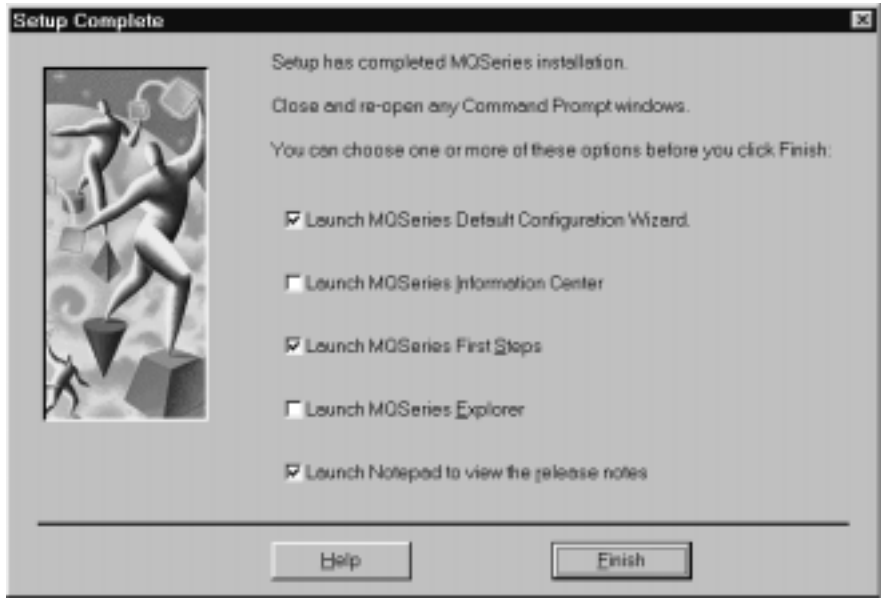

Figure 1. Setup complete

You can choose to launch one or more applications by selecting from the following options:

- **Launch MQSeries Default Configuration Wizard**

  You can use the Default Configuration wizard to add a configured queue manager to this computer for connecting easily with other queue managers in the same MQSeries cluster. This is the recommended option to get you up and running with an MQSeries configuration with the minimum effort on your part. You can use this wizard to create, view or alter your default configuration, which enables you to explore features of MQSeries using the Postcard, API Exerciser, and the MQSeries Explorer.

  This option is not available if other queue managers already exist. If you have previously created any other queue managers on this computer, you will need to delete them before running the Default Configuration wizard.

  Note: If you want to use the Postcard application to verify that MQSeries has installed successfully, it is recommended that you run the Default Configuration wizard first. For more information on using the Default Configuration wizard, see "Using the Default Configuration wizard" on page 21.

- **Launch MQSeries Information Center**

  The Information Center gives you quick access to all task-oriented help information, reference information, and the web-based online books and homepages.

- **Launch MQSeries First Steps**

  The First Steps application provides several small applications which allow new and experienced users to explore the MQSeries network (see"Using MQSeries First Steps" on page 24).

- **Launch MQSeries Explorer**

  The MQSeries Explorer allows you to view and administer your MQSeries network.

- **Launch Notepad to view the release notes**

  The release notes contain information on installing MQSeries and also "late-breaking news" that was not available for the printed publications.

12. Check for messages in the amqilogn.txt file in the MQSeries data files directory (`c:\Program Files\MQSeries`). This application log contains English messages (written during installation) and includes a message indicating whether or not the installation was successful and complete.
13. Following installation, some of the MQSeries icons may not appear in the Start menu until the machine has been restarted. If you need to use these

icons without restarting your machine, you can access them by opening the MQSeries folder. Open a new command prompt window ready to issue MQSeries commands.

Note: If the installation process is interrupted, you should run the installation again from the beginning.

14. You can now verify your installation using the Postcard application as described in "Chapter 5. Verifying the installation" on page 37.

Note: If you have applications that require Java support for MQSeries, you will need to install it separately after installing MQSeries. Support for Java on MQSeries is separately installable from the CD-ROM included in this product package. Alternatively, you can download support for Java on MQSeries from the IBM MQSeries Web site, at:

```
http://www.ibm.com/software/mqseries/
```

where the latest version of this support is always available.

If the MQSeries for Windows NT Version 5.1 Java support has been previously installed on the machine, the installation program for Version 5.2 will delete the old Version 5.1 Java files after displaying a warning message.

Compact installation

Follow the same steps as for a typical installation (shown above). The only difference is that you select **Compact** on the **Setup type** panel. This installs only the Server component of MQSeries for Windows NT and Windows 2000.

Custom installation

1. Follow steps 1 on page 18 to 8 on page 18 of the typical installation, and then select **Custom** on the **Setup Type** panel.
2. Select the components you want to install.
3. Follow steps 9 on page 18 to 14 of the typical installation.

Using the Default Configuration wizard

You can use the Default Configuration wizard to add a configured queue manager to this computer for connecting easily with other queue managers in the same MQSeries cluster. The default queue manager created during the default configuration can be used by the Postcard application, which allows you to verify quickly and easily that your MQSeries installation completed successfully. You can also use the Default Configuration wizard to alter or display details of an existing queue manager created by the default configuration.

1. Start the default configuration wizard in either of the following ways:

- At the end of MQSeries installation, select **Launch MQSeries Default Configuration Wizard** on the Setup Complete window

  or

- After MQSeries has been installed, select **Default Configuration** from the **MQSeries First Steps** application.

2. When you start the Default Configuration wizard, the following window is displayed:

Figure 2. Default Configuration Wizard

3. A default queue manager is created automatically during default configuration, and its name is based upon your computer name.
4. Choose whether to allow remote administration of the queue manager. Select **Allow** to allow a user on a remote MQSeries computer to administer this queue manager. It is recommended that you do this.
5. Choose whether you want this queue manager to join the default cluster (`DEFAULT_CLUSTER`). To enable you to verify the installation quickly and easily using the Postcard application, it is recommended that you do join this queue manager to the default cluster.
6. Select whether the queue manager should be a repository holder or join an existing cluster. If this is the first installation of MQSeries V5.2 on a computer in this cluster, select **Yes** to make this queue manager the

repository holder. If another queue manager (on this or another computer in the cluster) has already been defined as the repository holder, select **No**.

7. The **Repository Location** panel shows where the repository queue manager is located. If you have selected in the previous panel to have the queue manager as the repository, this panel shows the name of your machine. If you selected to have the queue manager join an existing cluster, enter the name of the machine on which the repository for that cluster is located (for example, machine2.hursley.ibm.com).

   When default configuration is complete, you will see a window similar to the following:

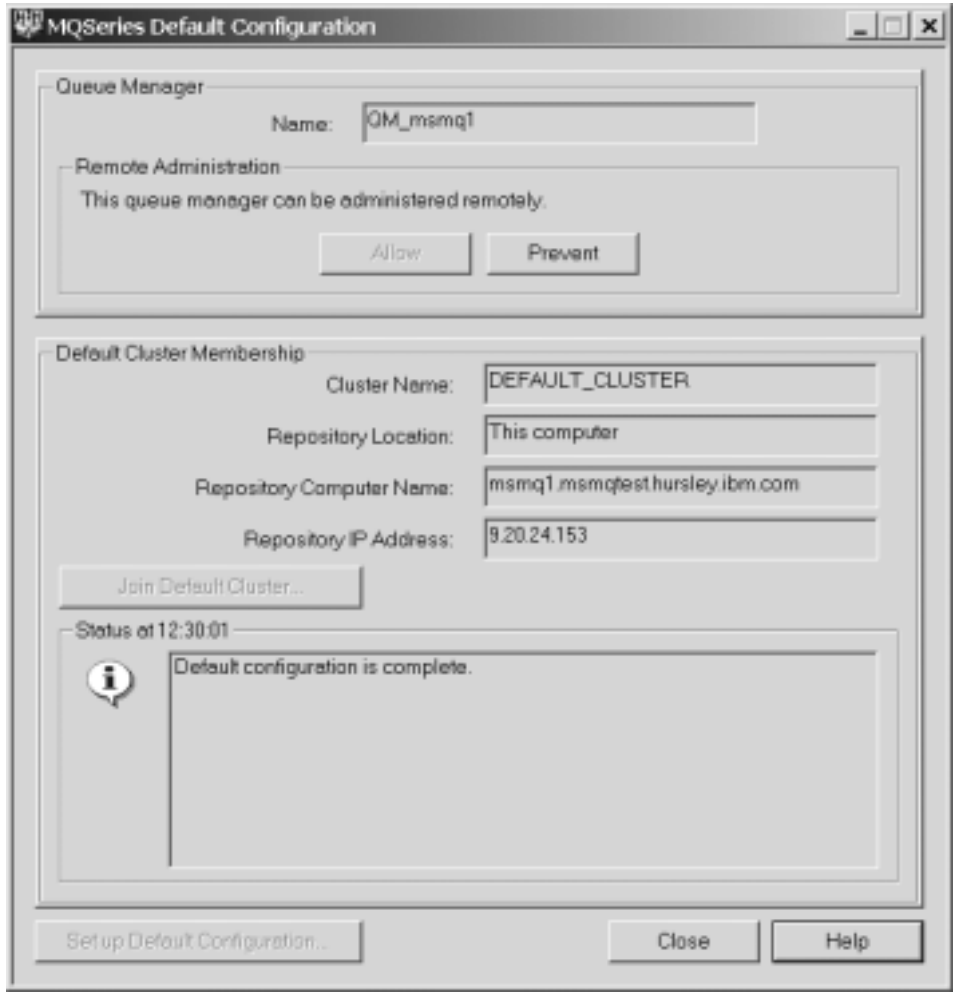

Figure 3. Default Configuration complete

Using MQSeries First Steps

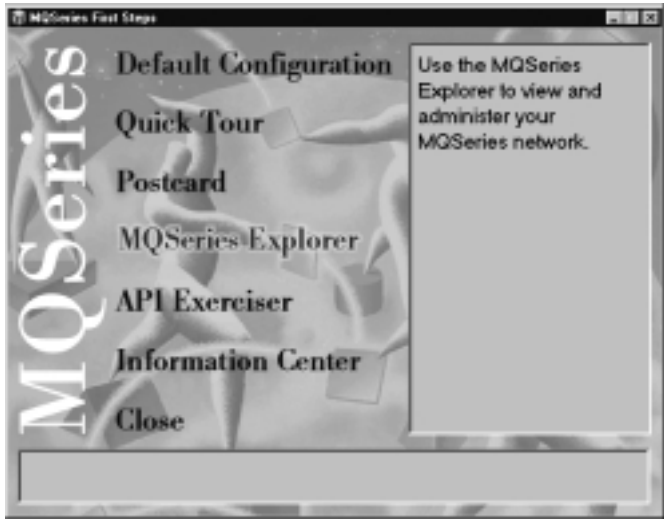

Figure 4. MQSeries First Steps

You can use the items in the MQSeries First Steps window to explore the facilities in MQSeries.

- **Default Configuration**

  Allows you to add a configured queue manager to this computer for connecting easily with other queue managers in the same MQSeries cluster. You can also use it to alter or display details of an existing queue manager created by the default configuration. This feature is available only using TCP/IP.

  Note: If you have created any queue managers since installing MQSeries, you cannot add the Default Configuration unless you delete all of the existing queue managers first.

- **Quick Tour**

  Gives a brief overview of MQSeries and helps you to learn more about the concepts and functions of the product.

Figure 5. Quick Tour

- **Postcard**

  Allows you to try out MQSeries messaging quickly and easily. You can send a message either to your own machine or to another named user's machine very simply. It is described in detail in "Chapter 5. Verifying the installation" on page 37.

- **MQSeries Explorer**

  Allows you to view and administer your MQSeries network. For more information on using the MQSeries Explorer, see "MQSeries Explorer" on page 78.

- **API Exerciser**

  Allows you to experiment with the API calls that are provided in the MQSeries programming interface.

  The Information Center provides help on using the API Exerciser, and can be displayed by selecting the **Help** button in the API Exerciser. You can also look at the *MQSeries Application Programming Guide* and the *MQSeries Application Programming Reference* manual from the "Reference" section of the Information Center.

Figure 6. API Exerciser

- **Information Center**

  Gives you fast access to task-oriented help information, reference information, and Web-based online books and homepages.

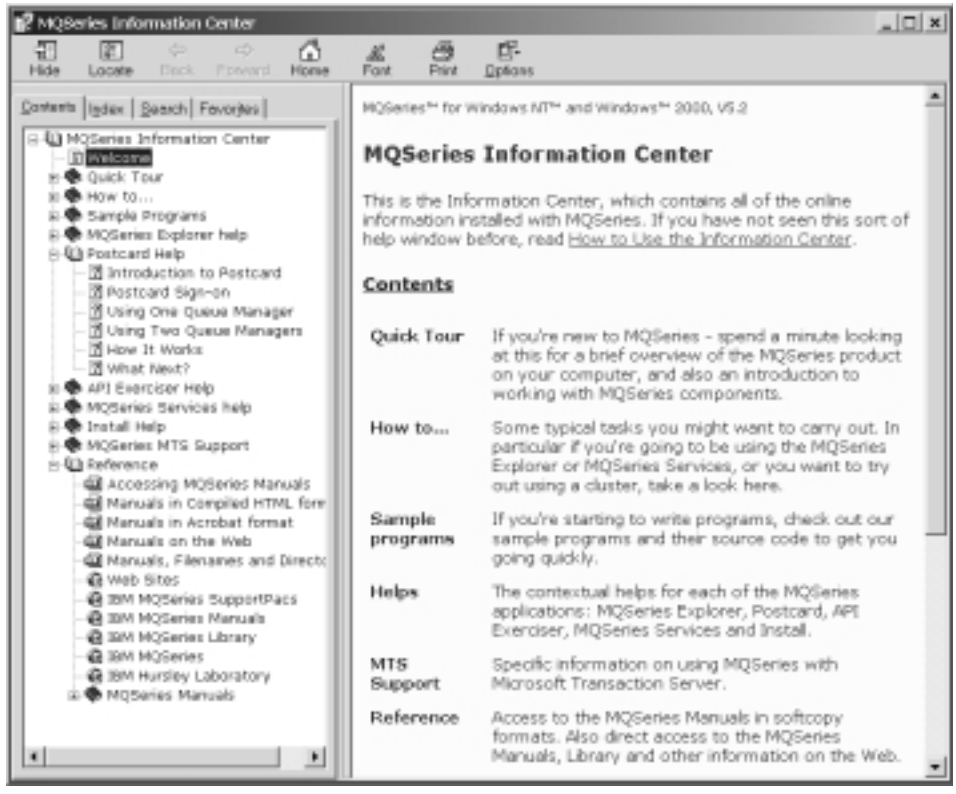

Figure 7. Information Center

Chapter 4. Other methods of installing

The previous chapter (“Chapter 3. Installing MQSeries” on page 15) describes how to install MQSeries interactively using the server CD-ROM. However, there are also other methods of installing MQSeries for Windows NT and Windows 2000:

- “Installing from a LAN”
- “Performing an unattended installation” on page 31
- “Using Microsoft System Management Server” on page 34

Before starting to install MQSeries, 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 MQSeries Version 5.2 product and may update information in this book.

During installation, the Release Notes file is copied to the Program Files directory and an icon is added to the MQSeries program folder.

Installing from a LAN

There are two ways to put MQSeries installation files on a LAN server for easier access:

1. You can make the MQSeries CD-ROM drive shareable, or:
2. You can copy the installation files from the CD-ROM to a server, by following these steps:
   a. Create a folder on the LAN server to store the installation files. For example:

      ```
      md m:\instmqs
      ```

   b. Load the MQSeries server CD-ROM. If you have autorun enabled, the language selection dialog will appear; cancel this dialog.
   c. Copy the entire CD-ROM to the m:\instmqs folder. For example:

      ```
      xcopy e:\*.* m:\instmqs /e
      ```

      Note: You can save space on the hard drive by copying only the Setups, Docs, and Prereqs subfolders for the languages you require.

The language subfolders are:
- en_us - English
- fr_fr - French
- de_de - German
- es_es - Spanish
- it_it - Italian
- ja_jp - Japanese
- ko_kr - Korean
- pt_br - Brazilian Portuguese
- zh_cn - Simplified Chinese
- zh_tw - Traditional Chinese

Also, you only need copy the Docs folder if you intend installing any of the online documentation.

d. Give all licensed users access to the folder that now contains the CD-ROM image (in this example, the m drive).

e. From a command prompt on the target machine, connect to the appropriate drive and folder using the `net use` command as follows:

```
net use devicename \\servername\netname
```

For example:

```
net use x: \\mqmnt\instmqs
```

where `x:` is the required mapped drive on the target machine.

Alternatively, you can use the Windows NT Explorer or some other method to map the shared resource to a drive letter.

f. Change to the installation directory (in this example x:).

g. Type **Setup**, and press Enter.

h. Follow the prompts.

Performing an unattended installation

You can install MQSeries on a remote machine without interaction, provided that the remote machine can share the server CD-ROM, or a copy of the files on it, and that you can execute a command on the remote machine. This process is called unattended (or silent) installation, and is particularly useful for installing MQSeries over a network because you can do it from a shared drive on a LAN server.

Because there is no user interaction, unattended installation uses a response file. A response file is an ASCII text file that contains values for the installation options you want to select. For more information on using response files, see "Appendix B. Using response files for installing and removing MQSeries" on page 111.

There are three ways to generate a response file for installation:

1. Copy and edit the response file (setup.iss) supplied in the Setups folder of the MQSeries Server CD-ROM, using an ASCII file editor.
2. Generate your own response file, using an ASCII file editor.
3. Carry out an installation on a machine and record the options selected to install the product in a response file. To do this you must run `setup.exe` with the -r parameter and (optionally) the -noinst parameter. Do this as follows:
   a. Load the MQSeries Server CD-ROM. If you have autorun enabled, the language selection dialog appears; cancel this dialog.
   b. Run `setup.exe` with the `-r` parameter, and, optionally, the -noinst parameter. If you specify -noinst to suppress the MQ installation, it must come before the `-r` parameter. You run setup.exe by dragging setup.exe from Windows NT Explorer to the Run dialog in the Start menu and adding `-r` to the end of the file name, or by typing the fully-qualified path to setup.exe into the Run dialog, followed by `-r`.
   c. The language selection dialog reappears. If you are only generating a response file, you do not need to restart the system, even if asked to do so.

The installation creates a file called setup.iss in the Windows directory, normally c:\winnt. Save this file; you can edit it, if necessary.

Use the response file you have created to install the product on a remote machine (see "Installing on the remote machine" on page 32).

See "Installation response file format" on page 111 for more information on creating a response file.

Installing on the remote machine

The remote machine must have access to a shared resource or drive on a server that contains the MQSeries Server CD-ROM or a copy of it. Perform the procedure described in “Installing from a LAN” on page 29 up to (and including) step 2f on page 30. Copy your response file to a location on the server that is accessible from the remote machine.

You can now start the installation on the remote machine in one of two ways:

1. On the remote machine, change to the Setups\xx_xx directory on the shared resource. Substitute xx_xx with the name of the language subfolder for the language that you require (see “Installing from a LAN” on page 29).

   Run setup.exe as follows:

   ```
   setup -g<logfile> -f1<responsefile> -f2<secondarylogfile> -s
   ```

   Note: Enclose long path name and file name expressions in double quotes. For example:

   ```
   -g"c:\log file.txt"
   ```

   This method installs MQSeries in the language specified by xx_xx.

2. On the remote machine, go to the root folder on the shared resource and run setup.exe as follows:

   ```
   setup -l<language> -g<logfile> -f1<responsfile> -f2<secondarylogfile> -s
   ```

   Note: Enclose long path name and file name expressions in double quotes. For example:

   ```
   -g"c:\log file.txt"
   ```

   This method installs MQSeries in the language specified by the -l parameter.

In the above:

<language>

The installation language.

If you omit the -l<language> parameter, or specify -lDefault, Setup runs in the default language of the user who is running Setup. If that language is not one of those supported by MQSeries, the machine's default language is used. If that language is not supported, English is used.

The value of <language> must be one of the following and must be in English (case is ignored):

Default
English or en_us
French or fr_fr
German or de_de
Spanish or es_es
Italian or it_it
Japanese or ja_jp
Korean or ko_kr
Brazilian Portuguese or pt_br
Simplified Chinese or zh_cn
Traditional Chinese or zh_tw

<logfile>

the full path to an installation log file. Setup creates an English ASCII text log file containing details of what happens during installation. You should check this file to see if any errors occurred.

If you omit the -g<logfile> parameter, Setup creates a file called amqilogn.txt in the data-files folder on the machine running the installation. In this case, any messages generated before the data-files folder is created are lost.

The -g<logfile> parameter must be placed before the -f1, -f2, and -s parameters; otherwise, it is ignored.

<responsefile>

The full path to the response file you prepared. If you omit the -f1<responsefile> parameter, the response file must be in the Setups\xx_xx language subdirectory.

<secondarylogfile>

The full path to a secondary log file. This file contains less detail than the other log file and should not be used as the primary source for information about the success of the unattended installation.

If you omit the `-f2<secondarylogfile>` parameter, Setup attempts to create a file called `setup.log` in the language subdirectory (xx.xx), which is not possible if the installation is being performed from CD-ROM.

`-s`

This parameter tells Setup to run in silent mode.

After installing, check the log file to ensure that it says the installation was successful and complete.

Using Microsoft System Management Server

There are two steps involved in installing MQSeries using the Microsoft System Management Server: creating an SMS software package (see "Creating the MQSeries SMS software package") and then creating an SMS job to distribute and install the package (see "Creating the MQSeries SMS job" on page 35).

Note that you need an MQSeries response file (see "Installation response file format" on page 111 and "Uninstallation response file format" on page 116) if you want to use the Microsoft System Management Server to install or remove an MQSeries for Windows NT server or client.

For more detailed information on how to create a software package and a job, refer to the Microsoft System Management Server documentation.

Creating the MQSeries SMS software package

To create the SMS software installation package:

1. From the Microsoft SMS Administrator application, open the **Packages** folder and create a new package.
2. In the SMS **Package Properties** dialog click on the **Import** button to create the software package by importing a Package Definition File (PDF).
3. In the **File Browser** dialog, select the drive where the IBM MQSeries CD-ROM is located.
4. Select the current root directory which contains the package definition file `MQSERIES.PDF`.

   You can also find the `MQSERIES.PDF` file in the local drive, or shared network drive to where you copied the MQSeries Installation software.
5. Select the **MQSERIES.PDF** file and click on the **OK** button.

6. Click on the **Workstation** button. In the **Source Directory** entry field, specify the fully qualified path name to the MQSeries root directory which contains the MQSeries installation software. See "Installing from a LAN" on page 29.
7. Select the appropriate Workstation Command Line:
   - **Automated Deinstallation of IBM MQSeries - NT**
   - **Automated Installation of IBM MQSeries - NT (US English)**
8. Click on the **Properties** button for each process and review the **Command Line** entry field to ensure that the parameters are what you require.

   Note: The `-i<miffilepath>` parameter specifies the full path and file name of a Management Information Format (.mif) file that installation and uninstallation can generate. Remove this parameter if you do not want to generate a .mif file.
9. Click on the **Close** button to close the **Workstation Properties** dialog.

   Note: If you specified a local path in the **Source Directory** entry field, you will get a pop-up dialog warning you that the local path you specified may not be accessible to SMS components running on another machine. Click on the **OK** button to continue.
10. Click on the **OK** button to close the **Package Properties** window.

    A pop-up dialog appears indicating that SMS will update the software package at all sites. Click the **OK** button to continue.

The software package has been created and can be installed by creating an SMS job.

Creating the MQSeries SMS job

You must now create an SMS job to distribute and install the software packages you created, which contain the MQSeries installation software.

Refer to the Microsoft System Management Server documentation for detailed information on how to create and run a job.

Notes:

1. You ***must*** be logged onto the target machine with Administrator authority in order to install the MQSeries Server.
2. When creating an SMS Job to distribute and install the software package, ensure that you select the appropriate workstation command. The workstation commands are displayed on the **Job Details** dialog in the **Run Phase** section and appear in a drop-down listbox.

Chapter 5. Verifying the installation

Before you can use MQSeries for Windows NT and Windows 2000, you need to verify that the product has installed correctly. You can verify:

- A local installation (see "Verifying a local installation")
- A server–to–server installation (see "Verifying a server-to-server installation" on page 41)
- A client–to–server installation (see "Chapter 9. Verifying a client installation" on page 55)

You can verify a *local* installation (which has no communication links with other MQSeries installations), using the Postcard application supplied with MQSeries. It allows you to send a message either to your own machine or to another named user's machine that is running MQSeries in the same cluster. When you or the other user sees the message arrive, you can safely assume that MQSeries has been successfully installed and that your communication links are working properly.

If you want to verify a TCP/IP communication link between multiple MQSeries installations (for example, between two servers) using the Postcard application, you must ensure that both computers are part of the same cluster, either by running the default configuration on both machines to link them to the default cluster, or by creating your own queue managers on both machines, creating a cluster, and ensuring that the queue managers that you have created belong to the same cluster. The simplest and fastest way of configuring the machines ready to use the Postcard application is to run the Default Configuration wizard on both machines (for more information on using the Default Configuration wizard, see "Using the Default Configuration wizard" on page 21).

Verifying a local installation

Note: TCP/IP must already be installed on the machine, and a queue manager must have been previously set up that can be used as a mailbox (this can be either the default queue manager, which is set up automatically during default configuration, or another queue manager that you have set up yourself).

To verify that the local installation is working, you can send a message to yourself using the Postcard application. This allows you to create two

postcards on the same machine and send messages between them, verifying that MQSeries messaging is working correctly on the machine.

1. Go to the MQSeries First Steps folder (if you selected First Steps in the final installation panel, this will be open already).

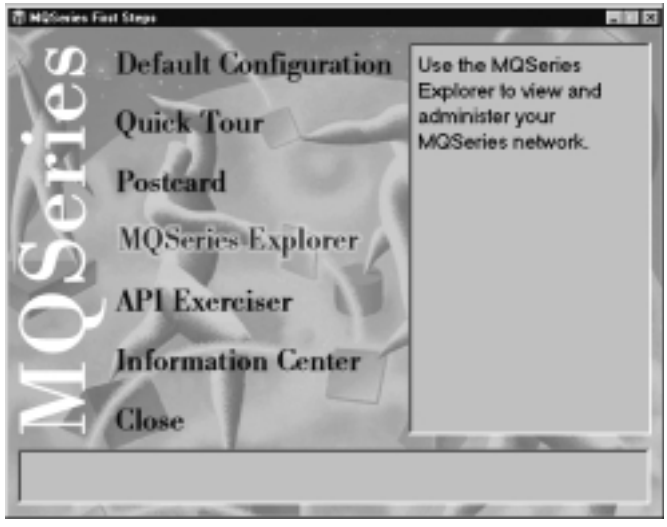

Figure 8. MQSeries First Steps

2. Click **Postcard**
3. The **MQSeries Postcard - Sign on** panel is displayed.

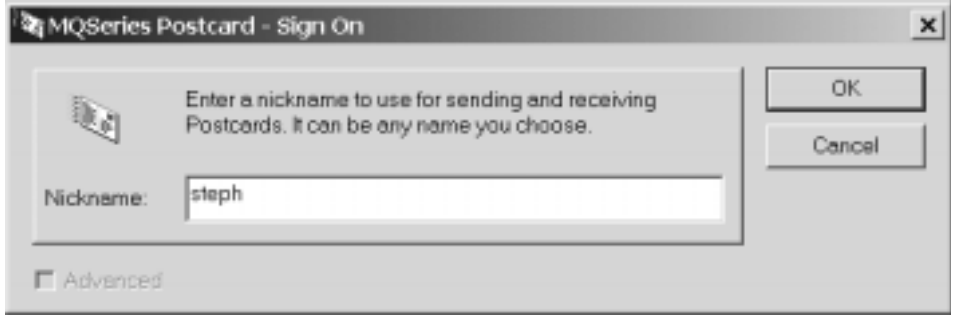

Figure 9. Postcard Sign-on

Type in a nickname that you want to use for sending messages within the postcard application (for example, Steph. If more than one queue manager exists (including the a default queue manager set up during default configuration), click the **Advanced** checkbox to display a list of queue managers that can be used as the mailbox, and select the one that

you want to use.. If there are no existing queue managers (or if the only queue manager is the default queue manager set up during default configuration), the Advanced checkbox is unavailable.

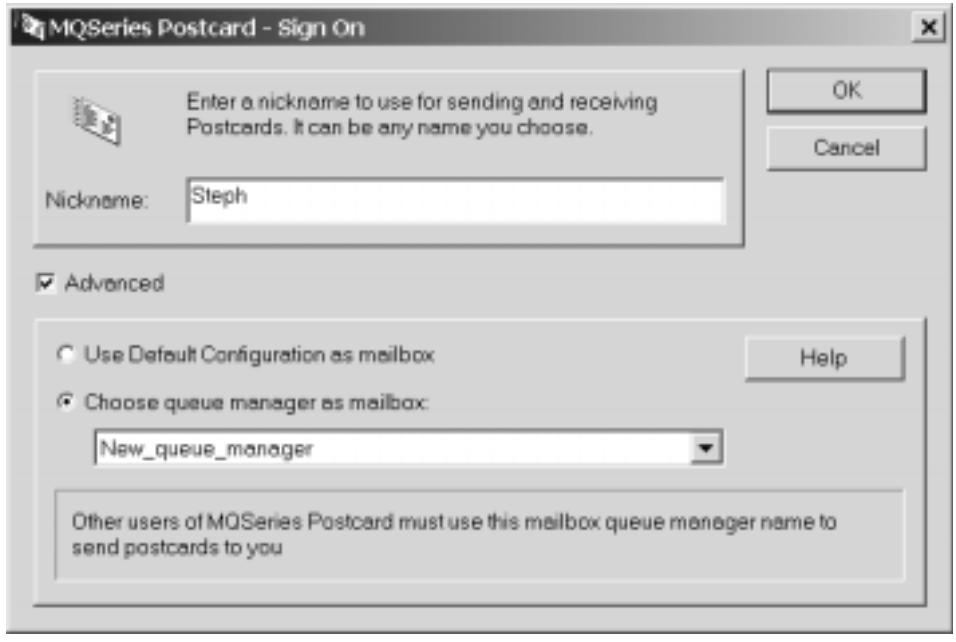

Figure 10. Postcard sign-on panel with a queue manager chosen as the mailbox

- If you have previously installed a default queue manager using the Default Configuration, (and you have set up no other queue managers) this will be used as your mailbox for postcards.
- If you have created one or more of your own queue managers (but have not run the Default Configuration) you must select the name of the queue manager that will be used as the mailbox for postcards.
- If you have run the Default Configuration and you have created one or more of your own queue managers, you must choose whether to use the Default Configuration queue manager or one of the other queue managers as the mailbox. Click either **Use Default Configuration as mailbox** or **Choose queue manager as mailbox**, and, if you are using a queue manager other the Default Configuration, select the appropriate queue manager from the list displayed.

4. Your first “postcard” is displayed.
5. Click **Postcard** again (from First Steps) to open a second postcard window.

6. The **MQSeries Postcard - Sign on** panel is displayed again. Type in a second nickname that you want to use for sending messages within the postcard application (for example, Joho).
7. Repeat the selection of the queue manager that you want to use as the mailbox (as shown above). The mailbox **must** be the same as the one specified in the first postcard.
8. You now have two postcards, one with the nickname Steph and one with the nickname Joho.
9. In one of the postcards (for example, Steph) type some message text in the **Message** box, the nickname of the other postcard (for example, Joho) in the **To** box, and the name of the machine that the recipient (Joho) is on in the **On:** field (in this case both Steph and Joho are on the same machine). The machine name will be something like machine1.hursley.ibm.com. If both the sender and receiver are on the same machine, you do not need to specify the machine name in the **On:** field. Click **Send**.

Figure 11. Completed postcard

10. Your message is displayed as *Sent* in the **Postcards sent and received** box in one postcard, and as *Received* in the other postcard.
11. Double-click on the received postcard to view it.
12. This verifies that MQSeries was installed correctly.

What next?

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

- Install the MQSeries product on other machines. If so, follow the same installation procedure you have just been through. Make sure you use the Join Default Cluster window to add the other machines to your first machine's cluster.
- Install the MQSeries for Windows NT and Windows 2000 client on other machines. See "Chapter 7. Installing the MQSeries client for Windows NT and Windows 2000" on page 45 if you need to do this.
- Install other MQSeries clients in your network. Refer to the *MQSeries Clients* book for full details if you need to do this task.
- If you want to continue with further administration tasks, go to "Part 3. Getting started with MQSeries" on page 65.

Verifying a server-to-server installation

You can use the Postcard application to verify communication between your machine and another named user's machine that is running on TCP/IP in the same MQSeries cluster. Therefore you can use Postcard to verify that you can communicate only with another server that supports clusters and is using TCP/IP. Before you start:

- Make sure that TCP/IP and MQSeries are installed on both machines.
- Check that both machines are part of the same cluster and that only one of them has its queue manager defined as the repository. If they are not part of the same cluster, you need to configure communications between the two machines as a separate step (see "Chapter 6. Setting up communications" on page 43).

The following steps describe how to verify communication between two machines – the *sender* of the message and the *receiver*.

On the sender machine:

1. Go to the MQSeries First Steps folder (if you selected First Steps in the final panel of installation, this window will be open already).
2. Click **Postcard**.

3. The **MQSeries Postcard - Sign on** panel is displayed. Type in a nickname that you want to use for sending and receiving messages within the postcard application, and then select which queue manager you want to use as the mailbox:
   - If you have previously installed a default queue manager using the Default Configuration, (and you have set up no other queue managers) this will be used as your mailbox for postcards.
   - If you have created one or more of your own queue managers (but have not run the Default Configuration) you must select the name of the queue manager that will be used as the mailbox for postcards.
   - If you have run the Default Configuration and you have created one or more of your own queue managers, you must choose whether to use the Default Configuration queue manager or one of the other queue managers as the mailbox. Select either **Use Default Configuration as mailbox** or **Choose queue manager as mailbox**, and, if you are using a queue manager other the Default Configuration, select the appropriate queue manager from the list displayed.
4. Click **OK**. Your first MQSeries Postcard is displayed.
5. Type in some message text in the **Message** field, the recipient's nickname in the **To** field, and the recipient's machine name (for example, `computer2.hursley.ibm.com`) in the **On:** field. Select **Send**

On the receiver machine:

1. To receive the message, open a Postcard (from the First Steps folder).
2. Type in the nickname (to which the sender machine sent the message) and select **Enter**.
3. A message appears in the **Postcards sent and received** field saying that the message has been received. When this message arrives you will have verified that MQSeries has been correctly installed and that your communication link between the two machines is working correctly.

Now that verification is complete you are ready to start using MQSeries (see "Part 3. Getting started with MQSeries" on page 65).

Chapter 6. Setting up communications

This chapter explains how to set up communications for the MQSeries for Windows NT server.

Communications must be configured between the two machines if MQSeries is to exchange messages with MQSeries on another machine. You will not be able to run a server-to-server verification process until cross-server communication has been enabled, unless you are using the Postcard application.

MQSeries supports the TCP/IP, SNA LU 6.2, NetBIOS, and SPX protocols. The installation process automatically sets up a default TCP/IP configuration, if requested.

This chapter describes the steps involved in setting up MQSeries to enable it to communicate using TCP/IP. For more information and extensive examples of using other transmission protocols, see the *MQSeries Intercommunication* manual.

For information on setting up communications between MQSeries servers and clients, see "Chapter 8. Configuring communications for the MQSeries client" on page 53.

Defining a connection

To enable two machines to exchange messages, communication *channels* must be defined between the machines. The channels must be defined on each machine, with a *sender channel* defined on the sender machine, and a *receiver channel* defined on the receiver machine. The channel definition at the sending end specifies the address of the target (the receiver machine).

Before the receiving machine can receive any messages from the sender, a *listener* must be running on the receiver machine.

The following sections guide you through the creation of the channels on the sender and receiver machines.

Sending end

To define a sender channel:

1. Start the MQSeries Explorer from the MQSeries folder.
2. Click the plus sign (+) to the left of **Queue Managers** item in the list.
3. Click the plus sign (+) to the left of your default queue manager in the list.
4. Click the plus sign (+) to the left of **Advanced.**
5. Click **Channels** (so that it is highlighted).
6. Click the **Action** menu, then click **New** and then **Sender channel.**
7. Use the online help to guide you in filling in the details of the **Create Sender Channel** panel.
8. When this is complete, create the receiver channel on the receiver machine as shown in the next section.

Receiving end

To define a receiver channel:

1. Start the MQSeries Explorer from the MQSeries folder.
2. Click the plus sign (+) to the left of **Queue Managers** item in the list.
3. Click the plus sign (+) to the left of your default queue manager in the list.
4. Click the plus sign (+) to the left of **Advanced.**
5. Click **Channels** (so that it is highlighted).
6. Click the **Action** menu, then click **New** and then **Receiver channel.**
7. Use the online help to guide you in filling in the details of the **Create Receiver Channel** panel.

Receiving channel programs are started in response to a startup request from the sending channel. To enable this to happen, a listener program has to be started to detect incoming network requests and start the associated channel.

To start the MQSeries listener, use the RUNMQLSR command:

```
RUNMQLSR -t tcp [-m QMNAME] [-p 1822]
```

The square brackets indicate optional parameters; `QMNAME` is not required for the default queue manager, and the port number is not required if you are using the default (1414).

What next?

- To verify the communication links, follow the steps given in "Verifying a server-to-server installation" on page 41.
- To find out more about how to use MQSeries, see "Part 3. Getting started with MQSeries" on page 65.

Chapter 7. Installing the MQSeries client for Windows NT and Windows 2000

You can install the MQSeries for Windows NT and Windows 2000 client in any of the following ways:

- Interactively using an MQSeries CD-ROM (see "Installing the MQSeries client" on page 46)
- From a LAN (see "Installing from a LAN" on page 47)
- Unattended (see "Performing an unattended installation" on page 50)
- Using SMS (see "Using the System Management Server with MQSeries for Windows NT and Windows 2000" on page 48)

There are two versions of the MQSeries for Windows NT and Windows 2000 client installation code, one on the MQSeries Server CD-ROM, and one on the Client CD-ROM. The one you use depends on whether or not the machine on which you are installing will also contain the MQSeries server software:

- If you want to install the MQSeries client on a machine that will *not* contain any of the MQSeries server software, use the MQSeries Client CD-ROM, as explained in this chapter.
- If you want to install the MQSeries client on an MQSeries server machine, use the MQSeries Server CD-ROM (not the MQSeries Client CD-ROM), and select the Client component.

You may install the client from the MQSeries Client CD-ROM and later decide to install the MQSeries server on the same machine. Use the MQSeries Server CD-ROM to install the server and the client.

See the *MQSeries Clients* manual for information about other clients that can run under Windows NT and Windows 2000.

When you have finished installing the MQSeries client, you need to set up communication between the client and server (see "Chapter 8. Configuring communications for the MQSeries client" on page 53) and then verify that they are installed and communicating successfully (see "Chapter 9. Verifying a client installation" on page 55).

Installing the MQSeries client

To install an MQSeries client for Windows NT and Windows 2000, you must be logged on to Windows as an administrator.

MQSeries checks for any existing MQSeries configuration files (MQS.INI). If it finds any, it automatically migrates configuration information to the Windows NT or Windows 2000 registry. Otherwise, MQSeries automatically puts its configuration information directly into the Windows NT or Windows 2000 registry.

The following instructions assume that you are installing an MQSeries client using the MQSeries Client CD-ROM supplied as part of the MQSeries product. If you plan to install an MQSeries client and server on the same machine, see the notes in the previous section of this chapter.

1. Insert the MQSeries Client CD-ROM into the CD-ROM drive.
2. The installation automatically starts and an MQSeries Language Selection window is displayed.

   Note: If you have disabled auto-playing of CD-ROMs, run SETUP instead, from the root directory.

   This window presents you with a list of the national languages that are available.
3. On the MQSeries Language Selection window select the language of your choice, and click **OK**.

   The MQSeries Welcome window is displayed.
4. ***Make sure you are installing the correct client*** for your system, as displayed in the Welcome window.
5. Choose Installation Folders lets you choose folders into which the MQSeries program files and data files will be installed.

   You can change the default shown by clicking the browse button and choosing a different drive and directory, then click **OK**. Click **Next** to continue.
6. Select Components displays a list of components from which you can select the ones you want to be installed.

   To select a component, click in the box next to it so that a check mark appears (just highlighting the line does not select it). The panel displays information about the amount of space available on the selected drive, and the amount of space required for each of the components.

   If you deselect a previously installed component it is removed. Click **Next** to continue.

7. Select Program Folder prompts you for a folder name to contain the MQSeries objects. The default name is IBM MQSeries Client. You can rename the default or choose an existing folder.
8. Ready to Copy Files displays all the selections you have made. Click **Back** if you want to return to a previous window to change any of your choices.

   When you have checked your choices, click **Next** to start the file copying process.

   The progress indicator shows which components are being copied and the percentage of copying completed.
9. The Setup Complete window appears when the selected components have been installed. Click **Finish** to close the window (after optionally selecting **Launch Notepad to View the Release Notes**).
10. The installation of the MQSeries client is now complete. When setup is complete, the IBM MQSeries Client folder is added to the Start menu. Note that MQSeries Clients are sets of services and do not have to be explicitly run, so the folder does not have an object called “client”.
11. You now need to establish communication between the MQSeries client and server (see “Chapter 8. Configuring communications for the MQSeries client” on page 53)
12. Finally, verify that the client was installed successfully (see “Chapter 9. Verifying a client installation” on page 55).

Installing from a LAN

There are two ways to put MQSeries installation files on a LAN file server for easier access: you can make the MQSeries for Windows NT and Windows 2000 Client CD-ROM drive shareable, or you can copy the installation files from the CD-ROM to a file server, by following these steps:

1. Create a folder on the LAN file server to store the installation files. For example:

   ```
   md m:\instmqs
   ```
2. Load the MQSeries client CD-ROM. If you have autorun enabled, the language selection panel will appear. Cancel this panel.
3. Copy the Winnt directory structure from the CD-ROM to the m:\instmqs folder. For example:

   ```
   xcopy e:\Winnt\*.* m:\instmqs /e
   ```

   You can save space on the hard drive by copying only the subfolders for the languages that you require.

The language subfolders are:

SetupCn	Simplified Chinese
SetupDe	German
SetupEn	U.S. English
SetupEs	Spanish
SetupFr	French
SetupIt	Italian
SetupJp	Japanese
SetupKo	Korean
SetupPt	Brazilian Portuguese
SetupTw	Traditional Chinese

4. Give all licensed users access to the folder that now contains the CD-ROM image (in this example, the m: drive).
5. From a command prompt on the target machine, connect to the appropriate drive and folder using the `net use` command:

   ```
   net use devicename \\servername\netname
   ```

   For example:

   ```
   net use x: \\mqmnt\instmqs
   ```

   where x: is the required mapped drive on the target machine. Alternatively, use Windows NT Explorer or another method to map the shared resource to a drive letter.
6. Change to the installation directory (in this example x), type `Setup` and press Enter.
7. Follow the prompts.

Using the System Management Server with MQSeries for Windows NT and Windows 2000

This section describes how to install, or remove, an MQSeries for Windows NT and Windows 2000 client using the System Management Server (SMS).

Creating SMS Packages and Jobs for MQSeries

You must create:

- An SMS software package containing the MQSeries software
- An SMS job to distribute and install the software package

For more detailed information on how to create a software package and a job, refer to the Microsoft System Management Server documentation.

Creating the MQSeries SMS Software Package

To create the SMS software installation package:

1. From the Microsoft SMS Administrator application, open the **Packages** folder and create a new package.
2. In the SMS **Package Properties** dialog click on the **Import** button to create the software package by importing a Package Definition File (PDF).
3. In the **File Browser** dialog, select the drive where the IBM MQSeries client CD-ROM is located.
4. Select the Winnt directory, which contains the package definition file `MQSERIES.PDF`.

   You can also find the `MQSERIES.PDF` file in the local drive, or shared network drive where you copied the MQSeries Installation software.
5. Select the **MQSERIES.PDF** file and select **OK**.
6. Select **Workstation**. In the **Source Directory** entry field, specify the fully-qualified path name to the MQSeries root directory that contains the MQSeries installation software.
7. Select the appropriate Workstation Command Line:
   - **Automated Uninstallation of IBM MQSeries - Windows NT client**
   - **Automated Installation of IBM MQSeries - Windows NT client (US English)**
8. Select **Properties** for each process and review the **Command Line** entry field to ensure that the parameters are correct.

   Note: The `-i<miffilepath>` parameter specifies the full path and file name of a Management Information Format (.mif) file that installation and uninstallation can generate. Remove the parameter if you do not want to generate a .mif file.
9. Select **Close** to close the **Workstation Properties** dialog.

   Note: If you specified a local path in the **Source Directory** entry field, you get a pop-up dialog warning you that the local path you specified might not be accessible to SMS components running on another machine. Select**OK** to continue.
10. Select **OK** to close the **Package Properties** window.

    A pop-up dialog appears indicating that SMS will update the software package at all sites. Select**OK** to continue.

The software package has been created and can be installed by creating an SMS job.

Client installation (SMS)

Creating the MQSeries SMS Job

You must now create an SMS job to distribute and install the software packages you created that contain the MQSeries installation software.

Refer to the Microsoft System Management Server documentation for detailed information on how to create and run a job.

Notes:

1. You ***must*** be logged onto the target machine with Administrator authority in order to install the MQSeries Server.
2. When creating an SMS Job to distribute and install the software package, ensure that you select the appropriate workstation command. The workstation commands are displayed on the **Job Details** dialog in the **Run Phase** section and appear in a drop-down list box.

Performing an unattended installation

You can install the MQSeries for Windows NT and Windows 2000 client on a remote machine without interaction, provided that the remote machine can share the client CD-ROM, or a copy of the files on it, and that you can execute a command on the remote machine. This process is called unattended (or silent) installation, and is particularly useful for installing MQSeries clients over a network because you can do it from a shared drive on a LAN file server.

Because there is no user interaction, unattended installation uses a response file. A response file is an ASCII text file containing values for the installation options you want to select. For information on the format of response files, see the *MQSeries Clients* book.

There are three ways to generate a response file:

1. Edit the response file (setup.iss) supplied in the Winnt directory on the MQSeries for Windows NT client CD-ROM, using an ASCII file editor.
2. Generate your own response file using an ASCII file editor.
3. Carry out an installation on a machine and record the options selected to install the product in a response file. To do this you must run setup.exe with the -r parameter, and optionally, the -noinst parameter:
   a. Load the MQSeries client CD-ROM. If you have autorun enabled, the language selection panel appears. Cancel this panel.
   b. Run `setup.exe` from the root folder of the CD-ROM, with the `-r` parameter. This can be done by dragging `setup.exe` from Windows NT Explorer to the Run dialog in the start menu and adding `-r` to the end of the file name, or by typing the fully-qualified path to setup.exe into the Run dialog, followed by `-r`.

c. The language selection dialog reappears. Perform an installation as you want it performed on the remote machine.

d. If you are asked to restart the system, you do not need to do so if you are only generating a response file.

The -noinst parameter (which, if present, must precede the -r parameter) suppresses the MQSeries installation; Setup generates only a response file. The installation creates a response file called setup.iss in the Windows directory, normally c:\winnt. Save this file; you can edit it if necessary.

Use the response file you have created to install the product on a remote machine. (See "Installing on the remote machine".)

Installing on the remote machine

The remote machine must have access to a shared resource or drive on a file server that contains the client CD-ROM or a copy of it. Perform the procedure described in "Installing from a LAN" on page 47 up to the point where you run Setup. Copy your response file to a location on the file server that is accessible from the remote machine. You can now start the installation on the remote machine:

1. On the remote machine, go to the Setupxx folder on the shared resource.
2. Substitute Setupxx with the name of the language subfolder for the language that you require.
3. Run setup.exe:

   ```
   setup -g<logfile> -f1<responsefile> -f2<secondarylogfile> -s
   ```

   This installs in the language specified by Setupxx.

In the above command:

<logfile>
The full path to an installation log file. Setup creates a U.S. English ASCII text log file containing details of what happens during installation. You should check this file to see if any errors occurred. If you omit the -g<logfile> parameter, Setup creates a file called amqilogn.txt in the data-files folder on the machine running the installation. In this case, any messages generated before the data-files folder is created are lost. The -g<logfile> parameter must be placed before the -f1 and the -s parameters, otherwise it is ignored.

<responsefile>
The full path to the response file you prepared. If you omit the -f1<responsefile> parameter, the response file must be in the Setupxx language subdirectory.

<secondarylogfile>
The full path to a secondary installation log file. This file contains less detail than the other log file and should not be used as the primary source for information about the success of the installation. If you omit the -f2<secondarylogfile> parameter, Setup attempts to create a file called setup.log in the language subdirectory (Setupxx), which is not possible if the installation is being performed from a CD-ROM. The -f2<secondarylogfile> parameter must be placed after the -f1 parameter.

-s This parameter tells Setup to run in silent mode.

Enclose the long path name and file name expressions in double quotes.

Chapter 8. Configuring communications for the MQSeries client

This chapter tells you how to configure the MQSeries for Windows NT and Windows 2000 client and server communication links, and how to enable the server to listen for communications from the MQSeries client.

In MQSeries, the logical communication links are called *channels*. The channels used to connect MQSeries clients to servers are called MQI channels. You set up channel definitions at each end of your link so that your MQSeries application on the MQSeries client can communicate with the queue manager on the server.

Before you define your MQI channels:

1. Decide on the form of communication you are going to use.
2. Define the connection at each end:
   a. Configure the connection.
   b. Record the values of the parameters that you need for the channel definitions.
   c. Enable the server to detect incoming network requests from your MQSeries client. This involves starting a *listener*.

This chapter explains how to perform these tasks, using TCP/IP as an example.

When you define your MQI channels, each channel definition must specify a transmission protocol (transport type) attribute. A server is not restricted to one protocol, so different channel definitions can specify different protocols. For MQSeries clients, it might be useful to have alternate MQI channels using different transmission protocols.

Defining a TCP/IP connection

The steps to take are detailed in the sections that follow:

On the MQSeries client
Initialize TCP/IP.

On the server
There are three things to do:

1. Decide on a port number.

The port to connect to defaults to 1414. Port number 1414 is assigned by the Internet Assigned Numbers Authority to MQSeries.

2. Initialize TCP/IP, and record the network address of the server machine.
3. Configure files (or run a command) to specify the port number and to run a listener program.

For more detailed step-by-step examples, see the *MQSeries Intercommunication* manual.

Defining TCP/IP on the client

Initialize TCP/IP.

The channel definitions that you create later will include the network address and port number of the server to which the MQSeries client is sending.

Defining TCP/IP on the server

TCP/IP is initialized automatically as a service during Windows NT and Windows 2000 startup. However, you also need to start a *listener*, which enables receiver channels to start automatically in response to a request from an inbound sender channel.

To run the listener supplied with MQSeries for Windows NT and Windows 2000 (which starts new MQI channels as threads), use the RUNMQLSR command. For example:

```
RUNMQLSR -t tcp [-m QMNAME] [-p 1822]
```

The square brackets indicate optional parameters:
`-m QMNAME` is not required for the default queue manager.
`-p 1822` is not required if the default port number 1414 is used.

It is possible to have more than one queue manager running on the server machine. Start a listener program for each one, on different ports. For example:

```
RUNMQLSR -t tcp
RUNMQLSR -t tcp -m QM2 -p 1415
```

Chapter 9. Verifying a client installation

You can verify your MQSeries client and server installation using the supplied sample PUT and GET programs. These verify that your installation has been completed successfully and that the client and server can communicate.

This chapter explains how to use the supplied sample PUT and GET programs to verify that an MQSeries client has been installed correctly, by guiding you through the following tasks:

1. "Setting up the MQSeries server" on page 56
2. "Setting up the MQSeries client" on page 56
3. "Putting a message on the queue" on page 57
4. "Getting the message from the queue" on page 58
5. "Ending verification" on page 58

These instructions assume that the full MQSeries product (Base Product and Server) has been installed on a server machine, and that the MQSeries client software has been installed on a client machine.

The transmission protocol used in the example is TCP/IP. It is assumed that you have TCP/IP configured on the server and the MQSeries client machines, and that it has been initialized on both the machines. There is more information about this in "Chapter 8. Configuring communications for the MQSeries client" on page 53.

Compiled samples AMQSPUTC and AMQSGETC are included in the MQSeries client directories that you installed.

The following sections provide step-by-step instructions for creating a queue manager called *queue.manager.1*, a local queue called *QUEUE1*, and a server-connection channel called *CHANNEL1* on the server. They show how to create the client-connection channel on the MQSeries client workstation, and how to use the sample programs to put a message onto a queue, and then get the message from the queue.

Note: MQSeries object definitions are case-sensitive. You must type the examples ***exactly*** as shown.

Setting up the MQSeries server

Create a directory to hold working files, for example `mqverify`, and make this the current directory. Then follow the steps below to set up the server workstation. Before you can verify the client installation, you need to:

1. Create a default queue manager (called *queue.manager.1*) by entering the following command at the command prompt:

   ```
   crtmqm -q queue.manager.1
   ```

2. Start the queue manager by entering the following command:

   ```
   strmqm
   ```

3. Start MQSeries commands (MQSC) by entering the following command:

   ```
   runmqsc
   ```

   MQSC does not provide a prompt, but should respond with the message `Starting MQSeries Commands.`

4. Create a local queue by entering the following command:

   ```
   DEFINE QLOCAL(QUEUE1)
   ```

5. Create a server-connection channel by entering the following command:

   ```
   DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) MCAUSER(' ')
   ```

6. Stop MQSC by typing `end` and then pressing Enter.
7. Start a listener by entering the following command at the command prompt:

   ```
   RUNMQLSR -t tcp -m queue.manager.1
   ```

8. The server is now ready to communicate with the client.

Setting up the MQSeries client

When an MQSeries application is run on the MQSeries client, the information it requires is the name of the MQI channel, the communication type, and the address of the server to be used. You provide this by defining a client-connection channel. The name used must be same as the name used for the server-connection channel defined on the server. In this example the MQSERVER environment variable is used to define the client-connection channel. This is the simplest method (but not the only one).

Before starting, type `ping server-address` (where `server-address` is the TCP/IP hostname of the server) to confirm that your MQSeries client and server TCP/IP sessions have been initialized. You can use the network address, in the format n.n.n.n, in the `ping` command instead of the hostname.

If the `ping` command fails, check that your TCP/IP software is correctly configured and has been started.

Defining a client-connection channel, using MQSERVER

Create a client-connection channel by setting the MQSERVER environment variable using the following command:

```
SET MQSERVER=CHANNEL1/TCP/server-address(port)
```

Where:

`server-address` is the TCP/IP hostname of the server.
`(port)` is optional and is the TCP/IP port number the server is listening on.

If you do not give a port number, MQSeries uses the one specified in the Windows NT or Windows 2000 registry. If no value is specified in the registry, MQSeries uses the port number identified in the TCP/IP services file for the service name `MQSeries`. If this entry in the services file does not exist, a default value of 1414 is used. It is important that the port number used by the client and the port number used by the server listener program are the same.

Putting a message on the queue

On the MQSeries client workstation, put a message on the queue using the AMQSPUTC sample program:

1. From a command prompt window, change to the directory containing the sample program amqsputc.exe. This is in the `\bin` directory. Then enter the following command:

   ```
   amqsputc QUEUE1 qmgr
   ```

   where `qmgr` is the name of the queue manager on the server.
2. The following message is displayed:

   ```
   Sample AMQSPUT0 start
   target qname is QUEUE1
   ```
3. Type some message text and then press Enter twice.
4. The following message is displayed:

   ```
   Sample AMQSPUT0 end
   ```
5. The message is now on the queue on the server queue manager.

Getting the message from the queue

On the MQSeries client workstation, get a message from the queue using the amqsgetc sample program:

1. Change to the directory containing the sample programs, and then enter the following command:

   ```
   amqsgetc QUEUE1 qmgr
   ```

   Where qmgr is the name of the queue manager on the server.
2. The message on the queue is removed from the queue and displayed.

Ending verification

The verification process is now complete.

You can stop the queue manager on the server by typing the following command on the server machine:

```
endmqm queue.manager.1
```

If you want to delete the queue manager on the server type:

```
dltmqm queue.manager.1
```

Now that verification is complete, see "Part 3. Getting started with MQSeries" on page 65.

Chapter 10. Applying maintenance

Maintenance updates in the form of a Program Temporary Fix (PTF) are supplied on CD-ROM, referred to as a Corrective Service Diskette (CSD). They can also be downloaded from:

```
http://www.ibm.com/software/mqseries/
```

Attention

- Do not have queue managers running when you install maintenance on MQSeries. End each queue manager that is running by issuing the command:

  ```
  endmqm -i QMgrName
  ```

  and check that the queue manager is not running.
- Do not have channel listeners running when you install maintenance on MQSeries. To end all running listener processes for a queue manager:
  1. Check that the queue manager has been stopped.
  2. End all listener processes by issuing the command:

     ```
     endmqlsr -m QMgrName
     ```
- Stop the MQSeries service by right-clicking on the MQSeries icon in the taskbar, and clicking **Stop IBM MQSeries**.

Applying the maintenance information

If you need to apply maintenance updates to your MQSeries product you must ensure that you are logged on with Administrator authority. You can install the updates either from the MQSeries Web site (see "Installing updates from the MQSeries Web site") or from CD-ROM (see "Installing updates from CD-ROM" on page 60).

Installing updates from the MQSeries Web site

To install maintenance updates from the MQSeries Web site:

1. Select a destination directory for the supplied executable file.
2. Change to the destination directory when the file has been downloaded and run the executable file. Running this file presents you with a dialog

screen on which you can choose to use the default temporary directory, or specify your own temporary directory into which to unload the executable file.

3. Select the default directory, or change it if required, and click **Next**.
4. Click **Finish** when the file has been unloaded into the temporary directory to end the dialog.

   The SETUP.EXE file now runs.
5. To view the MEMO.PTF file (which contains details of the maintenance applied), select:

   Start → Programs → IBM MQSeries → MQSeries Service Level

Installing updates from CD-ROM

If you are installing from CD-ROM:

1. Insert the MQSeries maintenance CD-ROM into the appropriate drive.
2. The **Welcome** dialog is displayed. Click **Next**.
3. Select the backup directory, and then click **Next**.

   Note: You are strongly advised to use the default directory for the backup operation.
4. From the **Copy Files** dialog, verify that the information displayed matches your choices.

   If the information is incorrect, click **Back** to return to the previous dialogs. Otherwise, click **Next** to proceed.
5. Click **Finish** to complete the installation of the CSD.
6. To view the MEMO.PTF file (which contains details of the maintenance applied), select:

   Start → Programs → IBM MQSeries → MQSeries Service Level

Restoring the previous backup version

If you need to restore MQSeries to a previous level of maintenance:

1. Ensure that you are logged on as an Administrator.
2. Ensure that all queue managers are stopped.
3. Ensure that all channel listeners are stopped.
4. Ensure that the IBM MQSeries Service has been stopped.
5. Select **IBM MQSeries** from the **Start** menu and then click **Remove Latest CSD**.
6. The CSD history log is displayed, and you are given the option to rollback the most recent CSD.

   Selecting the rollback option returns the installation to the state it was in before the CSD was applied.

Querying the service level

After initial installation, the service level has the value `0000520`.

After one or more updates, the service level indicates from which CSD the product was most recently updated. The service level is expressed in terms of the PTF number for a particular CSD. To query the service level, click **MQSeries Service Level** from the **Start** menu. By default this is by selecting the following:

Start → Programs → IBM MQSeries → MQSeries Service Level

This shows the MEMO.PTF file containing the service level and details of the maintenance applied (PTF number). This file is installed in the program file directory.

Chapter 11. Removing MQSeries

You can uninstall (remove) MQSeries in attended mode or unattended (silent) mode.

Before you uninstall MQSeries, ensure that there are no MQSeries programs running. To do this, right-click on the MQSeries icon in the task bar and click **Stop IBM MQSeries**, close all MQSeries windows, hide the task bar, and stop any monitoring service.

Performing an attended removal

You can start attended removal in either of the following ways:

- From the MQSeries program folder:
  1. Open the MQSeries program folder created during installation (by default, this folder is called IBM MQSeries).
  2. Select the MQSeries Uninstallation icon to run the uninstall program.

  or
- By selecting **IBM MQSeries V5.2** from the Add/Remove programs icon in the Control Panel.

When the uninstallation program is running, you must select one of:

- Uninstall one or more components:
- Uninstall all of MQSeries, excluding data
- Uninstall all of MQSeries, completely

If a message appears telling you that locked files have been found, stop MQSeries (by right-clicking on the MQSeries icon in the task bar and clicking **Stop IBM MQSeries**), close all MQSeries windows, hide the task bar, and stop any monitoring service. You should now be able to continue with the uninstallation.

Uninstalling MQSeries excluding data leaves your queue managers and their queues, and any Web Administration Server scripts intact. If you later decide to reinstall MQSeries, you can get the previous data by explicitly specifying the previous data directory. If you decide to uninstall one or more components, you must choose from a list of which components to uninstall. If you choose to uninstall the Server or the Web Administration Server components, you are asked whether you want to keep the data associated with each of those components.

Performing an unattended removal

You can remove (uninstall) MQSeries on a remote machine without interaction. This process is called unattended (or silent) removal, and uses a response file. A response file is an ASCII text file containing values for the options you select when you uninstall an MQSeries system. For more information on using response files, see "Appendix B. Using response files for installing and removing MQSeries" on page 111.

For unattended uninstallation, you can:

- Edit the response file (amqiunin.rsp) supplied on the MQSeries Server CD-ROM, using an ASCII file editor

  or

- Generate your own response file, using an ASCII file editor

When you have generated your response file, run the uninstall program amqiunin.exe directly, with two additional parameters, as follows:

```
c:\program files\mqseries\uninst\amqiunin.exe -g<logfile> -f1<responsefile> -s
```

where `<responsefile>` is the fully-qualified path to your response file. If this parameter is omitted, the response file must be called `amqiunin.rsp` and must be located in the same folder as amqiunin.exe. The `-s` parameter tells the uninstall program to run in silent mode.

Removing an MQSeries Client from Windows NT or Windows 2000

If you want to remove the MQSeries client files from your machine, use Settings/ Control Panel/ Add-Remove. First select IBM MQSeries Client, which launches the uninstall program. Alternatively, choose MQSeries Uninstallation from the IBM MQSeries client folder. You can choose to uninstall one or more components or the whole of the MQSeries client.

For more information on uninstalling an MQSeries for Windows NT client, see the *MQSeries Clients* book.

Part 3. Getting started with MQSeries

Chapter 12. About MQSeries

This chapter introduces IBM MQSeries. It describes its basic functions and its relationships with operating systems, applications, and other middleware products.

Introduction

MQSeries is a communications system that provides assured, asynchronous, once-only delivery of data across a broad range of hardware and software platforms.

These characteristics make MQSeries the ideal infrastructure for application-to-application communication, and make it an appropriate solution whether the applications run on the same machine or on different machines that are separated by one or more networks.

MQSeries supports all the important communication protocols and even provides routes between networks that use different protocols. MQSeries bridges and gateway products allow easy access (with little or no programming) to many existing systems and application environments—for example, Lotus Notes, Web browsers, Java applets, and many others.

The assured delivery capability reflects the many functions built in to MQSeries to ensure that data is not lost because of failures in the underlying system or network infrastructure. Assured delivery enables MQSeries to form the backbone of critical communication systems and to be entrusted with delivering high-value data. There are also options that allow you to select a less robust quality of service, where this is appropriate. For example, there might be circumstances where you might prefer faster delivery with less emphasis on assured delivery.

The asynchronous processing support in MQSeries means that the exchange of data between the sending and receiving applications is time independent. This allows the sending and receiving applications to be decoupled so that the sender can continue processing, without having to wait for the receiver to acknowledge that it has received the data. In fact, the target application does not even have to be running when the data is sent. Likewise, the entire network path between the sender and receiver may not need to be available when the data is in transit.

Once-only delivery of data is a vital consideration, particularly in financial and business applications where duplicate requests to move large sums of money from one account to another are precisely what you do not want to happen!

Messages, queues, and queue managers

The three fundamental concepts in MQSeries that you need to understand are:

- Messages
- Queues
- Queue managers

Messages

A *message* is a string of bytes that has meaning to the applications that use it. Messages are used for transferring data from one application to another (or to different parts of the same application). The applications can be running on the same platform, or on different platforms.

MQSeries messages have two parts; the *application data* and a *message descriptor*. The content and structure of the application data is defined by the application programs that use the data. The message descriptor identifies the message and contains other control information, such as the type of message and the priority assigned to the message by the sending application.

Queues

A *queue* is a data structure in which messages are stored. The messages may be put on, or got from, the queue by applications or by a queue manager as part of its normal operation.

Queues exist independently of the applications that use them. A queue can exist in main storage (if it is temporary), on disk or similar auxiliary storage (if it must be kept in case of recovery), or in both places (if it is currently being used, and must also be kept for recovery). Each queue belongs to a *queue manager*, which is responsible for maintaining it. The queue manager puts the messages it receives onto the appropriate queue.

Queues can exist either in your local system, in which case they are called *local queues*, or at another queue manager, in which case they are called *remote queues*.

Applications send to, and receive messages from, queues. For example, one application can put a message on a queue, and another application can get the message from the same queue.

Each queue has *queue attributes* that determine what happens when applications reference the queue. The attributes indicate:

- Whether applications can retrieve messages from the queue (get enabled)
- Whether applications can put messages onto the queue (put enabled)
- Whether access to the queue is exclusive to one application or shared between applications
- The maximum number of messages that can be stored on the queue at the same time (maximum queue depth)
- The maximum size of messages that can be put on the queue (maximum message size)

Queue managers

A queue manager provides queuing services to applications, and manages the queues that belong to it. It ensures that:

- Object attributes are changed according to the details received.
- Special events (such as instrumentation events or triggering) are generated when the appropriate conditions are met.
- Messages are put on the correct queue, as requested by the application. The application is informed if this cannot be done, and an appropriate reason code is given.

Each queue belongs to a single queue manager and is said to be a *local queue* to that queue manager. The queue manager to which an application is connected is said to be the local queue manager for that application. For the application, the queues that belong to its local queue manager are local queues. A *remote queue* is a queue that belongs to another queue manager. A *remote queue manager* is any queue manager other than the local queue manager. A remote queue manager may exist on a remote machine across the network or it may exist on the same machine as the local queue manager. MQSeries supports multiple queue managers on the same machine.

MQSeries configurations

In the simplest configurations, MQSeries is installed on a machine and a single queue manager is created. This queue manager then allows you to define queues. Local applications can then use these queues to exchange messages.

Communication by applications with queues managed by another queue manager requires *message channels* to be defined. It is not necessary to define a channel directly to the target queue manager and it is often appropriate to define one only to the next hop (that is, an intermediate queue manager). Message channels available from that queue manager will be used to deliver the message to the target queue manager (or even to a subsequent hop).

More complex configurations can be created using a client-server structure. The MQSeries product can act as an MQSeries server to MQSeries clients. The clients and server do not need to be on the same platform. MQSeries supports a broad range of client platforms. The MQSeries products typically include clients for a variety of platforms. Additional MQSeries clients are available from the MQSeries Web site.

In a client-server configuration, the MQSeries server provides messaging and queuing services to the clients, as well as to any local applications. The clients are connected to the server through dedicated channels (known as *client channels*) for clients. This is a cost-effective deployment method because a server can support hundreds of clients with only a single copy of the MQSeries server product. However, the client channel must be continuously available whenever the MQSeries applications on the client are running. This contrasts with the message channels, which need not be continuously available to support MQSeries applications running on the server.

See “Channels” for more information.

MQSeries also supports the concept of *clusters* to simplify setup and operation. A cluster is a named collection of queue managers and any one queue manager can belong to none, one, or several such clusters. The queue managers in a cluster can exist on the same or different machines.

The default configuration that is provided by the installation process for MQSeries for Windows NT and Windows 2000 can, if there is no other queue manager already defined, configure a queue manager that is joined to the local default cluster. The queue manager has a name that is based on the TCP/IP domain of the machine.

There are two major benefits from the use of clusters:

1. Communication between members of a cluster is greatly simplified, particularly because the channels required for exchanging messages are automatically defined and created as needed.
2. Some or all of the queues of participating queue managers can be defined as being cluster queues, which has the effect of making them automatically known and available to all other queue managers in the cluster.

See “Clusters” on page 71 for more information.

Channels

A channel provides a communication path to a queue manager. There are two types of channel: message channels and MQI channels.

A *message channel* provides a communication path between two queue managers on the same, or different, platforms. The message channel is used

for transmitting messages from one queue manager to another, and shields the application programs from the complexities of the underlying networking protocols. A message channel can transmit messages in one direction only. Two message channels are required if two-way communication is required between two queue managers.

A *client channel* (also known as an *MQI channel*) connects an MQSeries client to a queue manager on a server machine and is bidirectional.

If you want to read more information about channels and how MQSeries uses them to communicate across the systems in your network, see the *MQSeries Intercommunication* book.

Clients and servers

MQSeries supports client-server configurations for MQSeries applications.

An *MQSeries client* is a part of the MQSeries product that is installed on a machine to accept MQSeries calls from applications and pass them to an *MQSeries server* machine. There they are processed by a queue manager. Typically, the client and server reside on different machines, but they can also exist on the same machine.

An *MQSeries server* is a queue manager that provides queuing services to one or more clients. All the MQSeries objects (for example, queues) exist only on the queue manager machine (that is, on the MQSeries server machine). A server can support local MQSeries applications as well.

The difference between an MQSeries server and an ordinary queue manager is that the MQSeries server can support MQSeries clients, and each MQSeries client application has a dedicated communication link with the MQSeries server.

For more information about client support, see the *MQSeries Clients* book.

Clusters

A cluster is a named collection of queue managers.

Clusters require that at least one of the queue managers in the cluster be defined as a *repository* (that is, a place where the shared cluster information can be held). More typically, two or more such repositories are usually designated to provide continued availability in the case of system failure. MQSeries makes sure that the information in the repositories is synchronized.

When a queue is defined as a cluster queue, it can be regarded as a public queue in that it is freely available to other queue managers in the cluster. This contrasts with noncluster queues, which are accessible only when a local

definition of them is available. Thus, a noncluster queue has the characteristics of a private queue, accessible only to those queue managers that have been configured to know about them.

Public queues with the same name in the same cluster are regarded as equivalent. If a message is sent to that queue name, MQSeries (by default) sends it to any one of the instances, using a load-balancing algorithm. If you do not want this to happen, you can use the queue manager and queue name in the address, thus forcing the message to be delivered to a specific queue manager. Alternatively, you can replace the load-balancing routine with a different implementation. This is typical of MQSeries, in that there are many examples of where standard behavior can be changed by implementing user code in exits designed for this purpose.

You can read a full explanation in the *MQSeries Queue Manager Clusters* book.

MQSeries capabilities

MQSeries can be used to create many different types of solutions. Some exploit the platform support, or the bridge and gateway capabilities, to connect existing systems in an integrated way or to allow new applications to extract information from, or interchange information with, existing systems. Other solutions support business application servers, where a central pool of MQSeries applications can manage work sent across networks. Complex routing of information for workflow scenarios can be supported. Publish/subscribe or "send and forget" are other application scenarios that use different message flows. Load balancing and hot-standby systems can be built using the power and flexibility of MQSeries, which includes specific functions to support many of these diverse scenarios.

See the *MQSeries Application Programming Guide* for more information about writing MQSeries applications.

Transactional support

An application program may need to group a set of updates into a *unit of work*. Such updates are usually logically related and must all be successful for data integrity to be preserved. Data integrity would be lost if one update in the group succeeded while another failed. MQSeries supports transactional messaging.

A unit of work *commits* when it completes successfully. At this point all updates made within that unit of work are made permanent and irreversible. Alternatively, all updates are *backed out* if the unit of work fails. *Syncpoint coordination* is the process by which a unit of work is either committed or backed out with integrity.

A *local* unit of work is one in which the only resources updated are those of the MQSeries queue manager. Here, syncpoint coordination is provided by the queue manager itself, using a single-phase commit process.

A *global* unit of work is one in which resources belonging to other resource managers, such as XA-compliant databases, are also updated. Here, a two-phase commit procedure must be used and the unit of work may be coordinated by the queue manager itself, or externally by another XA-compliant transaction manager such as IBM CICS®, IBM Transaction Server, IBM TXSeries™, Transarc Encina, or BEA Tuxedo.

When the queue manager coordinates global units of work itself it becomes possible to integrate database updates within MQSeries units of work. That is, a mixed MQSeries and SQL application can be written, and commands can be used to commit or roll back the changes to the queues and databases together.

The queue manager achieves this using a two-phase commit protocol. When a unit of work is to be committed, the queue manager first asks each participating database manager whether it is prepared to commit its updates. Only if all of the participants, including the queue manager itself, are prepared to commit, are all of the queue and database updates committed. If any participant cannot prepare its updates, the unit of work is backed out instead.

Full recovery support is provided if the queue manager loses contact with any of the database managers during the commit protocol. If a database manager becomes unavailable while it is in doubt (that is, it has been called to prepare but has yet to receive a commit or backout decision), the queue manager remembers the outcome of the unit of work until it has been successfully delivered. Similarly, if the queue manager terminates with incomplete commit operations outstanding, these are remembered when the queue manager restarts.

Instrumentation events

You can use MQSeries instrumentation events to monitor the operation of queue managers.

Instrumentation events cause special messages, called *event messages*, to be generated whenever the queue manager detects a predefined set of conditions. For example, a *Queue Full* event message is generated if: Queue Full events are enabled for a specified queue; an application issues an MQPUT call to put a message on that queue; and the call fails because the queue is full.

Other conditions that can give rise to instrumentation events include:
- A predefined limit for the number of messages on a queue being reached
- A queue not being serviced within a specified time

- A channel instance being started or stopped

If you define your event queues as remote queues, you can put all the event queues on a single queue manager (for those nodes that support instrumentation events). You can then use the events generated to monitor a network of queue managers from a single node.

MQSeries instrumentation events are categorized as follows:

Queue manager events
These are related to the definitions of resources within queue managers. For example, if an application attempts to open a queue but the associated user ID is not authorized to perform that operation, a queue manager event is generated.

Performance events
These are notifications that a threshold condition has been reached by a resource. For example, a queue depth limit has been reached or, following an MQGET request, a queue has not been serviced within a predefined period of time. You can use the Windows NT performance monitor to check and control certain aspects of your queues (for example, the queue depth, percentage queue depth, and how many messages have been enqueued and dequeued).

Channel events
These are reported by channels as a result of conditions detected during their operation. For example, a channel event is generated when a channel instance is stopped.

Message-driven processing

When they arrive on a queue, messages can automatically start an application, using a mechanism known as *triggering*. If necessary, the application can be stopped when the message or messages have been processed.

Programming MQSeries

MQSeries applications can be developed using a variety of programming languages and styles. Procedural and object-oriented programming is supported, depending on the MQSeries platform, using, for example, Visual Basic®, C, C++, Java, COBOL, and PL/I. Microsoft Windows NT ActiveX/COM technology is also supported.

MQSeries function is logically divided into what is normally required by applications (such as putting messages on a queue) and what is necessary for administration (such as changing queue or queue manager definitions). Application function is known as the *MQI* (message queue interface).

Administration function is known as the *MQAI* (message queuing administration interface). Applications can mix MQI and MQAI functionality, as required.

The administration functions can be implemented in two ways:

1. Most often, using MQAI language bindings or ActiveX classes
2. Sending messages to administration queues, to achieve the same results as with the MQAI, using programmable command formats (PCFs)

Managing MQSeries

MQSeries is tightly integrated into the Windows NT operating system. The management facilities are, therefore, normally to be found alongside their Windows NT equivalents and are accessible through a graphical user interface. For example, MQSeries events are written to the Windows NT event log and are accessed through the event manager. Likewise, performance monitoring or checking the status of queues is available through the Windows NT performance monitor.

The operational side of MQSeries can be controlled through the Microsoft Management Console (MMC). The MQSeries Explorer, provided as an MMC snap-in, allows you to browse queues and change their properties. Both local and remote administration are available; thus, a single MQSeries installation on a Windows NT platform can be used to manage an entire MQSeries network.

A second MMC snap-in, MQSeries Services, allows you to control the operation of MQSeries and its various subcomponents (such as stopping and starting facilities, default configurations, and other administration actions).

For compatibility with previous levels of MQSeries, an alternative management method is available that uses programs and a command language known as MQSC. These can be accessed from a command prompt. Details of MQSC are given in the *MQSeries MQSC Command Reference*.

You can also use a Web browser to manage MQSeries. MQSeries includes an optional Web server that supports secure management of an MQSeries network from a browser. It is only necessary to install the Web server on one MQSeries machine (although it can, optionally, be installed on many machines within a network). The MQSeries Web Administration facility provides a scripting capability that allows automatic generation of MQSC.

Chapter 13. Using MQSeries

If MQSeries has been installed with the standard (default) settings, the MQSeries service starts automatically when the machine on which it is installed starts up. An MQSeries icon appears in the task bar, and color of the figure in the icon indicates whether MQSeries is running or not: when the figure is green MQSeries is running, and when the figure is red MQSeries has stopped. You can start and stop MQSeries by right-clicking on the icon in the task bar and then clicking on either **Start IBM MQSeries** or **Stop IBM MQSeries**.

It is the system administrator's job to monitor MQSeries and make any changes that might be necessary. To do this, you need to know where each MQSeries object resides, what its characteristics are, and who has access to it.

You can manage and monitor MQSeries resources by using:

- The MQSeries MMC snap-ins
  - IBM MQSeries Explorer
  - IBM MQSeries Services
- The MQSeries Web Administration server

MQSeries snap-ins

MQSeries provides snap-ins that run under the Microsoft Management Console (MMC):

- IBM MQSeries Explorer, described in "MQSeries Explorer" on page 78
- IBM MQSeries Services, described in "MQSeries Services" on page 79

These snap-ins provide a graphical user interface for administering the elements of your MQSeries network, allowing you to define and control:

- Queue managers
- Queues
- Clusters (networks of queue managers that can be on several MQSeries systems)
- Other MQSeries objects (such as channels, processes, client connections, and namelists)
- Services, which allow you to start and stop functions and to associate actions with the functions

The user interface provides extensive help information to guide you through the tasks involved. The MQSeries Information Center also provides a great deal of useful information.

MQSeries Explorer

You can get to the MQSeries Explorer from:

- The Windows NT or Windows 2000 Start menu, selecting MQSeries Explorer from the IBM MQSeries menu

or

- The MQSeries First Steps window, selecting MQSeries Explorer

You will then see a window similar to the following:

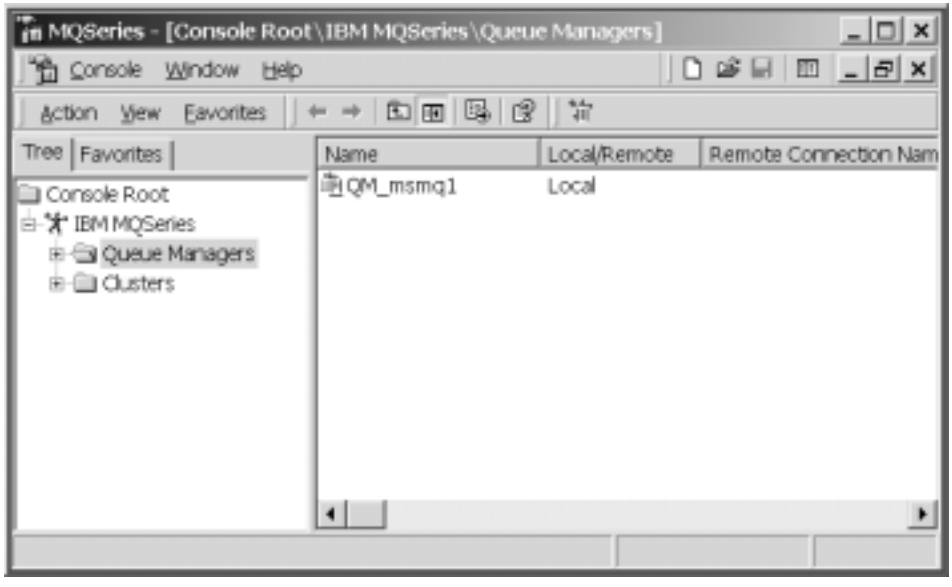

Figure 12. MQSeries Explorer

You can use the MQSeries Explorer to:

- Create, delete, or change any of the MQSeries objects (for example: queue managers, clusters, namelists, queues, channels, client connections, and processes)
- Start or stop a queue manager
- View queue managers and their objects on this or other computers
- Check the status of queue managers, clusters, and channels on this or other computers
- Browse messages on queues

Your user ID ***must*** belong to the local `mqm` or `Administrators` group in order to administer any queue manager on that system.

If the user ID under which you are running the MQSeries Explorer for your local system is a member of the `mqm` or `Administrators` group on a remote MQSeries system, you will be able to use the MQSeries Explorer to administer queue managers on the remote system.

MQSeries Services

You can get to the MQSeries Services from:

- The Windows NT or Windows 2000 Start menu, selecting MQSeries Services from the IBM MQSeries menu

or

- The MQSeries Explorer window, by right clicking on a queue manager name

or

- The IBM MQSeries icon in the taskbar menu and selecting MQSeries Services from the menu

You will then see a window similar to the following:

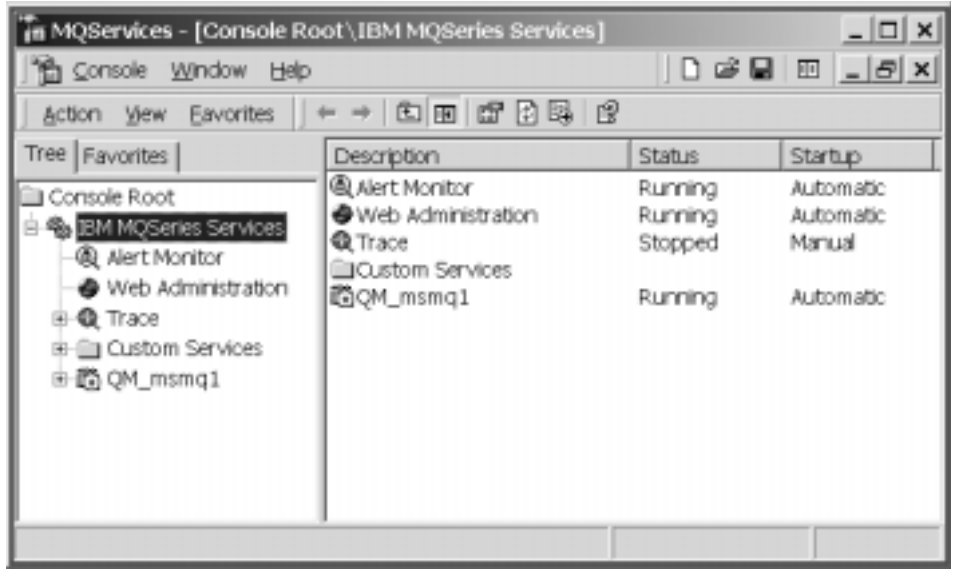

Figure 13. MQSeries Services

MQSeries snap-ins

You can use the MQSeries Services to:

- Start, stop, create, or delete a queue manager
- Start or stop the Web Administration server
- Start or stop individual services of a queue manager (for example: command server, channel initiator, and listener)
- Set up the configuration of the services (for example: automatic startup, failure recovery actions, and communication protocols)
- View the alert monitor
- Start and stop service trace
- Specify applications to be started automatically when MQSeries is started (see the *MQSeries System Administration* manual for more information on the Custom Service)

Web Administration

MQSeries also provides a Web-based application that allows you to administer all the systems in your MQSeries network from a Windows NT or Windows 2000 workstation. The application allows you to use MQSeries command (MQSC) facilities, either as individual commands or multiple commands in a script. The MQSeries Web Administration server can be started and stopped from the MQSeries Services. The server is configured to start automatically when installed (it is not installed by default).

To log on as an MQSeries administrator (client side), connect your Web browser to the MQSeries Web Administration Web server by using a URL of the form:

`http://<hostname>:<port_number>`

where:

<hostname>
Is the IP host name (or numeric TCP/IP address) of the computer running the Web server.

<port_number>
Is the IP port number assigned to the Web server. The default value for `port_number` is 8081. However, this value may be changed using the MQSeries Services. For example:

```
http://strugnel.hursley.ibm.com:8081/
```

or

```
http://9.20.20.92:8081/
```

This URL must be made known to all MQSeries administrators who will be using MQSeries Web Administration.

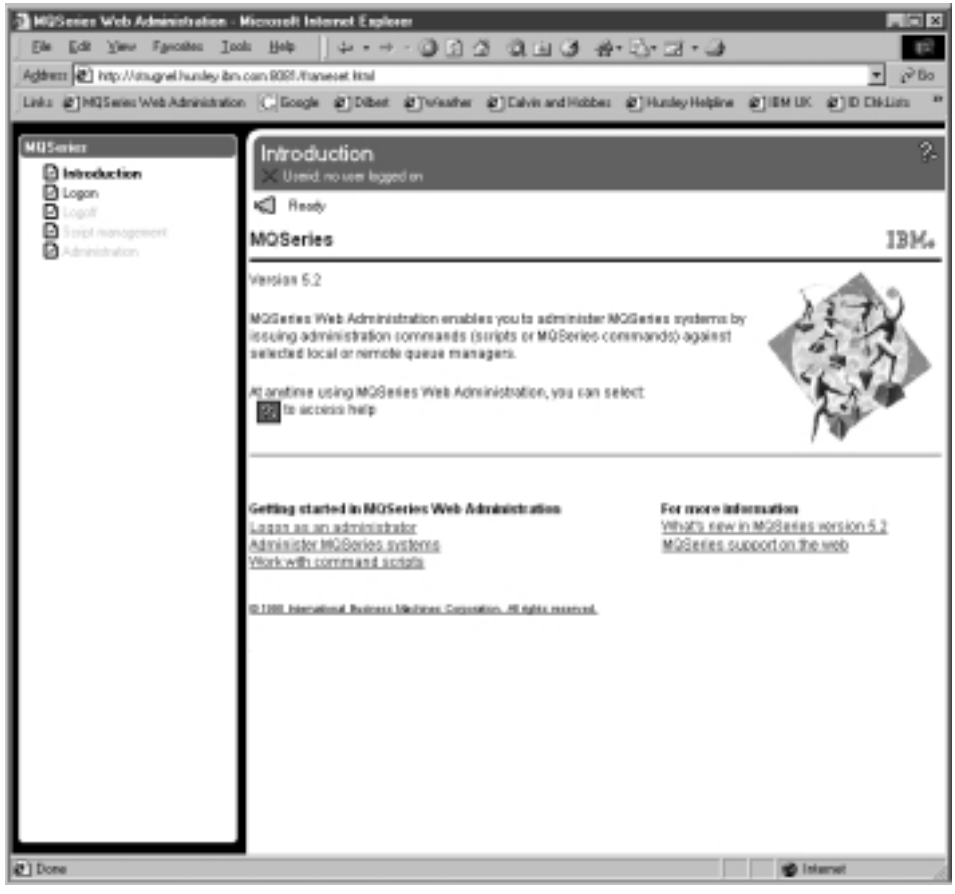

Figure 14. Web Administration

Note: If you use Netscape as your browser, you may find that the response time is relatively slow — this is a known problem with Netscape when it is run on the same system as the MQSeries server. To avoid this problem either use Internet Explorer, or run the browser and MQSeries server on different machines.

The left-hand pane of the browser window contains a navigation area. To log on as an MQSeries Web Administrator and administer MQSeries objects:

- Click **Logon** in the navigation area in the left-hand pane of the browser window. The Logon panel is displayed.
- Use the Logon panel to enter your Windows NT or Windows 2000 user ID and password for MQSeries Web Administration.

- Click **Logon** to start the logon process. The Administration panel is displayed.

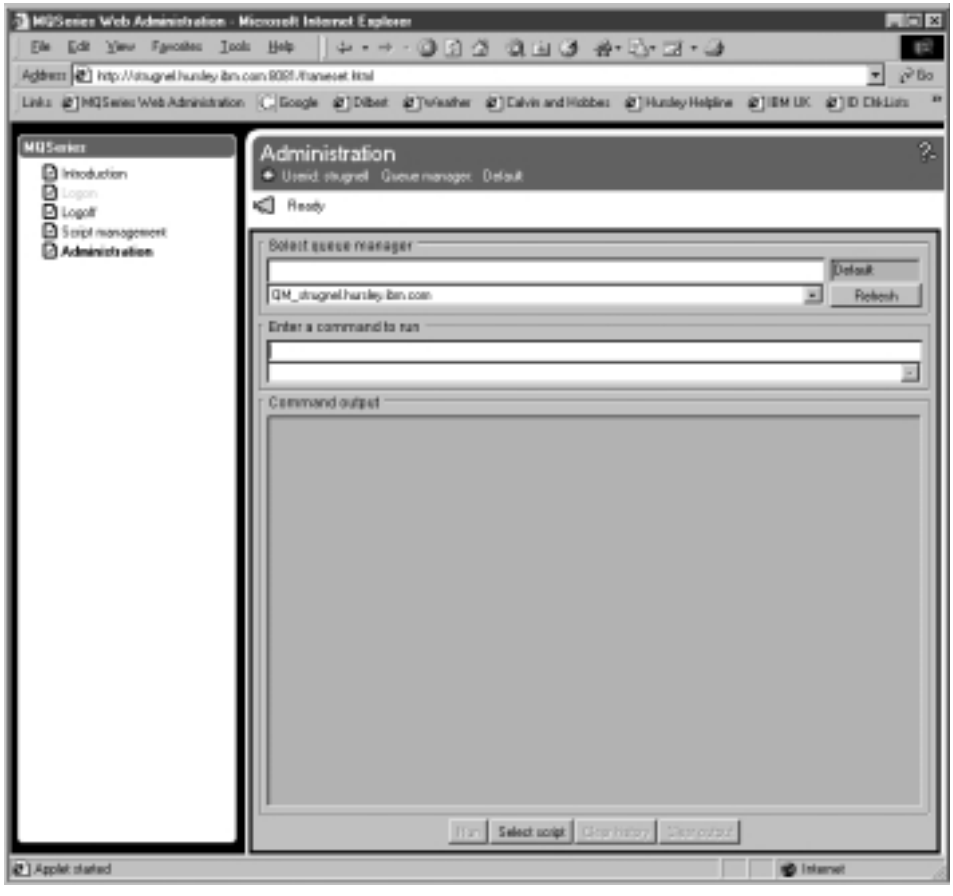

Figure 15. Administration panel

- Select the queue manager that you want to work with from the **Select Queue Manager** list. The default queue manager is selected by default.
- Enter the command that you want to run and click **Run**.

Your user ID needs the necessary administration privileges on the MQSeries server to perform administration tasks. Therefore, before attempting to log on to MQSeries Web Administration, ensure that you have the correct level of authorization. This means being one or more of the following:

- A member of the mqm group
- A member of the administrator group on the machine running MQSeries Web Administration
- Logged on using the SYSTEM ID

Some operations may require you to have authorization to use individual objects or object types. MQSeries Web Administration uses existing MQSeries rules for security to ensure that this happens.

MQSeries Web Administration controls remote queue managers by using MQSeries commands (MQSC). The Web Administration server adopts the user ID of each logged-on administrator prior to invoking MQSC commands on the administrator's behalf. Therefore, administrators have exactly the same privileges from MQSeries Web Administration as they would have using the **runmqsc** command locally on the Web Administration server.

Full details on MQSeries Web Administration are given in the *MQSeries System Administration* book.

Chapter 14. Using MQSeries command sets

This chapter introduces the command sets that can be used to perform system administration tasks on MQSeries objects.

Administration tasks include creating, starting, altering, viewing, stopping, and deleting MQSeries objects such as queue managers, queues, processes, and channels. To perform these tasks, you must select the appropriate command from one of the supplied command sets.

Note: The MQSeries Explorer (see "MQSeries Explorer" on page 78) provides a graphical user interface which allows you to do many of the tasks that can be done by using commands. See Table 3 on page 86 for details.

MQSeries provides three command sets for performing administration tasks:

- Control commands
- MQSC commands
- PCF commands

This section describes the command sets that are available. Some tasks can be performed using either a control command or an MQSC command, whilst other tasks can be performed using only one type of command. For a comparison of the facilities provided by the different types of command set, see the *MQSeries System Administration* manual.

Control commands

MQSeries provides control commands that you can enter through the Windows NT command line. Your user ID ***must*** belong to the local `mqm` or `Administrators` group in order run any of the MQSeries for Windows NT and Windows 2000 control commands.

Control commands fall into three categories:

- *Queue manager commands*, including commands for creating, starting, stopping, and deleting queue managers and command servers.
- *Channel commands*, including commands for starting and ending channels and channel initiators.
- *Utility commands*, including commands associated with authority management and conversion exits.

Using control commands

Under MQSeries, you enter control commands at a command prompt. Control commands and their flags are not case sensitive, but arguments to those commands (such as queue names and queue-manager names) are case sensitive. For example, in the command:

```
crtmqm -u SYSTEM.DEAD.LETTER.QUEUE jupiter.queue.manager
```

- The command name can be entered in uppercase or lowercase, or a mixture of the two. These are all valid: `crtmqm`, `CRTMQM`, and `CRTmqm`.
- The flag can be entered as `-u`, `-U`, `/u`, or `/U`.
- The arguments `SYSTEM.DEAD.LETTER.QUEUE` and `jupiter.queue.manager` must be entered exactly as shown.

The following table lists the control commands, gives a brief description of each control command, and indicates whether the command function can be achieved through the user interface.

Notes:

1. When working from the user interface, you can get help for the command you are using by pressing the Help button.
2. When working from the command prompt, you can get help for the syntax of any of the commands by entering the command followed by a space and then a question mark. MQSeries responds by listing the syntax required for the selected command.

Table 3. MQSeries control commands

Command	Function	Description	User interface?
crtmqcvx	Data conversion	Creates a fragment of code that performs data conversion on data type structures.	No
crtmqm	Create queue manager	Creates a local queue manager and defines the default and system objects.	Yes
dltmqm	Delete queue manager	Deletes a specified queue manager.	Yes
dmpmqlog	Dump log	Dumps a formatted version of the MQSeries system log.	No
dspmqaut	Display authority	Displays the current authorizations to a specified object.	No
dspmqcsv	Display command server	Displays the status of the command server for the specified queue manager.	Yes
dspmqfls	Display MQSeries files	Displays the real file system name for all MQSeries objects that match a specified criterion.	No
dspmqtrn	Display MQSeries transactions	Displays details of in-doubt transactions.	No

Table 3. MQSeries control commands (continued)

Command	Function	Description	User interfac
endmqcsv	End command server	Stops the command server on the specified queue manager.	Yes
endmqm	End queue manager	Stops a specified local queue manager.	Yes
endmqtrc	End MQSeries trace	Ends tracing for the specified entity or all entities.	Yes
rcdmqimg	Record media image	Writes an image of an MQSeries object, or group of objects, to the log for use in media recovery.	No
rcrmqobj	Recreate object	Recreates an object, or group of objects, from their images contained in the log.	No
rsvmqtrn	Resolve MQSeries transactions	Commits or backs out internally or externally coordinated in-doubt transactions.	No
runmqchi	Run channel initiator	Runs a channel initiator process.	Yes
runmqchl	Run channel	Runs either a Sender (SDR) or a Requester (RQSTR) channel.	Yes
runmqdlq	Run dead-letter queue handler	Starts the dead-letter queue (DLQ) handler, a utility that you can run to monitor and handle messages on a dead-letter queue.	No
runmqlsr	Run listener	Runs a listener process.	Yes
runmqsc	Run MQSeries commands	Issues MQSC commands to a queue manager.	Yes
runmqtrm	Start trigger monitor	Invokes a trigger monitor.	No
setmqaut	Set/reset authority	Changes the authorizations to an object or to a class of objects.	No
strmqcsv	Start command server	Starts the command server for the specified queue manager.	Yes
strmqm	Start queue manager	Starts a local queue manager.	Yes
strmqtrc	Start MQSeries trace	Enables tracing.	Yes

For more information about the syntax and purpose of control commands, see the *MQSeries System Administration* manual.

MQSeries (MQSC) commands

You use the MQSeries (MQSC) commands to manage queue manager objects, including the queue manager itself, channels, queues, and process definitions. For example, there are commands to define, alter, display, and delete a specified queue. The MQSC commands and their functions are shown in Table 4.

When you have finished using the MQSC commands, type **END** and press Enter to return to the Windows NT command prompt.

Table 4. MQSC commands

Command	Function
ALTER CHANNEL	Alter the attributes of a channel
ALTER NAMELIST	Alter a list of names (usually a list of cluster or queue names)
ALTER PROCESS	Alter the attributes of an existing MQSeries process definition
ALTER QALIAS	Alter the attributes of an alias queue
ALTER QLOCAL	Alter the attributes of a local queue
ALTER QMGR	Alter the queue manager attributes for the local queue manager
ALTER QMODEL	Alter the attributes of a model queue
ALTER QREMOTE	Alter the attributes of a local definition of a remote queue, a queue-manager alias, or a reply-to queue alias
CLEAR QLOCAL	Clear the messages from the local queue
DEFINE CHANNEL	Define a new channel and set its attributes
DEFINE NAMELIST	Define a list of names (usually cluster names or queue names)
DEFINE PROCESS	Define a new MQSeries process definition, and set its attributes
DEFINE QALIAS	Define a new alias queue and set its attributes
DEFINE QLOCAL	Define a new local queue and set its attributes
DEFINE QMODEL	Define a new model queue and set its attributes
DEFINE QREMOTE	Define a new local definition of a remote queue, a queue-manager alias, or a reply-to queue alias, and to set its attributes
DELETE CHANNEL	Delete a channel definition
DELETE NAMELIST	Delete a namelist definition
DELETE PROCESS	Delete a process definition

Table 4. MQSC commands (continued)

Command	Function
DELETE QALIAS	Delete an alias queue definition
DELETE QLOCAL	Delete a local queue definition. You can specify that the queue must not be deleted if it contains any messages, or that it can be deleted even if it does contain messages
DELETE QMODEL	Delete a model queue definition
DELETE QREMOTE	Delete a local definition of a remote queue. It does not affect the definition of that queue on the remote system.
DISPLAY CHANNEL	Display a channel definition
DISPLAY CHSTATUS	Display the status of one or more channels
DISPLAY CLUSQMGR	Display cluster information about queue managers in a cluster
DISPLAY NAMELIST	Display the names in a namelist
DISPLAY PROCESS	Display the attributes of one or more MQSeries processes
DISPLAY QALIAS	Display the attributes of one or more queues
DISPLAY QCLUSTER	Display the attributes of one or more queues
DISPLAY QLOCAL	Display the attributes of one or more queues
DISPLAY QMGR	Display the queue manager attributes for this queue manager
DISPLAY QMODEL	Display the attributes of one or more queues
DISPLAY QREMOTE	Display the attributes of one or more queues
DISPLAY QUEUE	Display the attributes of one or more queues of any type
PING CHANNEL	Test a channel by sending data as a special message to the remote queue manager, and checking that the data is returned. The data is generated by the local queue manager
PING QMGR	Test whether the queue manager is responsive to commands.
REFRESH CLUSTER	Discard all local held cluster information (including any autodefined channels that are in doubt) and force it to be rebuilt. This allow the cluster to be “cold-started”.
REFRESH SECURITY	Perform a security refresh.
RESET CHANNEL	Reset the message sequence number for an MQSeries channel, with, optionally, a specified sequence to be used the next time the channel is started
RESET CLUSTER	Perform special operations on clusters
RESOLVE CHANNEL	Request a channel to commit or back out in-doubt messages

MQSC commands

Table 4. MQSC commands (continued)

Command	Function
RESUME QMGR	Inform other queue managers in a cluster that the local queue manager is available again for processing, and can be sent messages. It reverses the action of the SUSPEND QMGR command.
START CHANNEL	Start a channel
START CHINIT	Start a channel initiator
START LISTENER	Start a channel listener
STOP CHANNEL	Stop a channel
SUSPEND QMGR	Inform other queue managers in a cluster that the local queue manager is not available for processing, and cannot be sent messages. Its action can be reversed by the RESUME QMGR command.

For detailed information about each MQSC command, see the *MQSeries MQSC Command Reference* manual.

Running MQSC commands

You run MQSC commands by invoking the control command **runmqsc**. You can run MQSC commands:

- Interactively by typing them at a command prompt
- As a sequence of commands from a text file (a script)

For more information about using MQSC commands, see the *MQSeries System Administration* manual.

PCF commands

MQSeries programmable command format (PCF) commands allow administration tasks to be programmed into an administration program. In this way you can create queues and process definitions, and change queue managers, from a program. PCF commands cover the same range of functions that are provided by the MQSC facility. You can therefore write a program to issue PCF commands to any queue manager in the network from a single node. In this way, you can both centralize and automate administration tasks.

Unlike MQSC commands, PCF commands and their replies are not in a text format that you can read. For a complete description of the PCF data structures and how to implement them, see the *MQSeries Programmable System Management* manual.

The MQSeries administration interface (MQAI) provides a more convenient programmable way of using the function provided by PCF commands. For further information, see the *MQSeries Administration Interface Programming Guide and Reference* manual.

Chapter 15. Using the MQSeries Internet Gateway

This chapter introduces the MQSeries Internet Gateway. It also explains how to get more information about using the MQSeries Internet Gateway.

The MQSeries Internet Gateway is one of the installable components on the MQSeries Server CD-ROM, and is also available from the MQSeries Web site.

The following Gateways are available:
- MQSeries Internet Gateway for AIX®
- MQSeries Internet Gateway for HP-UX
- MQSeries Internet Gateway for Linux
- MQSeries Internet Gateway for OS/2®
- MQSeries Internet Gateway for OS/390® OpenEdition®
- MQSeries Internet Gateway for Sun Solaris
- MQSeries Internet Gateway for Windows NT

Overview of MQSeries Internet Gateway

MQSeries Internet Gateway provides a bridge between the synchronous World Wide Web and asynchronous MQSeries applications. With the MQSeries Internet Gateway, Web server software and MQSeries together provide an Internet-connected Web browser with access to MQSeries applications. This means that enterprises can take advantage of the low-cost access to global markets provided by the Internet, while benefitting from the robust infrastructure and assured message delivery of MQSeries.

User interaction with the MQSeries Internet Gateway is through HTML fill-out form POST requests; MQSeries applications respond by returning HTML pages to the MQSeries Internet Gateway, via an MQSeries queue.

The MQSeries Internet Gateway supports the following Web server interfaces:
- Common Gateway Interface (CGI)
- Internet Connection Application Programming Interface (ICAPI)
- Internet Services Application Programming Interface (ISAPI)
- Netscape Connection Application Programming Interface (NSAPI)

Note that:
- HP-UX does not support NSAPI.
- Sun Solaris does not support ISAPI.
- Linux supports CGI only.

MQSeries Internet Gateway documentation

The MQSeries product family Web site is at:

```
http://www.ibm.com/software/mqseries/
```

The following documentation is accessible from this Web site:

- *Getting Started with MQSeries Internet Gateway*. This is the starting point for the download and installation of MQSeries Internet Gateway.
- *MQSeries Internet Gateway User's Guide*. This is the main documentation for users of the MQSeries Internet Gateway.

Chapter 16. Obtaining additional information

This chapter describes the documentation for MQSeries for Windows NT and Windows 2000. It starts with a list of the publications, and then discusses:

MQSeries for Windows NT and Windows 2000 is described in the following books:

Table 5. MQSeries for Windows NT and Windows 2000 books

Order Number	Title
	Windows NT and Windows 2000 Specific Books
GC34-5389	*MQSeries for Windows NT and Windows 2000 Quick Beginnings*
SC34-5404	*MQSeries LotusScript Extension*
SC34-5387	*MQSeries for Windows NT Using the Component Object Model Interface*
	MQSeries Family Books
GC34-5761	*MQSeries V5.2 Release Guide*
SC33-1872	*MQSeries Intercommunication*
SC34-5349	*MQSeries Queue Manager Clusters*
GC33-1632	*MQSeries Clients*
SC33-1873	*MQSeries System Administration*
SC33-1369	*MQSeries MQSC Command Reference*
SC33-1482	*MQSeries Programmable System Management*
SC34-5390	*MQSeries Administration Interface Programming Guide and Reference*
GC33-1876	*MQSeries Messages*
SC33-0807	*MQSeries Application Programming Guide*
SC33-1673	*MQSeries Application Programming Reference*
SX33-6095	*MQSeries Programming Interfaces Reference Summary*
SC33-1877	*MQSeries Using C++*

Hardcopy books

The book that you are reading now is *MQSeries for Windows NT and Windows 2000 V5.2 Quick Beginnings*. This book and the *MQSeries V5.2 Release Guide* are the only books that are supplied in hardcopy with the product. However, all books listed in Table 5 on page 95 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 “PDF” on page 97.

Online information

This section describes:

- “Publications supplied with the product”
- “HTML and PDF books on the World Wide Web” on page 98
- “BookManager CD-ROMs” on page 98
- “Online help” on page 99

Publications supplied with the product

The MQSeries online documentation is delivered in compiled HTML (.CHM) and PDF formats on the CD-ROM.

HTML

You can view the compiled HTML version of the books from the Information Center; look for “IBM MQSeries Manuals” under the “Reference” section in the Table of Contents. Compiled HTML files are also on the MQSeries CD-ROM in the \Docs\htmlhelp\ directory. Double click on a .CHM file to view it.

You can install the MQSeries manuals in your national language. Not all of the manuals are translated into every language so you should also install the English versions to get a full set.

PDF

A PDF (Portable Document Format), corresponding to each hardcopy book, is available on the CD-ROM. You can read PDFs using Adobe Acrobat Reader. Also, you can download them to your own file system, or you can print them on a PostScript printer. If you have a Web browser, you can access the PDFs on the product CD-ROM by pointing your browser to `\Docs\acrobat`.

The PDFs 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

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:

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

Within these directories, you can find the complete set of PDFs that are available. Table 6 shows the file names used for the PDF files.

Table 6. MQSeries publications – file names

Book	File Name
MQSeries for Windows NT and Windows 2000 Quick Beginnings	AMQTAC01
MQSeries LotusScript Extension	AMQZAU00
MQSeries LotusScript Extension	AMQTAN00
MQSeries V5.2 Release Guide	AMQZAY00
MQSeries Intercommunication	CSQZAE04

Table 6. MQSeries publications – file names (continued)

Book	File Name
MQSeries Queue Manager Clusters	CSQZAH02
MQSeries Clients	CSQZAF04
MQSeries System Administration	AMQZAG01
MQSeries MQSC Command Reference	CSQZAJ04
MQSeries Programmable System Management	CSQZAI03
MQSeries Administration Interface Programming Guide and Reference	CSQZAT01
MQSeries Messages	AMQZAO01
MQSeries Application Programming Guide	CSQZAL04
MQSeries Application Programming Reference	CSQZAK04
MQSeries Programming Interfaces Reference Summary	CSQZAM04
MQSeries Using C++	AMQZAN03

HTML and PDF books on the World Wide Web

The MQSeries 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 MQSeries product family Web site is at:

```
http://www.ibm.com/software/mqseries/
```

By following links from this Web site you can:

- Obtain latest information about the MQSeries product family.
- Access the MQSeries books in HTML and PDF formats.
- Download MQSeries SupportPacs.

You can access the Web versions of the books directly from the MQSeries Information Center (see the “Reference” section).

BookManager CD-ROMs

The MQSeries library is supplied in IBM BookManager® format on a variety of online library collection kits, including the *Transaction Processing and Data* collection kit, SK2T-0730. You can view the softcopy books in IBM BookManager format using the following IBM licensed programs:

BookManager READ/2
BookManager READ/6000
BookManager READ/DOS

BookManager READ/MVS
BookManager READ/VM
BookManager READ for Windows

Online help

To view the online help for a specific topic when using the MQSeries user interface, press F1, click on the Help button, or use the Help menu.

Alternatively, you can click on the Start menu and select the MQSeries Information Center.

Part 4. Appendixes

Appendix A. MQSeries sample programs

Code samples for the following interfaces are provided with MQSeries:

- C
- C++
- VisualBasic
- COBOL
- Internet gateway
- MQSC
- DCE
- Activex

They are stored in and below the TOOLS directory, which, if you installed MQSeries into the default directories, is in `C:\Program Files\MQSeries` You can use these samples directly or modify them for experimental purposes.

For more information on the samples provided with MQSeries, see the *MQSeries Application Programming Guide.*

MQSC command file samples

Table 7 lists the MQSC command file samples. These are simply ASCII text files containing MQSC commands. You can invoke the **runmqsc** command against each file in turn to create the objects specified in the file.

By default, these files are located in directory **`C:\Program Files\MQSeries`**`\TOOLS\MQSC\SAMPLES`.

Table 7. MQSC command files

File name	Purpose
amqscic.tst	Defines objects for use in the sample CICS programs
amqscos0.tst	Defines a set of MQI objects for use with the sample programs

C program samples

Table 8 lists the sample MQSeries C source files. By default, the samples are in the directory **C:\Program Files\MQSeries**\tools\c\samples.

Samples are also provided for XATM (see Table 13 on page 107), and the Dead Letter Handler (see Table 9 on page 105).

To find out more about what the programs do and how to use them, see the *MQSeries Application Programming Guide*.

Table 8. Sample programs - C source files

File name	Purpose
amqsaicq.c	Creates a local queue using the MQSeries Administration Interface (MQAI)
amqsaiem.c	Demonstrates a basic event monitor using the MQAI
amqsailq.c	Inquires the current depth of the local queues using the MQAI
amqsbcg0.c	Reads and then outputs both the message descriptor and message context fields of all the messages on a specified queue.
amqsblst.c	Writes messages to and reads messages from a given queue and queue manager
amqsdsc0.c	Demonstrates the DCE GSS channel exit routines
amqsecha.c	Echoes a message from a message queue to the reply-to queue. Can be run as a triggered application program.
amqsgbr0.c	Writes messages from a queue to stdout, leaving the messages on the queue. Uses MQGET with the browse option.
amqsget0.c	Removes the messages from the named queue (using MQGET) and writes them to stdout.
amqsgrma.c	Get reference message sample. Gets messages from the queue specified in the input trigger message and check the existence of the file
amqsinqa.c	Inquires about some of the attributes of a queue using the MQINQ call
amqsprma.c	Put reference message sample. Creates a reference message, referring to a file, and puts it to a queue
amqsptl0.c	Puts messages to a list of message queues, distribution lists (using MQPUT and lists)
amqsput0.c	Copies stdin to a message and then puts this message on a specified queue.
amqsreq0.c	Puts request messages on a specified queue and then displays the reply messages.
amqsseta.c	Inhibits puts on a named queue and responds with a statement of the result. Runs as a triggered application.

Table 8. Sample programs - C source files (continued)

File name	Purpose
amqstrg0.c	A trigger monitor that reads a named initiation queue and then starts the program associated with each trigger message. Provides a subset of the full triggering function of the supplied RUNMQTRM command.
amqsvfc0.c	A sample C skeleton of a Data Conversion exit routine.
amqsxrma.c	Exit reference message sample. Channel message exit program that processes reference messages.
Note: You can create the objects required by these samples using the MQSC command file AMQSCOS0.TST.	

Dead letter handler sample programs

By default, the samples for the dead letter handler are in the directory **C:\Program Files\MQSeries**\tools\c\samples\dlq.

Table 9. Sample programs - Dead Letter Handler

File name	Purpose
amqodqka.c	Main source for the MQSeries dead letter handler
amqodqla.c	Functions for the MQSeries dead letter handler
amqodqma.c	RUNMQDLQ Parser: YACC source in amqodqma.y
amqodqna.c	Functions for the MQSeries dead letter handler
amqodqoa.c	Functions for the MQSeries dead letter handler
amqodqpa.c	Lexical scanner
amqodqua.c	Functions for the MQSeries dead letter handler

COBOL program samples

Table 10 lists the sample COBOL source files. By default, the COBOL samples are in directory **C:\Program Files\MQSeries**\tools\cobol\samples. To find out more about what the programs do and how to use them, see the *MQSeries Application Programming Guide*.

Table 10. Sample programs - COBOL source files

File name	Purpose
amq0gbr0.cbl	Writes messages from a queue to stdout, leaving the messages on the queue. Uses MQGET with the browse option.
amq0get0.cbl	Removes the messages from the named queue (using MQGET) and writes them to stdout.
amq0ptl0.cbl	Puts messages to a list of message queues, distribution lists (using MQPUT and lists)

COBOL samples

Table 10. Sample programs - COBOL source files (continued)

File name	Purpose
amq0put0.cbl	Copies `stdin` to a message and then puts this message on a specified queue.
amq0req0.cbl	Puts request messages on a specified queue and then displays the reply messages.
amqiech2.cbl	IBM COBOL program which echoes messages to reply to queue
amqiinq2.cbl	IBM COBOL program which inquires about some of the attributes of a queue using the MQINQ call
amqiset2.cbl	IBM COBOL program using MQSET
amqmech2.cbl	Merant COBOL program which echoes messages to reply to queue
amqminq2.cbl	Merant COBOL program using MQINQ
amqmset2.cbl	Merant COBOL program using MQSET
Note: You can create the objects required by these samples using the MQSC command file AMQSCOS0.TST.	

Supporting CICS and Encina for transaction processing

The samples include a CICS transaction and some associated headers and initialization programs. By default, the samples are in the directory **`C:\Program Files\MQSeries`**`\tools\c\samples`.

Table 11. Samples for transaction processing with CICS and Encina

File name	Purpose
amqscic0.ccs	Sample CICS transaction program. Uses queues that can be created using the MQSC script 'amqscic0.tst'.
amqscih0.h	Header file for CICS transaction sample amqscic0
amqsxae0.c	Encina transaction
amqzscgn.c	GLUE program for the CICS Task termination user exit
amqzscin.c	XA switch program for CICS XA Initialization
amqzsc5n.c	NT exit program for CICS User Exit 15/16 - Task Attach/Detach
amqzsc7n.c	NT exit program for CICS User Exit 17 - Task Abend
Note: You can create objects to support transaction processing using the MQSC command file AMQSCIC0.TST.	

Supporting Tuxedo for transaction processing

The samples include client transactions and some associated definitions and configuration files. By default, the samples are in the directory **C:\Program Files\MQSeries**\tools\c\samples.

Table 12. Samples for transaction processing with Tuxedo

File name	Purpose
amqstxsx.c	Sample server
amqstxgx.c	Sample GET client application
amqstxpx.c	Sample PUT client application
AMQSTXVX.FLDS	Field definition
UBBSTXCX.CFG	Configuration file

Supporting databases

By default, the database samples are located in the directories **C:\Program Files\MQSeries**\tools\cobol\samples\xatm and **C:\Program Files\MQSeries**\tools\c\samples\xatm.

Table 13. Sample programs - databases

C	COBOL	Purpose
amqsxag0.c	amq0xag0.cbl	Coordinates XA-compliant database managers
amqsxab0.sqc	amq0xab0.sqb	Functions to access MQBankTB table in MQBankDB database
amqsxaf0.sqc	amq0xaf0.sqc	Functions to access MQFeeTB table in MQFeeDB database
amqsxas0.sqc	amq0xas0.sqb	SQC and SQB programs for MQ coordinating XA-compliant database managers
db2swit.c	–	MQSeries XA switch program for DB2®
oraswit.c	–	MQSeries XA switch program for Oracle
sybswit.c	–	MQSeries XA switch program for Sybase

Miscellaneous tools

These tool files are provided to support the formatter and code conversion.

Table 14. Miscellaneous files

File name	Location	Purpose
AMQTRC.FMT	**C:\Program Files\MQSeries**\LIB	Defines MQSeries trace formats.
CCSID.TBL	MQM\CONV\TABLE	Edit this file to add any newly supported CSSID values to your MQSeries system. For more information about CCSID, see the CDRA documentation.

C++ program samples

Table 15 lists the sample C++ source files. By default, the C++ samples are in directory **C:\Program Files\MQSeries**\tools\cplus\samples. To find out more about what the programs do and how to use them, see the *MQSeries Application Programming Guide*.

Table 15. Sample programs - C++ source files

File name	Purpose
imqdput.cpp	Puts messages to a distribution list containing two queues
imqsget.cpp	Gets messages from a named queue
imqsput.cpp	Puts messages to a names queue
imqwrld.cpp	Puts and gets a message to and from a queue

VisualBasic program samples

Table 16 lists the sample VisualBasic source files. By default, the VisualBasic samples are in directories **C:\Program Files\MQSeries**\tools\vb\sampvb5 and **C:\Program Files\MQSeries**\tools\vb\sampvb6. To find out more about what the programs do and how to use them, see the *MQSeries Application Programming Guide*.

Table 16. Sample programs - VisualBasic source files

File name	Purpose
amqsaicq.vbp	Creates queues
amqsaiem.vbp	Basic event monitor
amqsailq.vbp	Displays queue information for a given queue manager

Table 16. Sample programs - VisualBasic source files (continued)

File name	Purpose
amqsbcgb.vbp	Reads and outputs both the message descriptor fields and the message content of all the messages on a queue
amqsgetb.vbp	Gets messages from a queue
amqsputb.vbp	Puts messages onto a queue
mqtrivc.vbp	Connects to a queue manager, puts and gets messages. Client version.
mqtrivs.vbp	Connects to a queue manager, puts and gets messages. Server version.
pcfsamp.vbp	Channel administrator. Connects to a remote queue manager, starts/stops channels, tests PCF commands.
strings.vbp	Allows putting and getting of data structures onto queues.

Internet Gateway program samples

Table 17 lists the sample Internet gateway source files. By default, the Internet gateway samples are in directory **C:\Program Files\MQSeries**\tools\dmqgate\samples. To find out more about what the programs do and how to use them, see the *MQSeries Application Programming Guide*.

Table 17. Sample programs - Internet gateway source files

File name	Purpose
dmqsamp1.c	Sample C program. Reads data from a request queue and writes a response to a reply queue using MQSeries API calls. It is a web-aware application. Data read from the input queue is decoded from a web interface (URL) format. Data written to the reply queue is written in an HTML format so that it can be displayed by a web browser.
dmqsamp2.c	Reads data from a request queue and writes a response to a reply queue using MQSeries API calls. It differs from DMQSAMP1 in that it uses the MQSeries Internet Gateway's ability to form a SESSION between the browser and a back-end application (in this case, DMQSAMP2).

DCE program samples

Table 18 on page 110 lists the sample DCE source files. By default, the DCE samples are in directory **C:\Program Files\MQSeries**\tools\dce\samples. To find out more about what the programs do and how to use them, see the *MQSeries Application Programming Guide*.

DCE samples

Table 18. Sample programs - DCE source files

File name	Purpose
dcesetkt.bat	Batch file, sets up the DCE keytable file so that the MQSeries DCE naming service can run (Windows NT and Windows 95 platforms)
dcesetsv.bat	Batch file, sets up the DCE security and directory servers so that the MQSeries DCE naming service can run (Windows NT and Windows 95 platforms).

ActiveX program samples

Table 19 lists the sample Activex source files. By default, the ActiveX samples are in directories:

- **C:\Program Files\MQSeries**\tools\mqax\samples\excel.
- **C:\Program Files\MQSeries**\tools\mqax\samples\HTML.
- **C:\Program Files\MQSeries**\tools\mqax\samples\VB.

To find out more about what the programs do and how to use them, see the *MQSeries Application Programming Guide*.

Table 19. Sample programs - ActiveX source files

File name	Purpose
Excel	
mqax.xls	This Excel example macro code illustrates using the various MQAX objects to send a request to the bank server and process the reply.
mqaxtriv.xls	This is a very simple example of the use of MQAX from an Excel spreadsheet. This file shows a trivial MQ example, putting a message and getting it back.
HTML	
mqaxtriv.htm	A trivial MQ example, putting a message and getting it back.
VisualBasic	
mqaxbsrv.vbp	MQAX VB application. Simple bank server.
mqaxclss.vbp	MQAX VB application. Displays a form on which you can query attributes of an MQSeries queue manager or queue.
mqaxdlst.vbp	MQAX VB application. Distribution lists sample.
mqaxtriv.vbp	MQAX VB application. Trivial tester, using PUT and GET.

Appendix B. Using response files for installing and removing MQSeries

This chapter provides information on the required formats for installation and uninstallation response files.

Installation response file format

The installation response file has a standard Windows .ini file format. All text is in English. Lines beginning with a semicolon (;) are comments. Case is ignored. A sample file is listed below. It consists of stanzas, with titles in square brackets, and, within each stanza, parameters in keyword=value format. Five stanzas are required; they are called:

- [InstallShield Silent]
- [DlgOrder]
- [Application]
- [MQSeries-0]
- [Queue Managers-0]

Of these five, the [MQSeries-0] and [Queue Managers-0] stanzas specify the installation parameters and are described below. Leave the other three stanzas unchanged.

The MQSeries-0 stanza

The Default parameter is required and can be used to specify a default installation.

Its value can be NO, YES, CURRENT, or DEFAULT.

Default=NO

This value means that the installation is specified by the other keywords. You must code this value if you do not require one of the other values (YES, CURRENT, or DEFAULT).

Default=YES

The installation updates the target machine to the latest MQSeries level. No new components are installed and no changes to the existing queue manager configuration are made. If there is no existing MQSeries installation on the target machine, Setup installs the Server, the Development Toolkit, and the MQSeries manuals in compiled HTML format components. Any other components that are present are reinstalled. With the exception of LockedFiles, other keywords in this stanza and in the [Queue Managers-0] stanza are ignored (and can be omitted).

If there is an existing MQSeries installation on the machine, Setup installs files in the existing folders; otherwise, it installs into the default folder for the machine. See the **PgmDir**, **DatDir**, **LogDir**, and **Folder** keywords.

Default=CURRENT
Interchangeable with Default=YES.

Default=DEFAULT
The installation updates the target machine to the latest MQSeries level. No changes to the existing queue manager configuration are made. Setup installs the Server, the Development Toolkit, and the MQSeries manuals in compiled HTML format components. Any other installed components, if any, are removed. The program files and the data files top-level folders, the log files folder and the program folder are forced to their default values for the machine. With the exception of LockedFiles, other keywords in this stanza and in the [Queue Managers-0] stanza are ignored (and can be omitted).

PgmDir=<folder>
Specifies the top-level folder for program files. The value <folder> must be a valid path on the target machine, or it can be DEFAULT. The value DEFAULT tells Setup to take the default value for the folder on the target machine, which is usually c:\program files\MQSeries, or, if there is an existing MQSeries installation on the machine, it tells Setup to continue to use the existing folder. For example:

```
PgmDir=c:\mqm pgm or PgmDir=default
```

DatDir=<folder>
Specifies the top-level folder for data files. The value <folder> must be a valid path on the target machine, or it can be DEFAULT. The value DEFAULT tells Setup to take the default value for the folder on the target machine, which is usually c:\Winnt\Profiles\All Users\Application Data\MQSeries, or, if there is an existing MQSeries installation on the machine, it tells Setup to continue to use the existing folder. For example:

```
DatDir=c:\mqm data or DatDir=default
```

LogDir=<folder>
Specifies the folder for log files. The value <folder> must be a valid path on the target machine, or it can be DEFAULT. The value DEFAULT tells Setup to take the default value for the folder on the target machine, which is usually c:\Winnt\Profiles\All Users\Application Data\MQSeries\log, or, if there is an existing MQSeries installation on the machine, it tells Setup to continue to use the existing folder. For example:

```
LogDir=c:\mqm log or LogDir=default
```

ProgramFolder=<programfolder>
Specifies the program folder name for MQSeries. The value <programfolder> must be a valid program folder name, or it can be DEFAULT. The value DEFAULT tells Setup to take the default value for the program folder on the target machine, which is IBM MQSeries, or, if there is a previous MQSeries configuration on the machine, it tells Setup to continue to use the existing program folder. For example:

```
ProgramFolder=My MQ Programs or ProgramFolder=default
```

Skip=<option>
Specifies whether to reinstall any MQSeries components that are already installed and are at the Version 5.2 level. The value of <option> can be YES or NO. For example:

```
Skip=YES
```

LockedFiles=<option>
Before transferring files to the target computer, Setup checks to see if any of the files it needs to replace are locked. This option specifies what Setup is to do if it finds any such files.

CONTINUE
Ignore the locked files and continue. After file transfer, Setup will restart the computer and replace the files during the restart.

CANCEL
Terminate the installation before transferring any files.

NORESTART
Ignore the locked files and continue. If any files are still found to be locked during file transfer, Setup will not restart the computer. In this case, you must restart the computer manually so that MQSeries will work correctly.

The rest of the keywords in this stanza specify component selections. There must be one keyword=value pair for each component; the keyword is the component name and the value must be one of:

INSTALL
Install or reinstall the component.

REMOVE
Remove the component, if it is already installed. This leaves unchanged any user data associated with the component. Only the Server and the Web Administration Server components have any data associated with them. The Server has the queue manager configuration and queue data; the Web Administration Server has user-generated scripts.

REMOVEDATA
Remove the component, if it is already installed, along with any of its associated user data.

For example:

- Server=INSTALL installs the Server component, or if it is already installed, reinstalls it.
- Development Toolkit=REMOVE does not install the Development Toolkit component, or if it is already installed, removes it.

The component names are listed in "Component names used in response files" on page 118.

There is a special keyword: Java. If you are installing MQSeries Version 5.2 on top of Version 5.1, and the Version 5.1 Server or Java Client components were installed, you must include the line `Java=REMOVE`. This acknowledges that the Version 5.1 Java files are to be deleted during Version 5.2 installation.

The Queue Managers-0 stanza

If the keyword Default in the [MQSeries-0] stanza has the value NO, keywords in this stanza are effective; otherwise, they are ignored.

The DoExistingQueueManagers keyword determines whether the <queuemanagername> keywords in this stanza are processed (described below). A value of NO means do not process the <queuemanagername> keywords (they can be omitted); a value of YES means process them. For example:

```
DoExistingQueueManagers=NO
```

Each remaining keyword in this stanza is the name of an existing queue manager on the target machine. The value tells Setup whether to allow the queue manager to be administered from a remote machine using the MQSeries Explorer. To be administered in this way, a queue manager must have a server connection channel defined, and must have a listener so that incoming requests cause the channel to be started. These keywords are not examined if DoExistingQueueManagers=NO is specified; they can be omitted in that case.

There must be one keyword for each queue manager on the target machine. The installation will fail if any particular queue manager keyword is missing. The format is:

```
<queuemanagername>=<value>
```

The value can be -1, 0, or a valid TCP/IP port number.

<queuemanagername>=-1
Means do nothing for this queue manager. No change is made.

<queuemanagername>=0
Means create the server connection channel so that the queue manager can be remotely administered.

<queuemanagername>=<portnumber>
Means create the server connection channel and create a channel listener using port number <portnumber>.

For example:

```
MyQueueManager=1414
```

creates, for MyQueueManager, a server connection channel and a TCP/IP listener for port 1414.

Here is an example of a complete installation response file:

```
[InstallShield Silent]
Version=v5.00.000
File=Response File
[Application]
Name=MQSeries
Version=CurrentVersion
Company=IBM
[DlgOrder]
Dlg0=MQSeries-0
Count=2
Dlg1=Queue Managers-0
[MQSeries-0]
DEFAULT=NO
PgmDir=e:\mqmpgm
DatDir=e:\mqmdat
LogDir=e:\mqmdat\log
ProgramFolder=IBM MQSeries
Skip=NO
LockedFiles=CONTINUE
Java=REMOVE
Server=INSTALL
Development Toolkit=INSTALL
Local Clients\Windows NT Client=REMOVE
Internet Gateway=REMOVE
Documentation\Internet Gateway=REMOVE
Documentation\Library HTML=INSTALL
Documentation\Library PDF=REMOVE
Web Administration Server=REMOVE
Additional Languages\en_us=REMOVE
Additional Languages\fr_fr=INSTALL
Additional Languages\de_de=REMOVE
Additional Languages\es_es=REMOVE
Additional Languages\it_it=REMOVE
Additional Languages\ja_jp=REMOVE
```

```
Additional Languages\ko_kr=REMOVE
Additional Languages\pt_br=REMOVE
Additional Languages\zh_tw=REMOVE
Additional Languages\zh_cn=REMOVE
[Queue Managers-0]
DoExistingQueueManagers=YES
QueueManager1=-1
QueueManager2=0
QueueManager3=1414
```

Uninstallation response file format

Like the installation response file, the uninstallation response file consists of stanzas and keyword=value pairs. Lines beginning with a semicolon (;) are comments. All text is in English. Case is ignored. An example of a complete response file listed below. Two stanzas are required, called: [MQSeries], and [Components].

The MQSeries stanza

This stanza has two required keywords: `MQSeries` and `LockedFiles`.

MQSeries=REMOVEDATA
This causes uninstallation to remove all of MQSeries on the target machine, including any queue manager definitions and queues. All queue manager data is lost. The [Components] stanza is not referenced and can be omitted.

MQSeries=REMOVE
This causes uninstallation to remove all of MQSeries on the target machine, except queue manager definitions and queues. Also left untouched are any Web Administration Server scripts. The [Components] stanza is not referenced and can be omitted.

MQSeries=REMOVECOMPONENTS
This causes uninstallation to remove specified components. The components are listed in the [Components] stanza, which must be present.

LockedFiles=<option>
Before deleting files from the target computer, uninstallation checks to see if any of the files are locked. This parameter specifies what uninstallation is to do if it finds any locked files.

CONTINUE
Ignore the locked files and continue. You must remove any locked files manually after uninstallation.

CANCEL
Terminate the uninstallation before deleting any files.

The Components stanza

There can be one keyword=value pair in this stanza for each MQSeries component. The keyword is the component name and the value must be: KEEP, SKIP, REMOVE, or REMOVEDATA. The component names are the same strings that are used in an installation response file (see "Component names used in response files" on page 118). Any installed components for which there is no keyword=value pair are not uninstalled. For example:

```
Server=REMOVEDATA or Server=KEEP
```

The values KEEP and SKIP mean do not remove this component.

The value REMOVE means remove this component, but keep any user data associated with the component. Only the Server and the Web Administration Server components have any data associated with them.

The value REMOVEDATA means remove the component, including any user data associated with the component. Only the Server and the Web Administration Server components have any data associated with them.

Here is an example of a complete uninstallation response file:

```
[MQSeries]
MQSeries=REMOVECOMPONENTS
LockedFiles=CONTINUE
[Components]
Server=REMOVE
Development Toolkit=KEEP
Local Clients\Windows NT Client=REMOVE
Internet Gateway=KEEP
Documentation\Library HTML=KEEP
Documentation\Library PDF=KEEP
Documentation\Internet Gateway=KEEP
Web Administration Server=REMOVEDATA
Additional Languages\en_us=KEEP
Additional Languages\fr_fr=KEEP
Additional Languages\de_de=KEEP
Additional Languages\es_es=KEEP
Additional Languages\it_it=KEEP
Additional Languages\ja_jp=KEEP
Additional Languages\ko_kr=KEEP
Additional Languages\pt_br=KEEP
Additional Languages\zh_tw=KEEP
Additional Languages\zh_cn=KEEP
```

Uninstallation response files

In this example:

- The Server component is removed, leaving the queue manager definitions and their data untouched.
- The Windows NT Client component is removed.
- The Web Administration Server component is removed, along with its associated data.

All other components are left unchanged.

Component names used in response files

The component names used in response files have fixed English values that are the same as the values stored in the registry after installation, under the following key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\IBM\MQSeries\CurrentVersion\Components
```

Their names are:

```
Server
Development Toolkit
Local Clients\Windows NT Client
Internet Gateway
Documentation\Library HTML
Documentation\Library PDF
Documentation\Internet Gateway
Web Administration Server
Additional Languages\en_us
Additional Languages\fr_fr
Additional Languages\de_de
Additional Languages\es_es
Additional Languages\it_it
Additional Languages\ja_jp
Additional Languages\ko_kr
Additional Languages\pt_br
Additional Languages\zh_tw
Additional Languages\zh_cn
```

Note: The names presented to the user during an attended installation are slightly different from these and are translated into the installation language.

Appendix C. 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.

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
IBM	IBMLink	MQSeries
OpenEdition	OS/2	OS/390
TXSeries	VisualAge	WebSphere

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.

Microsoft, Windows, Windows NT, and Visual Basic are trademarks or registered trademarks of Microsoft Corporation in the United States, or other countries, or both.

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

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

Index

S

T

U

V

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 44–1962–870229
  - From within the U.K., use 01962–870229
- 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.

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC34-5389-01