MQSeries® for AS/400®

Quick Beginnings

Version 5 Release 2

IBM

Second edition (December 2000)

This edition applies to MQSeries for AS/400, Version 5 Release 2, and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Tables

Welcome to MQSeries for AS/400

This book describes IBM® MQSeries for AS/400, Version 5.2. It explains how to plan for, install, and begin to use the product. For latest information about MQSeries for AS/400, V5.2, refer also to the readme file on the publications CD-ROM, and to the MQSeries family Web site at:

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

Road map

Use Table 1 to find the information you need to get started with *MQSeries for AS/400*.

Table 1. Getting started road map

If you want to...	Refer to...
Learn about system requirements for MQSeries for AS/400	"Chapter 1. Planning to install the MQSeries for AS/400 server" on page 3
Install MQSeries for AS/400	"Chapter 2. Installing and migrating to MQSeries for AS/400" on page 9
Apply maintenance to MQSeries for AS/400	"Chapter 3. Applying maintenance to MQSeries for AS/400" on page 25
Delete MQSeries for AS/400	"Chapter 4. Deleting MQSeries for AS/400, V5.2" on page 27
Read about MQSeries in general	"Chapter 5. About MQSeries" on page 31
Start using command sets	"Chapter 6. Using MQSeries for AS/400, V5.2" on page 39
View or print online documentation	"Chapter 7. Obtaining additional information" on page 53
Run sample programs	"Appendix A. Sample MQI programs" on page 61
Contact IBM	*Sending your comments to IBM*

Conventions

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

- **Boldface type** indicates the name of an item you need to select.
- *Italic type* indicates new terms, book titles, or variable information that you must replace with actual values.
- `Monospace type` indicates an example (such as a fictitious path or file name) or text that is displayed on the screen.

What's new in MQSeries for AS/400, Version 5 Release 2

MQSeries for AS/400, 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.
- 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.
- Access to MQSeries objects for users with *ALLOBJ authority has been improved.
- The listener is now nonthreaded by default. The threaded listener can be enabled via an entry in the configuration file qm.ini.

What's new

- The ENDMQM CL command has been enhanced to allow you to quiesce all queue managers.

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

Part 1. Installing MQSeries for AS/400

Chapter 1. Planning to install the MQSeries for AS/400 server

This chapter provides a summary of the requirements for running MQSeries for AS/400. It discusses the network protocols and compilers supported, the delivery media, and the various components of the product.

Hardware requirements

MQSeries for AS/400, V5.2 runs on any machine that is capable of running OS/400® V4R4 or OS/400 V4R5. The machine must have sufficient storage to meet the combined requirements of the programming prerequisites, MQSeries for AS/400, V5.2, the access methods, and the application programs, whether from IBM or other vendors.

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

Disk space required

For the MQSeries for AS/400, V5.2 base code and server you should allow approximately 70 MB of storage.

Software requirements

These are the minimum supported software levels. Later levels, if any, will be supported unless otherwise stated.
- OS/400 Version 4 Release 4
- OS/400 Version 4 Release 5

Connectivity

The network protocols supported by MQSeries for AS/400 are:
- TCP
- SNA LU 6.2

AS/400 install libraries and directories

MQSeries for AS/400 installs into the AS/400 library structure and also uses directories in the IFS root file system.

Libraries and directories

AS/400 library structure

Installation of MQSeries for AS/400 creates the following libraries:

QMQM
> The MQSeries base product library.

QMQMSAMP
> The MQSeries samples library (if you choose to install the samples).

QMQM29xx
> A library containing a message file that is specific to your national language. This message file is automatically copied into the appropriate QSYS29xx library when the product is installed.

The following language versions are available for MQSeries for AS/400, V5.2:

Table 2. National-language versions of MQSeries for AS/400, V5.2

Language ID	Language
2909	Belgian English
2966	Belgian French MNCS (Multi-National Character Set)
2981	Canadian French MNCS
2950	English Uppercase
2924	English Uppercase and Lowercase
2938	English Uppercase DBCS (Double-Byte Character Set)
2984	English Uppercase and Lowercase DBCS
2928	French
2940	French MNCS
2932	Italian
2942	Italian MNCS
2931	Spanish
2962	Japanese
2986	Korean
2989	Simplified Chinese

Notes:

1. To run the Japanese language version of MQSeries for AS/400 the CCSID of the job must be 939 (5035) rather than 930 (5026) because MQSeries uses lowercase English alphabets.

2. If you are installing MQSeries for AS/400 onto a machine on which the primary language is not one of those listed, you will be asked to load a different CD-ROM and you will not be able to proceed. Rather than trying to install the product in a language that is not supplied, specify LNG(*SAVVOL) on the RSTLICPGM command. Alternatively, install the product in one of the supplied languages and then copy the installed files to your own QSYS directory.

QMxxxx library

In addition, each time you create a queue manager MQSeries automatically creates an associated library. This library contains objects specific to the queue manager, including journals and associated receivers. The name of this library is derived from the name of the queue manager prefixed with the characters QM. For example, for a queue manager called TEST, the library would be called QMTEST.

You can use the WRKLIB command to list all the libraries that MQSeries for AS/400 has created. Against the queue manager libraries, you will see the text QMGR: *QMGRNAME*. The format of the command is:

```
WRKLIB LIB(QM*)
```

These queue manager-associated libraries are retained when the product is deleted.

IFS root file system

The integrated file system (IFS) is a part of OS/400 that supports stream input/output and storage management similar to personal computer and UNIX® operating systems, while providing an integrating structure over all information stored in the AS/400. When you install MQSeries for AS/400, V5.2 two directories are created in the IFS *root* file system. They are:

/QIBM/ProdData/mqm

Subdirectories below this contain all the product data - for example, C++ classes. Data in this directory is deleted and replaced each time the product is installed.

/QIBM/UserData/mqm

Subdirectories below this contain all user data relating to queue managers. Three subdirectories, /QIBM/UserData/mqm/errors, /QIBM/UserData/mqm/qmgrs, and /QIBM/UserData/mqm/trace, are also created when the product is installed.

When you install the product, an mqs.ini file is created in directory /QIBM/UserData/mqm/ (unless it is already there from a previous installation). When you create a queue manager a qm.ini file is created in the directory /QIBM/UserData/mqm/qmgrs/*QMGRNAME*/ (where *QMGRNAME* is the name of the queue manager.) The qm.ini file is the equivalent of the QMINI file in previous releases of MQSeries for AS/400.

Data in this directory is retained when the product is deleted.

The IFS file structure is shown in the *MQSeries for AS/400 System Administration* book.

Libraries and directories

User profiles

When you install MQSeries for AS/400, V5.2, two user profiles are created; QMQM and QMQMADM. These two objects are central to the correct running of MQSeries for AS/400. Under no circumstances should you alter or delete them. If you do, IBM cannot guarantee correct behavior of your product.

These profiles are retained when the product is deleted.

Platform consideration

Note that in MQSeries for AS/400, directory names that begin with a @ character on other MQSeries platforms are changed to begin with a &. For example @SYSTEM is changed to &SYSTEM.

Compilers supported for MQSeries for AS/400, V5.2 applications

MQSeries for AS/400, V5.2 supports the following compilers:
- IBM ILE C for AS/400 compiler (5768-CX2)
- IBM ILE C++ for AS/400 compiler (5799-GDW)
- IBM VisualAge® for C++ for AS/400 compiler (5769-CX4)
- IBM ILE COBOL for AS/400 compiler Version 3 (5769-CB1)
- IBM ILE RPG for AS/400 compiler (5769-RG1)
- AS/400 Developer Kit for Java (5769-JV1)

Delivery

MQSeries for AS/400, V5.2 is supplied on CD-ROM.

There are four CD-ROMs.
- One contains the MQSeries for AS/400, V5.2 base product and sample programs.
- Another contains the client products that you can use with MQSeries for AS/400.
- The third contains MQSeries for AS/400 publications.
- Support for Java on MQSeries is separately installable from the fourth CD-ROM included in this product package. Alternatively, you can download support for Java on MQSeries from the MQSeries Web site at:

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

    where the latest version of this support is always available.

For further information about Java support and how to install it, refer to the *MQSeries Using Java* book.

MQSeries for AS/400 components

The components are as follows:

Base   Support to enable you to create and support your own applications. This includes the run-time component that provides support for external applications. It also includes support for client connections from other MQSeries products.

Samples (Option 1)

Sample application programs. The source is supplied in the QMQMSAMP library and executables are supplied in the QMQM library. See "Appendix A. Sample MQI programs" on page 61.

MQSeries online documentation

HTML and PDF versions of the books are supplied on the publications CD-ROM. They are not installable. You can read them directly from the CD-ROM or you can copy them to your system. For example, you could copy them to your PC hard drive, or store them in the AS/400 IFS and then access them using a network drive.

Each book is supplied 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

Readme file

Latest information about the product, which became available after publication of this book. You can find the readme file on the publications CD-ROM and should review it before starting to install MQSeries for AS/400.

Creating the system-default objects

System-default objects are created automatically when you issue the CRTMQM command to create a queue manager.

You can refresh them using the STRMQM command. Refer to the online help for information about using this command.

Chapter 2. Installing and migrating to MQSeries for AS/400

This chapter describes installation of and migration to MQSeries for AS/400, V5.2. It covers the following topics:
- "Before installation"
- "Installation procedure" on page 11
- "Verifying the installation" on page 12
- "Quiescing MQSeries" on page 13
- "Migrating to MQSeries for AS/400, V5.2 from V4R2M1 or earlier releases" on page 14
- "Migrating to MQSeries for AS/400, V5.2 from V5.1" on page 20
- "Verifying the migration" on page 22

In addition to reading the information in this chapter, you should refer to the readme file on the publications CD-ROM, and also to the latest information available on the MQSeries Web site at:

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

Before installation

This section describes how to install MQSeries for AS/400, V5.2. If you are migrating to V5.2 from V5.1, refer instead to "Migrating to MQSeries for AS/400, V5.2 from V5.1" on page 20. If you are migrating from an earlier version of MQSeries for AS/400, refer to "Migrating to MQSeries for AS/400, V5.2 from V4R2M1 or earlier releases" on page 14.

In order to run MQSeries for AS/400, V5.2, you must have the OS/400 V4R4 or V4R5 operating system installed on your machine . If you have OS/400 V4R2 or V4R3 operating system installed, you can use only MQSeries for AS/400 V4R2M1.

Setting system values

Before installing MQSeries for AS/400, you should use the DSPSYSVAL command to check that the following system values are set to the requirements of your enterprise:
- QCCSID
- QUTCOFFSET
- QSYSLIBL
- QALWOBJRST

You can change these values, if necessary, using the CHGSYSVAL command.

Before installation

QCCSID

Every message has coded-character set identifier (CCSID) in its header. The CCSID tag identifies the code page and character set of the source. For CCSIDs that are supported on the AS/400 see the *AS/400 National Language Support* book.

A queue manager obtains its CCSID from the job that created it. If the job CCSID is not a valid value in the range 1–65534, the queue manager uses the default CCSID value instead. You can change the CCSID used by the MQSeries queue manager by using the CL command CHGMQM.

Note: The CCSID must be either Single-byte character set (SBCS), or Mixed, that is SBCS and DBCS. It must not be DBCS only.

QUTCOFFSET

You should check that the coordinated universal time offset (QUTCOFFSET) system value has been set, to indicate the relationship between the system time and Greenwich Mean Time (GMT). You do this by working with the CHGSYSVAL command.

If QUTCOFFSET is not set, it takes the default value of zero. MQSeries for AS/400 then assumes that the local system time is universal time coordinated (UTC) - that is, GMT - and time stamps the MQSeries for AS/400 messages accordingly.

QSYSLIBL

Ensure that QSYS2 is included in the list of libraries that make up the system part of the library list.

MQSeries for AS/400 uses programs in this library for data conversion and SNA LU 6.2 communication.

QALWOBJRST

Ensure that the QALWOBJRST system value is set to *ALL or *ALWPGMADP before you install MQSeries for AS/400. If it is set to *NONE, the installation will fail.

After installation, reset QALWOBJRST to its original value to maintain system security.

Installation procedure

1. To install the MQSeries for AS/400, V5.2 base product, issue the command:

   ```
   RSTLICPGM LICPGM(5733A38) DEV(install device) OPTION(*BASE)
   ```

   where:
   - 5733A38 is the product identifier for MQSeries for AS/400, V5.2, and
   - `install device` is the device from which the product is to be loaded, typically a CD-ROM - for example, OPT01.

2. To install the sample applications, issue the command:

   ```
   RSTLICPGM LICPGM(5733A38) DEV(install device) OPTION(1)
   ```

   Note: You can install only one instance of MQSeries for AS/400 in each partition of your AS/400 machine.

If installation of MQSeries fails part way through, you should remove the partly-installed objects before attempting reinstallation. You can do this by deleting the QMQM library (and the QMQMSAMP library if necessary) and the IFS directory /QIBM/ProdData/mqm and its subdirectories. To delete the IFS directories, use the EDTF command, for example:

```
EDTF STMF('QIBM/UserData')
```

and select option 9 for the mqm directory.

Installation of translated versions

MQSeries for AS/400 is installed in the language that is the primary language on your system.

You can install additional versions of the product in any of the languages shown in Table 2 on page 4. To do so, issue the following command specifying the appropriate language ID:

```
RSTLICPGM LICPGM(5733A38) DEV(install device) RSTOBJ(*LNG) LNG(language ID)
```

This will install the commands, message file, and panel groups into the relevant QSYS library for the language. For example, library QSYS2928 is used for French.

Reinstallation

Reinstallation is discussed in "Reinstalling MQSeries for AS/400" on page 25.

Verifying the installation

To ensure that the product has loaded correctly, issue the Display Software Resources (DSPSFWRSC) command and check that the licensed program 5733A38 is listed. If you have installed the base and the optional samples, you should see:

```
Resource
 ID      Option  Feature  Description
 5733A38  *BASE   5050     MQSeries for AS/400
 5733A38  *BASE   2924     MQSeries for AS/400
 5733A38  1       5050     MQSeries for AS/400 - Samples
```

If you press F11 while viewing the Display Software Resources screen, you will see the library and version number of the products installed:

```
Resource                       Feature
 ID      Option  Feature  Type    Library   Release
 5733A38  *BASE   5050     *CODE   QMQM      V5R2M0
 5733A38  *BASE   2924     *LNG    QMQM      V5R2M0
 5733A38  1       5050     *CODE   QMQMSAMP  V5R2M0
```

If you have installed additional language versions, you will also see entries for these. For example, if you have installed the French version, for which the language ID is 2928, you will see:

```
Resource
 ID      Option  Feature  Description
 5733A38  *BASE   2928     MQSeries for AS/400
```

and when you press F11:

```
Resource                       Feature
 ID      Option  Feature  Type    Library   Release
 5733A38  *BASE   2928     *LNG    QSYS2928  V5R2M0
```

Post installation tasks

When you have correctly installed MQSeries for AS/400, V5.2 on your system:

1. Refer to the MQSeries family Web site at:

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

   for latest product information, and install and apply all PTFs that are recommended.

2. Start the MQSeries subsystem, by issuing the command:

   STRSBS SBSD(QMQM/QMQM)

3. Make your user profile a member of the group profile QMQMADM, as described in "Before you start" on page 39.

You are now ready to start using MQSeries for AS/400.

Quiescing MQSeries

The orderly shutdown of MQSeries for AS/400 is called *quiescing*. You may need to quiesce MQSeries for AS/400, for example, to:
- Take a backup of the system, or
- Update MQSeries for AS/400

Quiescing V4R2M1 or earlier releases of MQSeries for AS/400

To quiesce MQSeries for AS/400 V4R2M1 or an earlier release:

1. Sign on to a new interactive MQSeries for AS/400 session, ensuring that you are not accessing any MQSeries objects.
2. Ensure that you have:
   - *ALLOBJ authority, or object management authority for the QMQM and QMQMADM libraries. (The QMQMADM library exists only if you have installed the MQSeries for AS/400 Administration utility.)
   - *USE authority for the following program or programs:
     - QMQM/AMQIQES4

       or
     - QMQM/AMQIQEM4
     - QMQM/AMQSTOP4
     - QMQM/AMQSPECA
3. Warn all users that you are going to stop MQSeries for AS/400.
4. Warn users not to start Message Queue Interface applications while MQSeries for AS/400 is being quiesced. To ensure that they cannot start, you should revoke users' authorities to MQSeries for AS/400 by issuing the RVKMQMAUT command.
5. Exit the Administration utility, if it is running.
6. Quiesce the queue manager by running program AMQSTOP4:

   ```
   CALL QMQM/AMQSTOP4 PARM(QMGRNAME *CNTRLD 15)
   ```

   where *QMGRNAME* is the name of the queue manager and *15* is a time-out value in seconds.

   If this fails, you can force the queue manager to stop by issuing:

   ```
   CALL QMQM/AMQSTOP4 PARM(QMGRNAME *FORCE 15)
   ```

   If you do not have the AMQSTOP4 program on your system, use instead the AMQIQES4 program:

   ```
   CALL QMQM/AMQIQES4
   ```

Quiescing V5.1 or V5.2 of MQSeries for AS/400

To quiesce MQSeries for AS/400, V5.1 or V5.2:

1. Sign on to a new interactive MQSeries for AS/400 session, ensuring that you are not accessing any MQSeries objects.

Quiescing MQSeries

2. Ensure that you have:
   - *ALLOBJ authority, or object management authority for the QMQM library.
   - *USE authority for the following programs:
     - QMQM/AMQIQEM4
     - QMQM/AMQSTOP4
     - QMQM/AMQSPECA

   Also ensure that you have the necessary authority to run the ENDSBS command.
3. Warn all users that you are going to stop MQSeries for AS/400.
4. Warn users not to start Message Queue Interface applications while MQSeries for AS/400 is being quiesced. To ensure that they cannot start, you should revoke users' authorities to MQSeries for AS/400 by issuing the RVKMQMAUT command.
5. Quiesce all queue managers.

   On Version 5.2 do this using the ENDMQM command:

   ```
   ENDMQM MQMNAME(*ALL) OPTION(*CNTRLD) ENDCCTJOB(*YES) TIMEOUT(15)
   ```

   If this fails, you can force queue managers to stop by issuing:

   ```
   ENDMQM MQMNAME(*ALL) OPTION(*IMMED) ENDCCTJOB(*YES) TIMEOUT(15)
   ```

   On Version 5.1 quiesce queue managers by running program AMQSTOP4:

   ```
   CALL QMQM/AMQSTOP4 PARM(*ALL *CNTRLD 15)
   ```

   If this fails, you can force queue managers to stop by issuing:

   ```
   CALL QMQM/AMQSTOP4 PARM(*ALL *FORCE 15)
   ```
6. End the MQSeries subsystem, by entering:

   ```
   ENDSBS SBS(QMQM)
   ```

Migrating to MQSeries for AS/400, V5.2 from V4R2M1 or earlier releases

This section describes how to migrate to MQSeries for AS/400, V5.2 from MQSeries for AS/400 V3R7, MQSeries for AS/400 V4R2, or MQSeries for AS/400 V4R2M1. You must pay particular attention when migrating to MQSeries for AS/400, V5.2 from one of these releases. The product has been altered and enhanced significantly in Version 5.

This section outlines the steps involved in migrating to MQSeries for AS/400, V5.2. You may find additional, important information that became available after publication of this book if you refer to the readme file on the publications CD-ROM, or to the MQSeries family Web site at:

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

What are the differences in Version 5

MQSeries for AS/400, Version 5 provides significant new function over previous releases, including support for multiple queue managers, queue manager clusters, and substantially increased message and queue sizes. Enabling this new function has involved some changes to the internal architecture of MQSeries. For example this release makes substantial use of the Integrated File System whereas, previously, all MQSeries data was stored in OS/400 objects in libraries.

Another significant change with this release concerns security. To increase compatibility with the other V5 MQSeries platforms and also to provide authorization to individual MQI calls, the OS/400 object-based security model used in previous releases has been replaced by one using the MQSeries OAM (Object Authority Manager).

As a consequence of these, and other changes, migration to MQSeries for AS/400, V5.2 from V4R2M1 or an earlier release requires special consideration and careful planning. Areas to which you should pay close attention include:

- Existing RPG, COBOL, and C applications using the MQI (MQSeries Interface) should work successfully without the need for recompilation. There have been some changes in detail in the way some CL commands are implemented. CL programs using these commands might need to be edited and recompiled.

- If you intend to use threaded channel listeners, all user exits including channel and data conversion exits must be recompiled using ILE compilers, bound to the threaded MQSeries libraries, made thread-safe and enabled for teraspace memory. Details can be found in the *MQSeries Application Programming Guide* and the *MQSeries Intercommunication* book.

- MQSeries authorizations continue to be administered through the GRTMQMAUT (Grant MQM Authority) and RVKMQMAUT (Revoke MQM Authority) commands, but there are substantial changes:
  - Authorization lists and reference objects are no longer supported.
  - Authorizations that may be given to a user for MQSeries objects are now grouped into four categories: MQI, Context, Administration, and Generic.

There are additional security considerations to take into account. For details see the *MQSeries for AS/400 System Administration* book.

Overview of migration process from V4R2M1 or earlier

There are two possible migration scenarios:

- Either you are migrating from a previous version of MQSeries on the same machine, or
- You are transferring MQSeries to a new machine.

Migrating MQSeries

The process is the same except that:

- If you are moving to a new machine, there is one additional step, which is to transfer data from the old machine to the new.
- If you are upgrading the system on the same machine, you must save your MQSeries data because the migration process completely deletes your old version of the product.

To help you migrate, MQSeries for AS/400, V5.2 includes a utility program called MIGRATEMQM. This program migrates your MQSeries configuration and objects to MQSeries for AS/400, V5.2. It also creates a new queue manager with the same name as the original one. (This queue manager will act as the default queue manager on your new system unless you have already defined a default queue manager). The migration program reads information from the earlier release and creates a matching queue manager at the new level. It also creates all the MQSeries objects such as queues and channels. It copies messages held on the old queue manager's queues to the new queue manager. It does **not**, however, migrate your authority definitions.

In MQSeries for AS/400, V5.2, each queue manager has its own library containing that queue manager's journals and journal receivers. In V4R2M1 and earlier versions of MQSeries for AS/400, all the journals were in QUSRSYS and all the receivers were in QMQMDATA. The migration program obtains the configuration and message data from the journals and data library of the earlier release.

Before migration from V4R2M1 or earlier

Before migrating to MQSeries for AS/400, V5.2 from V4R2M1 or earlier , carry out the following procedure on the version of MQSeries for AS/400 that is currently installed:

1. Stop all applications that are using the existing version of MQSeries for AS/400.
2. End all MQSeries channels. To do this, use the WRKMQMCHL command and select option 15.
3. Exit the administration utility if it is running.
4. End the MQSeries command server. To do this, enter the command:

   `ENDMQMCSVR MQMNAME(`*QMGRNAME*`) OPTION(*IMMED)`

   where *QMGRNAME* is the name of the queue manager.
5. Remove any unwanted messages from your MQSeries queues. To do this, use the WRKMQMMSG command and select option 4.
6. Resolve any in-doubt messages that are held by sender or server channels. To do this, use the WRKMQMCHST command and select option 17.

7. Save details of your existing MQSeries authorities. To find these details, issue the DSPMQMOBJN command to list the queue manager and all the MQSeries queues and processes that you have. Then issue the DSPMQMAUT command to list the access that users have to each of these objects.

8. Quiesce the product. To do this, refer to "Quiescing V4R2M1 or earlier releases of MQSeries for AS/400" on page 13.

9. Save the latest media recovery checkpoint. To do this enter the following three commands:

   ```
   STRMQM MQMNAME(QMGRNAME)
   RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL)
   ENDMQM MQMNAME(QMGRNAME)
   ```

10. Quiesce the product again.

11. Copy your MQSeries CL and MQSC files to a suitable save library such as QGPL.

12. Save your MQSeries data:
    - Create a save file for the libraries QMQMDATA and QMQMPROC. To do this, issue the commands:
      ```
      CRTSAVF FILE(QGPL/QMQMDATA)
      CRTSAVF FILE(QGPL/QMQMPROC)
      ```
    - Save the libraries QMQMDATA and QMQMPROC into these save files. To do this, enter the commands:
      ```
      SAVLIB LIB(QMQMDATA) DEV(*SAVF) SAVF(QGPL/QMQMDATA)
      SAVLIB LIB(QMQMPROC) DEV(*SAVF) SAVF(QGPL/QMQMPROC)
      ```
    - Create a save file for your journals. To do this, issue the command:
      ```
      CRTSAVF FILE(QGPL/QMQMJRN)
      ```
    - Save the journals into the save file:
      ```
      SAVOBJ OBJ(AMQ*) LIB(QUSRSYS) DEV(*SAVF) OBJTYPE(*JRN) SAVF(QGPL/QMQMJRN)
      ```

13. If you are transferring MQSeries to a new machine, transfer the save files to the new machine.

14. **Do not start this step until you have saved all your MQSeries data. When you have completed this step your data will be recoverable only from your saved images**.

    Delete your old version of MQSeries for AS/400. To do this, use the GO LICPGM menu and select option 12 (Delete licensed program). Select all the installable options of MQSeries for AS/400 and delete them.

    Note that when deleting MQSeries for AS/400 V4R2M0 base option you may first have to issue the command:

    ```
    DLTJRNRCV JRNRCV(QMQMDATA/AMQ*)  DLTOPT(*IGNINQMSG)
    ```

Migrating MQSeries

Migration from V4R2M1 or earlier

To migrate to the new release of MQSeries for AS/400, follow this procedure. Note that you need *ALLOBJ authority to complete the procedure.

1.

   a. Install MQSeries for AS/400, V5.2. See "Installation procedure" on page 11 for details.

   b. Verify the installation. See "Verifying the installation" on page 12 for details.

   c. Perform the tasks described in "Post installation tasks" on page 12.

2. Restore the MQSeries objects for libraries QMQMDATA and QMQMPROC using the following commands:

```
RSTLIB SAVLIB(QMQMDATA) DEV(*SAVF) SAVF(QGPL/QMQMDATA)
RSTLIB SAVLIB(QMQMPROC) DEV(*SAVF) SAVF(QGPL/QMQMPROC)
```

3. Restore the MQSeries local and remote journals using the following command:

```
RSTOBJ OBJ(AMQ*) SAVLIB(QUSRSYS) DEV(*SAVF) OBJTYPE(*JRN) SAVF(QGPL/QMQMJRN)
```

4. Associate the journal receivers by:
   a. Issuing the command WRKJRN, specifying the journals QUSRSYS/AMQAJRN and QUSRSYS/AMQRJRN
   b. Using option 9 to associate the journals

5. Make sure your user profile is a member of the group profile QMQMADM.

6. Invoke the migration program MIGRATEMQM. To do this, issue the command:

```
CALL PGM(QMQM/MIGRATEMQM)
```

This command may be issued from the command prompt, or may be submitted as a batch job. Depending on the number of MQSeries objects and the number of messages on queues, the program may take some time to complete. The program creates a default queue manager that matches the one on your previous release.

When the program has completed, check the job log for errors. If any errors were encountered during channel-definition migration (for example, no channel-definition file was found) or during the migration of channel-synchronization information (for example, no synchronization file was found), you can rerun these steps individually. To rerun channel-definition migration, enter:

```
CALL PGM(QMQM/AMQRMCHA) PARM(QMGRNAME)
```

where *QMGRNAME* is the name of the queue manager.

To rerun the migration of channel-synchronization information, enter:

```
CALL PGM(QMQM/AMQRMSYA) PARM(QMGRNAME)
```

7. If you wish to set up your work management environment, job descriptions, and pools, see the *MQSeries for AS/400 System Administration* book for guidance. Otherwise, use the default setup.

8. If you intend to use threaded channel listeners, migrate and recompile your user exit programs. Recompile all OPM programs and programs that are not thread-safe, to make them ILE compliant and thread-safe. See the *MQSeries Application Programming Guide* and the *MQSeries Intercommunication* book for guidance on how to do this.

9. Reapply security to your MQSeries objects, using the command GRTMQMAUT. As mentioned earlier the authority model has changed in Version 5 of MQSeries for AS/400. Use the information in the following tables as guidance for mapping the old authority to the new.

   Note: Any PCF applications must be given explicit access to the queues. See the *MQSeries for AS/400 System Administration* book for further information about security.

Table 3. Authorities required to open an object

Opening MQSeries object with option MQOO_	MQSeries object authority required - earlier releases	MQSeries object authority required - V5.1 and V5.2
BROWSE	*READ	*BROWSE
INPUT_AS_Q_DEF	*READ and *DLT	*GET
INPUT_EXCLUSIVE	*READ and *DLT	*GET
INPUT_SHARED	*READ and *DLT	*GET
INQUIRE	*READ	*INQ
OUTPUT	*ADD	*PUT
SET	*UPD	*SET

Table 4. Authorities for Context and AlternateUserID

Opening MQSeries object with option MQOO_	MQSeries object authority required - earlier releases	MQSeries object authority required - V5.1 and V5.2
ALTERNATE_USER AUTHORITY	*MQMALTUSR (+ *DLT on ADM object)	*ALTUSR
PASS_ALL CONTEXT	*MQMPASSALL (+ *READ and *OBJOPR on ADM object)	*PASSALL
PASS_IDENTITY CONTEXT	*MQMPASSID (+ *READ on ADM object)	*PASSID

Migrating MQSeries

Table 4. Authorities for Context and AlternateUserID (continued)

Opening MQSeries object with option MQOO_	MQSeries object authority required - earlier releases	MQSeries object authority required - V5.1 and V5.2
SET_ALL CONTEXT	*MQMSETALL (+ *ADD and *UPD on ADM object)	*SETALL
SET_IDENTITY CONTEXT	*MQMSETID (+ *UPD on ADM object)	*SETID

Migrating to MQSeries for AS/400, V5.2 from V5.1

There are two major types of upgrade:

- The upgrade takes place on the same machine and may optionally be accompanied by a hardware upgrade. This is sometimes referred to as a *slip install*.

- The upgrade takes place on a different machine. This is sometimes referred to a *side-by-side install*.

Preparing for a slip install or side-by-side install

Before performing a slip install or side-by-side install carry out the following procedure:

1. Stop all applications that are using the existing version of MQSeries for AS/400.

2. End all MQSeries channels for all queue managers on the system. To do this, use the WRKMQMCHL command and select option 15.

3. On each queue manager, end the MQSeries command server. To do this, enter the command:

   ```
   ENDMQMCSVR MQMNAME(QMGRNAME) OPTION(*IMMED)
   ```

   where *QMGRNAME* is the name of the queue manager.

4. Remove any unwanted messages from your MQSeries queues.

5. Resolve any in-doubt messages that are held by sender or server channels. To do this, use the WRKMQMCHST command and select option 17.

6. Quiesce the product. To do this, refer to "Quiescing V5.1 or V5.2 of MQSeries for AS/400" on page 13.

7. On each queue manager, save the latest media recovery checkpoint. To do this enter the following four commands:

   ```
   STRMQM MQMNAME(QMGRNAME)
   RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(QMGRNAME)
   ENDMQM MQMNAME(QMGRNAME)
   ```

8. Quiesce the product again.

9. Copy your MQSeries CL and MQSC files to a suitable save library such as QGPL.
10. End the QMQM subsystem by entering the command:

    ENDSBS SBS(QMQM)

11. Save your MQSeries data:
    - Create a save file for every queue-manager library on your system. To do this, issue the command:

      CRTSAVF FILE(QGPL/queue-manager-library)

    - Save your queue-manager libraries into the save files. To do this, enter the commands:

      SAVLIB LIB(queue-manager-library) DEV(*SAVF) SAVF(QGPL/queue-manager-library)

    - Create a save file for MQSeries IFS data. To do this, issue the command:

      CRTSAVF FILE(QGPL/QMUSERDATA)

    - Save your MQSeries IFS data:

      SAV DEV('/QSYS.LIB/QGPL.LIB/MQUSERDATA.FILE') OBJ('/QIBM/UserData/mqm')

12. If you are transferring MQSeries to a new machine, transfer the save files to the new machine.

Performing a slip install

Follow this procedure.

1.
   a. Install MQSeries for AS/400, V5.2. See "Installation procedure" on page 11 for details.
   b. Verify the installation. See "Verifying the installation" on page 12 for details.
   c. Perform the tasks described in "Post installation tasks" on page 12.

The default TCP channel listener in MQSeries for AS/400, V5.2 is a nonthreaded listener, which starts nonthreaded channels. This is different from MQSeries for AS/400, V5.1 where a multi-threaded listener is used.

2. If you wish to run threaded TCP channel listeners in MQSeries for AS/400, V5.2, you must add an attribute to the CHANNELS stanza of each queue manager's qm.ini file. The attribute is ThreadedListener=Yes. You can find a discussion on the merits of threaded listeners in the *MQSeries V5.2 Release Guide*. Information about qm.ini files is in the *MQSeries System Administration* book.

V5.1 to V5.2 migration

Performing a side-by-side install

Follow this procedure.

1.

   a. Install MQSeries for AS/400, V5.2. See "Installation procedure" on page 11 for details.

   b. Verify the installation. See "Verifying the installation" on page 12 for details.

   c. Perform the tasks described in "Post installation tasks" on page 12.

2. Restore the queue-manager libraries for every queue manager:

   `RSTLIB SAVLIB(queue-manager-library) DEV(*SAVF) SAVF(QGPL/queue-manager-library)`

3. Restore the MQSeries IFS data:

   `RST DEV('/QSYS.LIB/QGPL.LIB/MQUSERDATA.FILE') OBJ('/QIBM/UserData/mqm')`

4. Associate the journal receivers by issuing WRKJRN command on the AMQAJRN file in each queue-manager library.

5. If you wish to set up your work management environment, job descriptions, and pools, see the *MQSeries for AS/400 System Administration* book for guidance. Otherwise, use the default setup.

Verifying the migration

Use the following procedure to check that you have migrated to MQSeries for AS/400, V5.2 successfully:

1. Make QMQMADM either the primary or a secondary group profile for your user profile. To do this, enter one of the following commands:

   ```
   CHGUSRPRF USRPRF(YOUR PROFILE) GRPPRF(QMQMADM)
   CHGUSRPRF USRPRF(YOUR PROFILE) SUPGRPPRF(QMQMADM)
   ```

2. Ensure that the MQSeries subsystem, QMQM, is active. To do this, issue the command:

   `DSPSBSD SBSD(QMQM/QMQM)`

   If it is inactive, issue the command:

   `STRSBS SBSD(QMQM/QMQM)`

3. Start a queue manager by issuing the command:

   `STRMQM MQMNAME(QMGRNAME)`

   Ensure that it is available. To check this, use the command:

   `DSPMQM MQMNAME(QMGRNAME)`

4. If applicable, verify that the queue manager and its objects have been successfully retained. Some of the commands that you might use to do this are:

DSPMQM	Display Message Queue Manager (MQM)
DSPMQMOBJN OBJ(*ALL)	Display MQM object names
WRKMQMCHL	Work with MQM channel
WRKMQMMSG	Work with MQM message
WRKMQMQ	Work with MQM queue

5. Use the commands DSPOBJAUT or DSPMQMAUT to review your authorizations to all types of MQSeries objects (*MQM, *Q, *ALSQ, *LCLQ, *RMTQ, *PRC, *CMD, *NMLIST).

Refer to "Chapter 6. Using MQSeries for AS/400, V5.2" on page 39 or to the online help for more information about these commands and how to use them.

After migration from V4R2M1 or earlier

If you have migrated from MQSeries for AS/400 Version 4 to MQSeries for AS/400 Version 5, when you are satisfied that the migration has completed successfully you may delete:

- MQSeries journals. To do this issue the commands:
  ```
  DLTJRN JRN(QUSRSYS/AMQAJRN)
  DLTJRN JRN(QUSRSYS/AMQRJRN)
  ```
- MQSeries data libraries: To do this issue the commands:
  ```
  DLTLIB LIB(QMQMDATA)
  DLTLIB LIB(QMQMPROC)
  ```

After migration from V5.1

If you have migrated from MQSeries for AS/400, V5.1 to MQSeries for AS/400, V5.2, when you are satisfied that the migration has completed successfully you may delete the saved data in the save files in QGPL.

Chapter 3. Applying maintenance to MQSeries for AS/400

This chapter tells you how to apply maintenance to MQSeries for AS/400.

Service updates for MQSeries for AS/400 are supplied as PTFs (Program Temporary Fixes). They may be supplied on a CD-ROM, or you may obtain them electronically as save files, which are normally stored in the QGPL library. For details of how to load and apply PTFs and read their cover letters, see the *AS/400 Basic System Operation, Administration, and Problem Handling manual*, SC41-5206.

Before loading and applying a PTF:
- Read the cover letter carefully to see if you need to take any special actions.
- Quiesce all queue managers, as described in "Quiescing V5.1 or V5.2 of MQSeries for AS/400" on page 13.

Reinstalling MQSeries for AS/400

You can reinstall MQSeries for AS/400, V5.2 without losing any of your data.

When you reinstall MQSeries for AS/400, the system checks whether the MQSeries configuration file (mqs.ini) exists. If the file exists, it is kept and used with the newly installed system. If the file does not exist, an empty mqs.ini file is placed in the directory /QIBM/UserData/mqm.

All data that you have in the UserData directory is referenced by the newly installed system. In addition, all the queue manager-associated libraries containing journal and receiver information are referenced by the new system.

Chapter 4. Deleting MQSeries for AS/400, V5.2

There are two ways of deleting MQSeries for AS/400:
- A standard deletion removes MQSeries product code but preserves user data.
- An entire deletion removes both MQSeries product code and user data.

Standard deletion

Perform a standard deletion of the MQSeries for AS/400 product if you wish to retain your user data, for example, because you intend to reinstall the product at a later date.

To perform this deletion:

1. Quiesce MQSeries for AS/400. (See "Quiescing V5.1 or V5.2 of MQSeries for AS/400" on page 13.)
2. End the MQSeries subsystem, by entering:
   ```
   ENDSBS SBS(QMQM)
   ```
3. Ensure that no locks are held on the library QMQM.
4. Use the Delete Licensed Program (DLTLICPGM) command to delete the base product (and also the samples if you chose to install them). To delete the base product and the samples enter:
   ```
   DLTLICPGM LICPGM(5733A38) OPTION(*ALL)
   ```

   To delete only the samples, you could enter:
   ```
   DLTLICPGM LICPGM(5733A38) OPTION(1)
   ```

Deleting MQSeries for AS/400 in this way deletes only the objects that belong to MQSeries. That is, the QMQM library and /QIBM/ProdData/mqm and its subdirectories. None of the queue manager journal libraries or IFS directories are removed.

Entire deletion

You can, if you wish, delete MQSeries entirely, including all user data. If you do this, save your user data first. It will not be recoverable.

To perform this deletion:

1. Using the WRKMQM command, select option 4 to delete all queue managers on the system.

Deleting MQSeries

2. Quiesce MQSeries for AS/400. (See "Quiescing V5.1 or V5.2 of MQSeries for AS/400" on page 13.)

3. End the MQSeries subsystem, by entering:

```
ENDSBS SBS(QMQM)
```

4. Delete the directory /QIBM/UserData/mqm and its subdirectories. Do this using the EDTF command and then use option 9 for the mqm directory.

   Note: If you do this, you will no longer have any information regarding your installation. Use this command with extreme caution. The format of the command is:

   ```
   EDTF STMF('/QIBM/UserData')
   ```

   Alternatively, you can delete /QIBM/UserData/mqm and its subdirectories by repeated use of the RMVLNK and RMVDIR commands.

5. Ensure that no locks are held on the library QMQM.

6. Use the Delete Licensed Program (DLTLICPGM) command to delete the base product (and also the samples if you chose to install them). To delete the base product and the samples enter:

```
DLTLICPGM LICPGM(5733A38) OPTION(*ALL)
```

7. Identify all the users who belong to the QMQMADM group. Use the DSPUSRPRF command to display a list of them. You need to remove the QMQMADM group profile from their user profiles before you can delete the QMQMADM user profile. The format of the command is:

```
DSPUSRPRF USRPRF(QMQMADM) TYPE(*GRPMBR)
```

8. For each of the user profiles QMQM and QMQMADM, use the WRKOBJOWN command to list all the objects owned by the profile. The format of the command is:

```
WRKOBJOWN USRPRF(PROFILE)
```

   You need to alter the ownership or delete the objects. Then you must delete the two user profiles.

Part 2. Getting started with MQSeries

Chapter 5. 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.

Introduction

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).

MQSeries configurations

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.

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 35 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)

MQSeries configurations

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 must be coordinated externally by another XA-compliant transaction manager such as IBM CICS®, IBM Transaction Server, IBM TXSeries™, Transarc Encina, or BEA Tuxedo.

On MQSeries for AS/400 queue-manager-coordinated global units of work are not supported. Queue-manager-coordinated local units of work can be used only when there is no commitment definition at the job level. Externally-coordinated global units of work can be used only when there is a commitment definition at the job level.

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.

Capabilities

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.

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.

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
2. Sending messages to administration queues, to achieve the same results as with the MQAI, using programmable command formats (PCFs)

Chapter 6. Using MQSeries for AS/400, V5.2

This chapter introduces MQSeries command sets and some basic tasks that you can perform to get yourself started with MQSeries for AS/400, V5.2. Tasks include creating and starting a queue manager, and creating, clearing, browsing, and deleting a local queue. The examples given use CL commands, but you can select the appropriate command from any of the supplied command sets.

Note: If you are using a DBCS-language version of MQSeries for AS/400, take care when running AS/400 commands and entering a description. You should enter quotation marks around descriptions that contain DBCS characters.

Before you start

Before you can use MQSeries for AS/400 you must start a subsystem. If you have not already done so, issue the command:

```
STRSBS SBSD(QMQM/QMQM)
```

Before you can use MQSeries CL commands, you must make your user profile a member of the group profile QMQMADM. Use the CHGUSRPRF command to change your user profile, and specify QMQMADM against either the group profile (GRPPRF) parameter or the supplementary group profile (SUPGRPPRF) parameter.

It is not essential for your user profile to belong to the QMQMADM group profile for issuing PCF commands from an administration program or MQI calls from an application program.

Some MQSeries CL commands, for example CRTMQM (create queue manager) and DLTMQM (delete queue manager), rely on using AS/400 system commands for creating and managing objects, files, and libraries. The commands listed below must either have *PUBLIC *USE authority, or explicit *USE authority to the MQSeries user profiles QMQM and QMQMADM:
- ADDLIBLE
- ADDPFM
- CHGCURLIB
- CHGJOB
- CRTJRN
- CRTJRNRCV
- CRTJOBQ
- CRTJOBD

Before you start

- CRTLIB
- CRTMSGQ
- CRTPF
- CRTSRCPF
- DLTJRN
- DLTJRNRCV
- DLTLIB
- DLTMSGQ
- OVRPRTF
- RCLACTGRP

If you have installed the samples, ensure that *all* users have access to the QMQMSAMP library. Do this using the GRTOBJAUT command:

```
GRTOBJAUT OBJ(QMQMSAMP) OBJTYPE(*LIB) USER(*PUBLIC) AUT(*USE)
```

For details of MQSeries Work Management and security, see the *MQSeries for AS/400 System Administration* book.

Introducing command sets

MQSeries provides three command sets for performing administration tasks:
- Command Language commands (CL commands)
- MQSC commands
- PCF commands

CL commands

The AS/400 CL commands for MQSeries can be grouped as follows:
- *Channel commands* - used to create, start, and delete channels.
- *Cluster commands* - used to work with queue manager clusters.
- *Command server commands* - used to display, start, and end the command server.
- *Data type conversion command* - used to convert data types.
- *Dead-letter queue handler command* - used to start the dead-letter queue handler.
- *Media recovery commands* - used to record and re-create objects.
- *MQSeries command* - used to start a batch of MQSeries commands.
- *Name command* - used to display object names.
- *Namelist commands* - used to work with namelists.
- *Process commands* - used to change, copy, and delete processes.
- *Queue commands* - used to create, delete, and work with queues.
- *Queue manager commands* - used to connect, create, and delete queue managers.
- *Security commands* - used to display, grant, and revoke object authority.
- *Trace command* - used to trace service jobs.
- *Transaction commands* - used to display and resolve transactions.
- *Trigger monitor commands* - used to start trigger monitors.

Using CL commands
You can enter CL commands from a command line.

The following table gives a brief description of each MQSeries CL command. To display online help for a CL command, enter the command and press F1. To see a full list of MQSeries commands, enter the command GO CMDMQM from the MQSeries menu.

Command	Usage
CCTMQM (connect message queue manager)	supported only for compatibility with previous releases.
CHGMQM (change message queue manager)	to change the attributes of a message queue manager.
CHGMQMCHL (change MQM channel)	to change the attributes of an existing channel.
CHGMQMNL (change MQM namelist)	to change the attributes of a namelist.
CHGMQMPRC (change MQM process)	to change the attributes of a process definition.
CHGMQMQ (change MQM queue)	to change the attributes of a queue.
CLRMQMQ (clear MQM queue)	to delete all messages from a local queue.
CPYMQMCHL (copy MQM channel)	to create a copy of an existing channel definition.
CPYMQMNL (copy MQM namelist)	to create a copy of a namelist.
CPYMQMPRC (copy MQM process)	to create a copy of a process definition.
CPYMQMQ (copy MQM queue)	to create a copy of an existing queue definition.
CRTMQM (create message queue manager)	to create a local queue manager.
CRTMQMCHL (create MQM channel)	to create a new channel definition.
CRTMQMNL (create MQM namelist)	to create a new namelist.
CRTMQMPRC (create MQM process)	to create a new process definition.
CRTMQMQ (create MQM queue)	to create a queue definition.
CVTMQMDTA (convert MQM data type command)	to create a fragment of code that performs data conversion on data type structures.
DLTMQM (delete message queue manager)	to delete a local queue manager.
DLTMQMCHL (delete MQM channel)	to delete a channel definition.
DLTMQMNL (delete MQM namelist)	to delete a namelist.
DLTMQMPRC (delete MQM process)	to delete a process definition.
DLTMQMQ (delete MQM queue)	to delete a queue.
DSCMQM (disconnect message queue manager)	supported only for compatibility with previous releases.
DSPMQM (display message queue manager)	to display the attributes of a local queue manager.
DSPMQMAUT (display MQM object authority)	to display a list of authorized users and their authorities for a specified object.
DSPMQMCHL (display MQM channel)	to display the attributes of a channel definition.
DSPMQMCSVR (display MQM command server)	to display the status of the command server.
DSPMQMNL (display MQM namelist)	to display the attributes of a namelist.
DSPMQMOBJN (display MQM object names)	to display the AS/400 name and type for any object name and type.
DSPMQMPRC (display MQM process)	to display the attributes of a process definition.
DSPMQMQ (display MQM queue)	to display the attributes of a queue definition.

Introducing command sets

Command	Usage
ENDMQM (end message queue manager)	to end a message queue manager.
ENDMQMCHL (end MQM channel)	to close a channel.
ENDMQMCSVR (end MQM command server)	to stop the command server for a specified local queue manager.
ENDMQMLSR (end MQM listener)	to stop the MQSeries listener program.
GRTMQMAUT (grant MQM object authority)	to grant specific authority for an object to other users.
PNGMQMCHL (ping MQM channel)	to test a channel.
RCDMQMIMG (record MQM object image)	to provide a marker for a selected set of objects.
RCRMQMOBJ (re-create MQM object)	to provide recovery for a damaged set of objects.
RFRMQMAUT (refresh MQSeries authority)	to refresh a user's authorizations to MQSeries objects.
RFRMQMCL (refresh cluster)	to refresh the cluster information that a queue manager has stored.
RSMMQMCLQM (resume cluster queue manager)	to inform cluster queue managers that the local queue manager has become available.
RSTMQMCHL (reset MQM channel)	to reset the message sequence number for a channel to a specific number.
RSTMQMCL (reset queue manager cluster)	to reset the information held in a cluster about its queue managers.
RSVMQMCHL (resolve MQM channel)	to request a channel to commit or back out in-doubt messages.
RVKMQMAUT (revoke MQM authority)	to remove authority for named objects.
SPDMQMCLQM (suspend cluster queue manager)	to inform cluster queue managers that the local queue manager is not available.
STRMQM (start message queue manger)	to start a local queue manager.
STRMQMCHL (start MQM channel)	to start a channel.
STRMQMCHLI (start MQM channel initiator)	to start a channel initiator.
STRMQMCSVR (start MQM command server)	to start a command server for a queue manager.
STRMQMDLQ (start MQSeries dead-letter queue handler)	to start the dead-letter queue handler.
STRMQMLSR (start MQM listener)	to start a message queue manager listener.
STRMQMMQSC (start MQSeries commands)	to initiate a set of MQSeries commands.
STRMQMTRM (start trigger monitor)	to start a trigger monitor.
TRCMQM (trace MQM job)	to start and stop MQSeries trace.
WRKMQM (work with message queue managers)	to work with queue managers.
WRKMQMCHL (work with MQM channel)	to work with channel definitions.
WRKMQMCHST (work with MQM channel status)	to work with the status of channel definitions.
WRKMQMCL (work with clusters)	to work with clusters.
WRKMQMCLQ (work with cluster queues)	to work with cluster queues.
WRKMQMMSG (work with MQM messages)	to work with specified messages on a local queue.
WRKMQMNL (work with MQM namelists)	to work with namelists.
WRKMQMPRC (work with MQM process)	to work with process definitions.

Command	Usage
WRKMQMQ (work with MQM queues)	to work with queues.
WRKMQMTRN (work with MQSeries transactions)	to work with in-doubt MQSeries transactions.

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.

When you display a queue, using the DISPLAY QUEUE command, you display the queue *attributes*. For example, the MAXMSGL attribute specifies the maximum length of a message that can be put on the queue. The command does not show you the messages on the queue.

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

Running MQSC commands

You run MQSC commands by entering the CL command STRMQMMQSC on the command line. This allows you to run a series of commands that you have previously entered in a member of a source physical file.

PCF commands

MQSeries programmable command format (PCF) commands allow you to program administration tasks into an administration program. For a complete description of the PCF commands and how to use them, see the *MQSeries Programmable System Management* book.

Working with MQSeries

MQSeries for AS/400, V5.2 has a full compliment of AS/400 commands with which you can administer the product. These commands can be used directly from the command line or alternatively can be accessed by a hierarchical command structure based on the familiar 'Work with' command concept. At the top of this structure is the WRKMQM command. Use this command to display all queue managers defined to the system and to allow you maintain existing queue managers or add new ones.

An example of the WRKMQM panel, obtained if you issue the WRKMQM command with default parameters, (that is, to work with all queue managers) is shown in Figure 1 on page 44.

Working with MQSeries

```
                       Work with queue managers

Type options, press Enter.
  2=Change   4=Delete   5=Display   14=Start   15=End   18=Work with Queues
  19=Work with Processes   20=Work with Channels   21=Work with NameLists

Opt  Name                                          Status  Default
  __   EUROPE                                        INACTIVE   NO
  __   ASIA                                          INACTIVE   NO
  __   US                                             ACTIVE    NO
  __   MQDEFAULT                                     INACTIVE   YES
  __   MPHTEST_REFMSG                                  ACTIVE    NO




                                                               Bottom


Parameters for options 2, 4, 5, 14, 15, 18, 19, 20, 21 or command
===> _____
F3=Exit    F4=Prompt       F5=Refresh   F6=Create   F9=Retrieve   F12=Cancel
F16=Repeat position to   F17=Position to             F20=Right    F21=Print
```

Figure 1. WRKMQM panel

Starting from this command, at the top of the hierarchy, you can step down to other commands that enable you to work with the different categories of queue manager objects. If you are familiar with the previous 'Administration Utility' you should find this particularly easy to use, although it does not allow administration of remote queue managers or clusters in its present form.

As you can see, you can use WRKMQM as a *way in* to any of the other panels that are mentioned in the rest of this section. For example, to create a queue manager, you could use the CRTMQM command as described below, or you could use F6 from the WRKMQM panel. To create a local queue, you could use the CRTMQMQ command, or you could select option 18 from the WRKMQM panel. Experiment with WRKMQM as well as entering the commands themselves when you carry out the following tasks.

Creating a queue manager

If you are new to MQSeries for AS/400, you are recommended to set up a test system. Then you can verify that you have installed the product correctly and perform some basic MQSeries operations.

To set up a test system you need to create a queue manager and a local queue. Create these objects by entering the CL commands from the command line or by working from the WRKMQM panel.

To create a queue manager:

1. Issue the CRTMQM command from the command line and press F4. The screen shown in Figure 2 appears.

```
                    Create Message Queue Manager (CRTMQM)

 Type choices, press Enter.

 Message Queue Manager name . . .   _____
 ____
 Text 'description' . . . . . . .   *BLANK_____
 _____
 Trigger interval . . . . . . . .   999999999__   0-999999999
 Undelivered message queue  . . .   *NONE_____
 _____
 Default Transmission queue . . .   *NONE_____
 _____
 Maximum handle limit . . . . . .   256_____    1-999999999
 Maximum uncommitted messages . .   10000_____    1-10000
 Default Queue Manager  . . . . .   *NO           *YES, *NO



                                                                     Bottom
  F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
  F24=More keys
  Parameter MQMNAME required.
```

Figure 2. CRTMQM panel

If the screen does not appear, it may be that you have not installed MQSeries for AS/400 correctly. If this is the case, delete the product as described in "Chapter 4. Deleting MQSeries for AS/400, V5.2" on page 27. Then reinstall the *BASE product and try the CRTMQM command again.

2. On the CRTMQM panel, type in a Message Queue Manager name, for example TEST.QMANAGER and press Enter.

   The name of the queue manager can be a maximum of 48 characters in length. Valid characters are detailed in *MQSeries for AS/400 System Administration* book.

   The queue manager is then created, along with a set of default objects.

Starting a queue manager

Start the queue manager from the command line by issuing the command:
STRMQM MQMNAME(*QMGRNAME*)

where *QMGRNAME* is the name of the queue manager that you have just created.

After a short period you receive the message Message Queue Manager started.

Creating a local queue

To create a local queue:

1. Type CRTMQMQ on the command line and press F4. This displays the following panel:

```
                      Create MQM Queue (CRTMQMQ)

 Type choices, press Enter.

 Queue name . . . . . . . . . .    _____
 _____
 Queue type . . . . . . . . . .                 *ALS, *LCL, *MDL, *RMT
 Message Queue Manager name . . .  *DFT_____
 _____




                                                                Bottom
 F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
 F24=More keys
 Parameter QNAME required.
```

Figure 3. CRTMQMQ panel (part 1)

2. On the Create MQM Queue panel, type the name of the queue you want to create in the Queue name field. For example TEST.QUEUE. To specify a mixed-case name, enclose the name in apostrophes.
3. Type *LCL in the Queue type field.
4. Fill in the name of your queue manager in the Message Queue Manager name field. For example TEST.QMANAGER.
5. Press Enter.

There are three more parts to the CRTMQMQ panel. Scroll through them and enter values for the other options if you wish, or otherwise just accept all the defaults. The panels are shown in Figure 4 on page 47 through Figure 6.

```
                    Create MQM Queue (CRTMQMQ)

 Type choices, press Enter.

 Queue name . . . . . . . . . . > TEST.QUEUE_____
 ____
 Queue type . . . . . . . . . . > *LCL          *ALS, *LCL, *MDL, *RMT
 Message Queue Manager name . . > TEST.QMANAGER_____
 _____
 Replace . . . . . . . . . . . .  *NO           *NO, *YES
 Text 'description' . . . . . . .  ' '_____
 _____
 Put enabled . . . . . . . . . .  *YES          *SYSDFTQ, *NO, *YES
 Default message priority . . . .  0            0-9, *SYSDFTQ
 Default message persistence . .  *NO           *SYSDFTQ, *NO, *YES
 Process name . . . . . . . . . .  ' '_____
 _____
 Triggering enabled . . . . . . .  *NO           *SYSDFTQ, *NO, *YES
 Get enabled . . . . . . . . . .  *YES          *SYSDFTQ, *NO, *YES
 Sharing enabled . . . . . . . .  *YES          *SYSDFTQ, *NO, *YES
                                                               More...
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 4. Create MQM Queue panel (part 2)

```
                    Create MQM Queue (CRTMQMQ)

 Type choices, press Enter.

 Default share option . . . . . .  *YES          *SYSDFTQ, *NO, *YES
 Message delivery sequence . . .  *PTY          *SYSDFTQ, *PTY, *FIFO
 Harden backout count . . . . . .  *YES          *SYSDFTQ, *NO, *YES
 Trigger type . . . . . . . . . .  *FIRST        *SYSDFTQ, *FIRST, *ALL...
 Trigger depth . . . . . . . . .  1             1-999999999, *SYSDFTQ
 Trigger message priority . . . .  0            0-9, *SYSDFTQ
 Trigger data . . . . . . . . . .  ' '_____
 _____
 Retention interval . . . . . . .  999999999     0-999999999, *SYSDFTQ
 Maximum queue depth . . . . . .  5000          0-640000, *SYSDFTQ
 Maximum message length . . . . .  4194304       0-4194304, *SYSDFTQ
 Backout threshold . . . . . . .  0             0-999999999, *SYSDFTQ
 Backout requeue name . . . . . .  ' '_____
 _____
 Initiation queue . . . . . . . .  ' '_____
 _____
                                                               More...
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 5. Create MQM Queue panel (part 3)

```
                    Create MQM Queue (CRTMQMQ)

 Type choices, press Enter.

 Usage  . . . . . . . . . . . .    *NORMAL       *SYSDFTQ, *NORMAL, *TMQ
 Queue depth high threshold . . .  80            0-100, *SYSDFTQ
 Queue depth low threshold  . . .  20            0-100, *SYSDFTQ
 Queue full events enabled  . . .  *YES          *SYSDFTQ, *NO, *YES
 Queue high events enabled  . . .  *YES          *SYSDFTQ, *NO, *YES
 Queue low events enabled . . . .  *YES          *SYSDFTQ, *NO, *YES
 Service interval . . . . . . . .  999999999     0-999999999, *SYSDFTQ
 Service interval events  . . . .  *NONE         *SYSDFTQ, *HIGH, *OK, *NONE
 Distribution list support  . . .  *NO           *SYSDFTQ, *NO, *YES
 Cluster Name . . . . . . . . . .  ' '_____

 ____
 Cluster Name List  . . . . . . .  ' '_____

 ____
 Default Binding  . . . . . . . .  *OPEN         *SYSDFTQ, *OPEN, *NOTFIXED


                                                                 Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 6. Create MQM Queue panel (part 4)

When you have finished reviewing and making changes to the values, press Enter to create the queue.

Sending a test message

To put a test message on your queue use the supplied sample programs. For example, to put messages to a queue called TEST.QUEUE, defined for a queue manager called TEST.QMANAGER, enter the command:

CALL PGM(QMQM/AMQSPUT0) PARM(TEST.QUEUE TEST.QMANAGER)

You can now type a message (a simple character string is sufficient) and press Enter to send the message to the queue. Press Enter again to return to a command line.

To see the message on the queue, either:

- Browse the queue (see "Browsing queues"), or
- Type CALL PGM(QMQM/AMQSGET4) PARM(TEST.QUEUE TEST.QMANAGER) on a command line.

See "Appendix A. Sample MQI programs" on page 61 for a list of the supplied sample programs.

Browsing queues

To browse the messages on a local queue called TEST.QUEUE, defined for a queue manager called TEST.QMANAGER, use the following command:

WRKMQMMSG MQMNAME(*TEST.QMANAGER*) QNAME(*TEST.QUEUE*)

This command displays all messages on the queue, and allows you to view the message contents.

Note: If your messages contain DBCS data, note that DBCS characters are not displayed correctly in the Text field. However, the correct hexadecimal values are displayed in the Hexadecimal field.

Clearing a local queue

To delete all the messages on a local queue called TEST.QUEUE, defined for a queue manager called TEST.QMANAGER, use the following command:

CLRMQMQ MQMNAME(*TEST.QMANAGER*) QNAME(*TEST.QUEUE*)

Deleting a local queue

To delete a local queue called TEST.QUEUE, defined for a queue manager called TEST.QMANAGER, use the following command:

DLTMQMQ MQMNAME(*TEST.QMANAGER*) QNAME(*TEST.QUEUE*)

Note: You must delete all messages from the queue using the CLRMQMQ command before you can delete a queue.

Stopping a queue manager

There are two ways of quiescing a queue manager:

1. Either use the command ENDMQM MQMNAME(*QMGRNAME*), where *QMGRNAME* is the name of the queue manager, or
2. Enter the command WRKMQM *ALL, enter option 6 (end) against the queue manager name, and then press F4.

There are 4 options you can use when ending a queue manager:

1. OPTION (*CNTRLD). This allows current transactions to complete, and all applications to disconnect, before ending the queue manager. This option is the default.
2. OPTION (*IMMED). This immediate shutdown allows any current calls to complete, but stops any new calls. It does not wait for applications to disconnect from the queue manager.
3. OPTION (*WAIT). This performs the same type of shutdown as the (*CNTRLD) option, but gives you confirmation that the queue manager has ended. The command prompt does not return until the queue manager has ended. You cannot use this option if you specify MQMNAME(*ALL*).
4. OPTION (*PREEMPT). This preemptive shutdown ends all queue manager code immediately. It can have unpredictable consequences and should not be used unless all other options have failed.

Working with MQSeries

Deleting a queue manager

To delete a queue manager, first stop it, then issue the command:

DLTMQM MQMNAME(*QMGRNAME*)

Note: When you delete a queue manager you also delete all resources associated with that queue manager, including all queues, all messages, and all object definitions.

Checking what queue managers you have running

Sometimes you may want to remind yourself how many queue managers you have running. You can do this using the WRKMQM command. Enter the command with no parameters. A list of all configured queue managers will be displayed, showing which are active and which are inactive.

Other things to do

The test system set up in "Working with MQSeries" on page 43 allows you to manage a single queue manager and queue. In reality you will need many more tasks and MQSeries objects to use MQSeries effectively. For example, you must also create:

- A remote queue (that is, a local definition for a queue on a remote queue manager)
- A transmission queue
- Sender and receiver channels

You also need to start a channel listener if you are using TCP, and you may want to start a channel initiator too. Refer to the *MQSeries for AS/400 System Administration* book, for details of how to set up and use these objects.

User exits

Before using your user exits on MQSeries for AS/400, V5.2, consider whether you plan to run your channel listeners as threaded or nonthreaded processes. If you intend to use threaded channel listeners, you must relink your user exits with the threaded libraries to ensure that they are thread-safe and teraspace enabled. See the *MQSeries Application Programming Guide* and the *MQSeries Intercommunication* book for further information about creating user exit programs.

Setting the queue manager CCSID for MQSeries for AS/400

When you create a queue manager, its coded character set identifier (CCSID) is set by default. It is set to the value of the CCSID of the OS/400 job you used to create the queue manager.

You can query the value of this CCSID by entering the DSPMQM CL command. This displays the attributes of the local queue manager, including the coded character set identifier.

You can modify the queue manager's CCSID by using the CHGMQM CL command. Before doing this, make a note of the original CCSID value in case you need to reset it at a later date. To change the CCSID, follow this procedure:

1. Issue the DSPMQM command and make a note of the CCSID value.
2. Change the CCSID to the new CCSID, with the command:

   ```
   CHGMQM CCSID(value)
   ```
3. Stop the queue manager with the ENDMQM command.
4. Restart the queue manager with the STRMQM command and also restart any channels that it uses.

For information about code sets and CCSIDs that are supported by MQSeries for AS/400 see the *AS/400 National Language Support* book. See the *MQSeries MQSC Command Reference* for more information about MQSeries commands.

Chapter 7. Obtaining additional information

This chapter describes the documentation for MQSeries for AS/400. It starts with a list of the publications, and then discusses:
- "Hardcopy books"
- "Online information" on page 54

MQSeries for AS/400 is described in the following books:

Table 5. MQSeries for AS/400 books

Order Number	Title
AS/400 Specific Books	
GC34-5557	*MQSeries for AS/400 Quick Beginnings*
SC34-5558	*MQSeries for AS/400 System Administration*
GC34-5559	*MQSeries for AS/400 Application Programming Reference (ILE RPG)*
MQSeries Family Books	
GC34-5761	*MQSeries V5.2 Release Guide*
SC33-1872	*MQSeries Intercommunication*
SC34-5349	*MQSeries Queue Manager Clusters*
GC33-1632	*MQSeries Clients*
SC33-1369	*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 AS/400, 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 are available for you to order or print.

Hardcopy books

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 55.

Online information

This section describes:
- "Publications supplied with the product"
- "HTML and PDF books on the World Wide Web" on page 57
- "BookManager CD-ROMs" on page 57
- "Online help" on page 57

Publications supplied with the product

Included with the product is a publications CD-ROM. On this CD-ROM there are three directories; books, amqaa60w, and readme.

CD-ROM books directory
The books directory contains MQSeries books in HTML and PDF formats. To access them point your Web browser to books/start.htm.

HTML
You can view the MQSeries online documentation in HTML format directly from the CD-ROM. All books except for the *MQSeries Programming Interfaces Reference Summary* are available in U.S. English and also in some or all of the following national languages:
- Brazilian Portuguese
- French
- German
- Italian
- Japanese
- Korean
- Spanish
- Simplified Chinese
- Traditional Chinese

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

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 books/start.htm.

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 AS/400 Quick Beginnings	AMQWAC01
MQSeries for AS/400 System Administration	AMQWAG00

Online information

Table 6. MQSeries publications – file names (continued)

Book	File Name
MQSeries for AS/400 Application Programming Reference (ILE RPG)	AMQWAK00
MQSeries V5.2 Release Guide	AMQZAY00
MQSeries Intercommunication	CSQZAE04
MQSeries Queue Manager Clusters	CSQZAH02
MQSeries Clients	CSQZAF04
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

CD-ROM amqaa60w directory
The amqaa60w directory contains a package of searchable PDF files. The package comprises a set of panels, books, and index files and provides a method of searching the MQSeries library on your workstation using Adobe Acrobat Reader with Search. The package is available in U.S. English only.

In addition to providing a search facility, the package offers the following advantages:
- Full cross-referencing within books that allows you to jump to other parts of the same document
- Hyperlinked URLs to enable transparent integration with the Web
- Thumbnail images of all pages in the book to allow you to scan for diagrams, tables and so on

To take advantage of the usability enhancements provided in this package, you need Adobe Acrobat Reader with Search Version 4.

To use the package, start Adobe Acrobat Reader and open the file amqab400.pdf located in the amqaa60w/panels directory on the publications CD-ROM.

The package is also supplied in a zip file and a tar file. If you wish, you can unpack these onto your own file system and access them from there instead of directly from the CD-ROM.

CD-ROM readme directory
The readme directory contains, in each national language, a file of up-to-date product information that you should read before installing or using this product.

There might also be a file of latest information (in U.S. English only) that became available after the readme files were translated.

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.

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

MQSeries for AS/400 provides context-sensitive help. If you place your cursor on a command or panel field and press PF1, you are presented with relevant help information.

Part 3. Appendixes

Appendix A. Sample MQI programs

MQSeries for AS/400 provides a set of short sample MQI programs. You can choose to install these when you install the base product. If you do install them, you get the C samples in both source form and object form, and the COBOL and RPG samples in source form only. You can use the C objects as they are or you can modify and recompile the source for experimental purposes.

C, COBOL, and RPG program samples

Table 7 lists the sample MQI source files. By default, the source files are in the QMQMSAMP library, which is available as an installable option. To find out more about what the programs do and how to use them, see the *MQSeries Application Programming Guide*.

Table 7. Sample programs – source files

Technique	C	COBOL	RPG
Putting messages using MQPUT	AMQSPUT0	AMQ0PUT4	AMQnPUT4
Putting messages from a data file using MQPUT	AMQSPUT4	–	–
Putting a single message using MQPUT1	AMQSINQ4 AMQSECH4	AMQ0INQ4 AMQ0ECH4	AMQnINQ4 AMQnECH4
Putting messages to a distribution list	AMQSPTL4	–	–
Replying to a request message	AMQSINQ4	AMQ0INQ4	AMQnINQ4
Getting messages (no wait)	AMQSGBR4	AMQ0GBR4	AMQnGBR4
Getting messages (wait with time limit)	AMQSGET4	AMQ0GET4	AMQnGET4
Getting messages (unlimited wait)	AMQSTRG4	–	AMQ3TRG4
Getting messages (with data conversion)	AMQSECH4	AMQ0ECH4	AMQnECH4
Putting reference messages to a queue	AMQSPRM4	–	–
Getting reference messages from a queue	AMQSGRM4	–	–
Using reference message channel exit	AMQSQRM4 AMQSXRM4	–	–

Samples

Table 7. Sample programs – source files (continued)

Technique	C	COBOL	RPG
Using message exit	AMQSCMX4	–	–
Browsing first 20 characters of a message	AMQSGBR4	AMQ0GBR4	AMQnGBR4
Browsing complete messages	AMQSBCG4	–	–
Using a shared input queue	AMQSINQ4	AMQ0INQ4	AMQnINQ4
Using an exclusive input queue	AMQSREQ4	AMQ0REQ4	AMQnREQ4
Using the MQINQ call	AMQSINQ4	AMQ0INQ4	AMQnINQ4
Using the MQSET call	AMQSSET4	AMQ0SET4	AMQnSET4
Using a reply-to queue	AMQSREQ4	AMQ0REQ4	AMQnREQ4
Requesting message exceptions	AMQSREQ4	AMQ0REQ4	AMQnREQ4
Accepting a truncated message	AMQSGBR4	AMQ0GBR4	AMQnGBR4
Using a resolved queue name	AMQSGBR4	AMQ0GBR4	AMQnGBR4
Triggering a process	AMQSTRG4	–	AMQ3TRG4
Using a trigger server	AMQSERV4	–	AMQ3SRV4
Using a trigger server (including CICS transactions)	AMQSERV4	–	AMQ3SRV4
Using data conversion	AMQSVFC4	–	–

Notes:

1. Source for the ILE C samples is in QMQMSAMP/QCSRC. Include files exist as members in the file QMQM/H.

2. Source for the COBOL samples is in QMQMSAMP/QLBLSRC for the OPM compiler, and QMQMSAMP/QCBLLESRC for the ILE compiler. Copy members exist in QMQM/QLBLSRC and QMQM/QCBLLESRC respectively. The members are named AMQ0xxx4 for OPM, and AMQ5xxx4 for ILE. xxx indicates the sample function.

3. There are three sets of RPG sample programs:

   a. OPM RPG programs. The source for these is in QMQMSAMP/QRPGSRC. The members are named AMQ1xxx4, where xxx indicates the sample function. Copy members exist in QMQM/QRPGSRC.

   b. ILE RPG programs using the MQI through a call to QMQM. The source for these is in QMQMSAMP/QRPGLESRC. The members are named AMQ2xxx4, where xxx indicates the sample function. Copy members exist in QMQM/QRPGLESRC. Each member name has a suffix of 'R'.

c. ILE RPG programs using prototyped calls to the MQI. The source for these is in QMQMSAMP/QRPGLESRC. The members are named AMQ3xxx4, where xxx indicates the sample function. Copy members exist in QMQM/QRPGLESRC. Each member name has a suffix of 'G'.

Samples

Appendix B. 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

Notices

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	IBM	IBMLink
MQSeries	OS/390	OS/400
TXSeries	VisualAge	VSE/ESA
WebSphere		

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

Java is a registered trademark of Sun Microsystems, Inc. in the United States, other countries, or both.

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

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

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

Index

ordering publications   53

P

PCF commands   43
PDF (Portable Document
   Format)   55
performance events   38
Portable Document Format
   (PDF)   55
printing books   55
program samples
   C, COBOL, and RPG   61
program temporary fix (PTF)   25
programmable command format
   (PCF)
      administration with   43
programming with MQSeries   38
PTF (program temporary fix)   25
publications   53

Q

QALWOBJRST   9
QCCSID   9
qm.ini file   5
QMQMADM group profile   39
QSYSLIBL   9
queue manager
   active   50
   creating   45
   description   33
   events   37
   monitoring   37
   quiescing   49
   starting   45
queues
   attributes   32
   browsing   48
   description   32
   local
      clearing   49
      deleting   49
quiescing   13
quiescing a queue manager   49
QUTCOFFSET   9

R

readme file   7
reinstalling   25
remote queue   33
remote queue manager   33
requirements
   disk storage   3
   MQSeries for AS/400, V5.2
      software   3
RPG sample programs   61
RSTLICPGM command   11

S

sample programs
   C, COBOL, and RPG   61
sending a test message   48
server-client configurations   35
setting system values   9
setting the CCSID (coded character
   set identifier)   50
side-by-side install   22
slip install   21
softcopy information   54
software requirements
   MQSeries for AS/400, V5.2   3
starting a queue manager   45
stopping a queue manager   49
STRMQMMQSC command   43
system values   9

T

threaded channel listener   21
transactional support   36
triggering   38

U

user exits, linking   50
user profile   6
   granting authority   39

V

verifying installation   12
verifying migration   22

W

what's new   15

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.

IBM®