MQSeries®

Messages

Third edition (December 2000)

This edition applies to the following products:
- MQSeries for AIX® Version 5.2
- MQSeries for HP-UX Version 5.2
- MQSeries for Linux Version 5.2
- MQSeries for Sun Solaris Version 5.2
- MQSeries for Windows NT® and Windows® 2000 Version 5.2
- MQSeries for Windows Version 2.0
- MQSeries for Windows Version 2.1

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

Contents

About this book

This book describes the user messages returned by the MQSeries products listed below, with explanations and suggested actions. It is designed for use as a quick reference.

This book applies to the following MQSeries products:
- MQSeries for AIX, V5.2
- MQSeries for HP-UX, V5.2
- MQSeries for Linux, V5.2
- MQSeries for Sun Solaris, V5.2
- MQSeries for Windows NT and Windows 2000, V5.2
- MQSeries for Windows V2.0
- MQSeries for Windows V2.1

UNIX® systems

In this book, references to "UNIX systems" include:
    AIX
    HP-UX
    Linux
    Sun Solaris

References to MQSeries for "UNIX systems" include:
    IBM® MQSeries for AIX, V5.2
    IBM MQSeries for HP-UX, V5.2
    IBM MQSeries for Linux, V5.2
    IBM MQSeries for Sun Solaris, V5.2

Who this book is for

The information is intended for system operators, system programmers, and anyone who needs to understand and take action in response to MQSeries user messages.

How to use this book

See "Chapter 1. Finding and reading a message" on page 1.

Summary of changes

This section describes changes in this edition of *MQSeries Messages*. Changes since the previous edition of the book are marked by vertical lines to the left of the changes.

Changes for this edition (GC33-1876-02)

The changes include:

- Addition of messages specific to the MQSeries V5.2 products
- Minor technical and editorial improvements throughout the book

Changes for the previous edition (GC33-1876-01)

Changes included:

- Messages for cluster support on queue managers
- A section describing all the Message Queue Interface return codes
- Minor technical and editorial improvements throughout the book

Changes

Chapter 1. Finding and reading a message

MQSeries user messages are numbered AMQ4000 through AMQ9999, and they are listed in this book in numeric order. However, not all numbers have been used, and therefore, the list is not continuous.

Finding a message

MQSeries messages are grouped according to the part of MQSeries from which they originate:

"Chapter 2. AMQ3500-AMQ3999 MQSeries for Windows messages" on page 3.

"Chapter 3. AMQ4000-AMQ4999 MQSeries for Windows NT User Interface messages" on page 9.

"Chapter 4. AMQ5000-AMQ5999 Installable services messages" on page 29.

"Chapter 5. AMQ6000-AMQ6999 Common services messages" on page 37.

"Chapter 6. AMQ7000-AMQ7999 MQSeries product messages" on page 45.

"Chapter 7. AMQ8000-AMQ8999 MQSeries administration messages" on page 65.

"Chapter 8. AMQ9000-AMQ9999 Remote messages" on page 85.

The return codes for the MQSeries programming interface, the MQI, are shown in tabular form in "Chapter 9. API completion and reason codes" on page 103.

Reading a message

The format of an MQSeries user message is:

- The message identifier, in two parts:
  1. The characters 'AMQ' which identify the message as being from MQSeries
  2. A four-digit decimal code
- The text of the message

Message information

For each message, this information is provided:

Explanation:
    Why the message was issued.

User action:
    Instructions to the user.

Message variables

Some messages display text or numbers that vary according to the circumstances giving rise to the message; these are known as *message variables*. The message variables are indicated in this book by the use of the '&' symbol and a number: &1, &2, and so on.

In some cases a message may have variables in the Explanation or User action. Find the values of the message variables by looking in the error log. The complete message, including the Explanation and the User action, is recorded there.

Chapter 2. AMQ3500-AMQ3999 MQSeries for Windows messages

> **Reading variables in the message**
> In some messages, the text includes strings or numbers that vary depending on the circumstances that cause the message. These message variables are shown in the text of the message like *this*.

AMQ3500   *primary-keyword secondary-keyword name* **was successful.**
Explanation: The operation was successful.
User Response: None required.

AMQ3501   *character* **- string expected.**
Explanation: The first quotation mark of a quoted string was expected, but the character *character* was found.
User Response: Enclose the string in quotation marks and retry the command.

AMQ3502   *integer* **- value out of range.**
Explanation: The specified integer value for an attribute is outside the allowed range.
User Response: Change the integer to a value that is within the allowed range and retry the command.

AMQ3503   *primary-keyword secondary-keyword name* **failed. Return code =** *code***.**
Explanation: The MQSC command is syntactically correct, but the queue manager could not perform the command.
User Response: Look up the return code in the *MQSeries for Windows User's Guide*, correct the problem, then retry the command.

AMQ3504   *string* **- attribute keyword expected.**
Explanation: An attribute keyword was expected, but *string* was found.
User Response: Ensure the attribute keyword is valid and spelled correctly, then retry the command.

AMQ3505   *string* **- left parenthesis expected.**
Explanation: A left parenthesis was expected, but *string* was found.
User Response: Correct the command, adding a left parenthesis, then retry the command.

AMQ3506   *string* **- number expected.**
Explanation: An attribute requires an integer value, but *string* was found.

User Response: Correct the attribute value, then retry the command.

AMQ3507   *string* **- right parenthesis expected.**
Explanation: A right parenthesis was expected, but *string* was found.
User Response: Correct the command by adding a right parenthesis. If you have used a name that contains special characters, ensure that the name is enclosed in single quotation marks. Then retry the command.

AMQ3508   *string* **- string expected.**
Explanation: A quoted string value was expected, but *string* found.
User Response: Correct the string value, then retry the command.

AMQ3509   *keyword* **not valid for this command.**
Explanation: The keyword or attribute is not valid for this command.
User Response: Correct the keyword or attribute, then retry the command.

AMQ3510   **ACTION keyword required.**
Explanation: A RESOLVE CHANNEL command was issued without the ACTION keyword.
User Response: Specify the ACTION keyword and an appropriate parameter, then retry the command.

AMQ3511   **CHLTYPE must be specified immediately after the channel name.**
Explanation: On the DEFINE CHANNEL and ALTER CHANNEL commands, you must specify the channel type immediately after the channel name. The validity of many of the following parameters are determined by the channel type.
User Response: Specify the CHLTYPE immediately after the channel name, then retry the command.

AMQ3512   **CONNAME keyword required.**
Explanation: The channel being defined has a channel type of SDR or RQSTR. With these channel types, you

3

must specify a connection name on the DEFINE CHANNEL command.
User Response: Specify the CONNAME keyword, then retry the command.

___

AMQ3513   Channel already exists - specify REPLACE.
Explanation: The channel being defined already exists and the replace option has not been specified.
User Response: If you want to replace the existing channel, specify the REPLACE keyword; otherwise choose a different channel name.

___

AMQ3514   Channel not found.
Explanation: The channel specified in the ALTER CHANNEL command does not exist.
User Response: Correct the channel name, then retry the command.

___

AMQ3515   LIKE channel not found.
Explanation: The LIKE channel name specified on the DEFINE CHANNEL command does not exist.
User Response: Correct the LIKE channel name, then retry the command.

___

AMQ3516   Primary keyword *string* not valid.
Explanation: A primary keyword was expected, but *string* was found.
User Response: Correct the primary keyword, then retry the command.

___

AMQ3517   Error opening channel definition file.
Explanation: The operating system could not open the channel definition file.
User Response: Try the command again. If the error persists, reinstall MQSeries, then try the command again. If this does not solve the problem, contact your MQSeries administrator.

___

AMQ3518   Error reading channel definition file.
Explanation: The operating system could not read the channel definition file.
User Response: Try the command again. If the error persists, reinstall MQSeries, then try the command again. If this does not solve the problem, contact your MQSeries administrator.

___

AMQ3519   Secondary keyword *string* not valid.
Explanation: A secondary keyword was expected, but *string* was found.
User Response: Correct the secondary keyword, then retry the command.

___

AMQ3520   TRPTYPE keyword required.
Explanation: When you define a channel of this type, you must specify the transport type.
User Response: Specify the TRPTYPE keyword on the DEFINE CHANNEL command, then retry the command.

___

AMQ3521   Cannot read continuation line.
Explanation: The operating system cannot read the line following the plus (+) character.
User Response: A plus character means that the command continues on the following line. Ensure the continuation character is required, correct the continuation line, then retry the command. If the error persists, reinstall MQSeries, then try the command again. If this does not solve the problem, contact your MQSeries administrator.

___

AMQ3522   Unexpected comma.
Explanation: A keyword was expected, but a comma (,) was found.
User Response: Correct the keyword, then retry the command.

___

AMQ3523   Unexpected left parenthesis.
Explanation: A keyword was expected, but a left parenthesis was found.
User Response: Correct the keyword, then retry the command.

___

AMQ3524   Unexpected number *number*.
Explanation: A keyword was expected, but *number* was found.
User Response: Correct the keyword, then retry the command.

___

AMQ3525   Unexpected right parenthesis.
Explanation: A keyword was expected, but a right parenthesis was found.
User Response: Correct the keyword, then retry the command.

___

AMQ3526   Value type *type* not supported.
Explanation: There may be a fault in the MQSC command processor.
User Response: Restart the MQ connection. If the error persists, reinstall MQSeries, then restart the MQ connection. If this does not solve the problem, contact your MQSeries administrator.

___

AMQ3527   Keyword *keyword* is not valid for given channel type.
Explanation: The keyword *keyword* is not valid for use with the channel type specified by the CHLTYPE keyword.
User Response: Correct the keyword, then retry the command.

AMQ3528   Wrong CHLTYPE for LIKE channel.
Explanation: The channel type for the channel defined by the LIKE keyword is different from that specified by the CHLTYPE keyword. In the DEFINE CHANNEL command, the channel types must match.
User Response: Either correct the LIKE keyword to specify a channel of the required type, or correct the CHLTYPE keyword to match the channel type of the LIKE channel. Then retry the command.

AMQ3529   Wrong CHLTYPE for given definition.
Explanation: The channel type of the named channel is different from the CHLTYPE specified. In the ALTER CHANNEL command, the channel types must match.
User Response: Correct the CHLTYPE keyword to match the channel type of the named channel, then retry the command.

AMQ3530   XMITQ keyword required.
Explanation: The channel being defined has a channel type of SDR or SVR. With these channel types, you must specify a transmission queue name on the DEFINE CHANNEL command.
User Response: Specify the XMITQ keyword, then retry the command.

AMQ3531   Character attribute buffer exceeded.
Explanation: A character attribute is greater than 2000 characters in length.
User Response: Correct the attribute, then retry the command. If the error persists, reinstall MQSeries, then try the command again. If this does not solve the problem, contact your MQSeries administrator.

AMQ3532   File ended unexpectedly.
Explanation: The file ended unexpectedly while reading a continuation line.
User Response: Ensure that the command correctly uses the continuation character, then retry the command.

AMQ3533   Character *character* not valid.
Explanation: While parsing the keywords and attributes, *character* was found. It is not valid in this context.
User Response: Correct the character. If you have used a name that contains special characters, ensure that the name is enclosed in single quotation marks. Then retry the command.

AMQ3534   Keyword *keyword* not valid.
Explanation: You cannot use this keyword in this situation.
User Response: Correct the syntax of the command, then retry it.

AMQ3535   LIKE channel name too long.
Explanation: The length of the channel name specified with the LIKE keyword is too long. Channel names can be a maximum of 20 characters.
User Response: Correct the channel name, then retry the command.

AMQ3536   Terminator of number *number* not valid.
Explanation: The characters following the number *number* are not valid.
User Response: Correct the number, then retry the command.

AMQ3537   Keyword beginning *keyword* too long.
Explanation: The length of the keyword *keyword* is more than 10 characters.
User Response: Correct the keyword, then retry the command.

AMQ3538   Name type is not a string.
Explanation: There may be a fault in the MQSC command processor.
User Response: Restart the MQ connection. If the error persists, reinstall MQSeries, then restart the MQ connection. If this does not solve the problem, contact your MQSeries administrator.

AMQ3539   Number beginning *number* is too long.
Explanation: The number beginning with *number* is longer than the maximum of 9 digits.
User Response: Correct the number, then retry the command.

AMQ3540   Keyword DEADQ *(string)* must be blank.
Explanation: MQSeries does not support dead-letter queues. You can set this keyword to blanks only.
User Response: Remove the DEADQ keyword, then retry the command.

AMQ3541   String *string* ended unexpectedly.
Explanation: The string *string* ended with a new-line character instead of a quotation mark.
User Response: Correct the string, then retry the command.

AMQ3542   String beginning *string* is too long.
Explanation: The string *string* is too long for its associated keyword.
User Response: Correct the string, then retry the command.

AMQ3543   Too many attributes, ignoring this one.
Explanation: You can specify a maximum of 256
attributes on one command; any more are ignored.
User Response: Correct the number of attributes
associated with the command, then retry it. If the error
persists, reinstall MQSeries, then try the command
again. If this does not solve the problem, contact your
MQSeries administrator.

AMQ3544   Keyword MCANAME *(name)* **must be
blank.**
Explanation: MQSeries does not support the
MCANAME keyword. You can set it to blanks only.
User Response: Remove the MCANAME keyword,
then retry the command.

AMQ3545   Internal error number = *return code*
Explanation: There may be a fault in the MQSC
command processor.
User Response: Restart the MQ connection. If the
error persists, reinstall MQSeries, then restart the MQ
connection. If this does not solve the problem, contact
your MQSeries administrator.

AMQ3546   Secondary keyword not specified.
Explanation: The command must have a secondary
keyword, but none has been specified.
User Response: Specify the secondary keyword, then
retry the command.

AMQ3547   String length of string *string* **is not
valid.**
Explanation: Each string associated with a keyword
can be of a certain length only. The length of string
string is not valid for its associated keyword.
User Response: Correct the string, then retry the
command.

AMQ3548   Line too long - MQSC file error
Explanation: The MQSC file did not end with the
expected new-line character.
User Response: Edit the file and ensure that the final
command ends correctly. If necessary, add a blank line
to the end of the file. Then run the MQSC command
file again.

AMQ3549   MQSC file *filename* **ran successfully.**
Explanation: The MQSC command file ran
successfully.
User Response: None required.

AMQ3550   MQSC file *filename* **did not run
successfully. See MQSC.LOG for more
details.**
Explanation: Some of the commands in the MQSC
command file *filename* contained errors.

User Response: Correct the commands in the file, then
run the file again.

**AMQ3551   *string* contains characters not valid for
MQSeries objects.**
Explanation: The name *string* contains characters that
are not valid for MQSeries objects. When you name
MQSeries objects, you can use only the following
characters:
*   Uppercase A-Z
*   Lowercase a-z
*   Integers 0-9
*   Period (.)
*   Forward slash (/)
*   Underscore (_)
*   Percent sign(%)
User Response: Replace the nonvalid characters in the
string, then retry the command.

**AMQ3552   Internal error in the DISPLAY
command. Return code =** *code***.**
Explanation: The MQSC command is syntactically
correct, but the queue manager could not perform the
command.
User Response: Look up the return code in the
MQSeries for Windows User's Guide and try to correct the
problem. Then stop and restart the MQ connection. If
the error persists, contact your MQSeries administrator.

**AMQ3553   Insufficient memory for the DISPLAY
command.**
Explanation: MQ cannot allocate sufficient memory to
display all the information generated by the DISPLAY
command.
User Response: If you used an asterisk (*) to specify a
generic name, retry the command using a more
restrictive name so MQ has to display fewer objects. If
you cannot do this, free some memory by stopping
some applications you are no longer using, then retry
the command.

**AMQ3554   The TYPE value specified is not
supported for this DISPLAY command.**
Explanation: Either:
*   You have specified a queue type on the DISPLAY
    CHANNEL command.
*   You have specified a channel type on the DISPLAY
    QUEUE command.
User Response: Retry the command using the correct
syntax. For explanations of the syntax of the MQSC
commands, see the *MQSeries MQSC Command Reference*.

AMQ3555   MQSeries object *name* **not found.**
Explanation: There are no objects matching the name
you specified in the command.
User Response: If you expected MQ to return
information, retry the command, ensuring you type the
correct object name.

AMQ3601 MQSeries for Windows has failed to verify correctly. Reason: *reason*.
Explanation: The *reason* describes the task that the Verify program was attempting when it found an error.
User Response: For information on the possible causes of this message, see the *MQSeries for Windows User's Guide*. If you cannot rectify the problem, uninstall the product, reinstall it, then retry the Verify program. If the error persists, contact your MQ administrator.

AMQ3700 Command server started successfully for queue manager *name*.
Explanation: The operation was successful.
User Response: None required.

AMQ3701 Command server stopped successfully.
Explanation: The operation was successful.
User Response: None required.

AMQ3702 Cannot stop command server. Return code = *code*.
Explanation: There is an internal error in the command server.
User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. Then restart the command server. If the error persists, stop the connection, then restart it and the command server. If this does not solve the problem, contact your MQSeries administrator.

AMQ3703 Cannot create the command queue. Return code = *code*.
Explanation: The command queue (SYSTEM.ADMIN.COMMAND.QUEUE) does not exist and MQ cannot create it.
User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. If the error persists, contact your MQSeries administrator.

AMQ3704 Cannot open the command queue. Return code = *code*.
Explanation: MQ cannot open the command queue (SYSTEM.ADMIN.COMMAND.QUEUE).
User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. If the error persists, contact your MQSeries administrator.

AMQ3705 Cannot allocate memory to service the command queue.
Explanation: The command server must allocate sufficient memory to service the longest message on the command queue. This memory is not available.
User Response: Free some memory by stopping any applications you no longer need to run, then restart the command server.

AMQ3706 Cannot query the attributes of the command queue. Return code = *code*.
Explanation: The command server uses the MQINQ call to find the length of the longest message on the command queue. This call has failed.
User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. If the error persists, contact your MQSeries administrator.

AMQ3707 Internal error in the command server. Return code = *code*.
Explanation: There is an internal error in the command server.
User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. Then restart the command server. If the error persists, stop the connection, then restart it and the command server. If this does not solve the problem, contact your MQSeries administrator.

AMQ3708 The reply-to queue does not exist.
Explanation: The reply-to queue specified in the PCF command does not exist. The command server cannot process the command.
User Response: Before you retry the command, ensure that all the queues required by the PCF application exist on the queue manager.

AMQ3709 Cannot open or put a message on the reply-to queue. Return code = *code*.
Explanation: MQ cannot open the reply-to queue or it cannot put a message on the queue.
User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. If the error persists, contact your MQSeries administrator.

AMQ3710 Cannot close the reply-to queue. Return code = *code*.
Explanation: MQ cannot close the reply-to queue. queue.
User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. If the error persists, contact your MQSeries administrator.

AMQ3711 Internal error in the command server. Return code = *code*.
Explanation: There is an internal error in the command server.
User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. Then restart the command server. If the error persists, stop the connection, then restart it and the

command server. If this does not solve the problem, contact your MQSeries administrator.

AMQ3712   Cannot get a message from the administration command queue. Return code = *code*.

Explanation: MQ cannot get a message from the administration command queue (SYSTEM.ADMIN.COMMAND.QUEUE). The command server has stopped.

User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. Then restart the command server. If the error persists, contact your MQSeries administrator.

AMQ3713   Internal error in the command server. Return code = *code*.

Explanation: There is an internal error in the command server.

User Response: Look up the reason code in the *MQSeries for Windows User's Guide* and try to correct the problem. Then restart the command server. If the error persists, stop the connection, then restart it and the command server. If this does not solve the problem, contact your MQSeries administrator.

AMQ3714   Cannot get messages from the administration command queue because it is inhibited.

Explanation: The InhibitGet attribute of the administration command queue (SYSTEM.ADMIN.COMMAND.QUEUE) is set so that MQGET calls are not allowed. This means the command server cannot get messages from the queue.

User Response: If you do not want to prevent MQGET calls, change the InhibitGet attribute of the queue to allow MQGET calls. To do this on the Complete version of the product, use the Components page of the MQSeries Properties dialog box. To do this on the Compact version of the product, use an MQSC command.

AMQ3715   The administration command queue is no longer inhibited.

Explanation: The InhibitGet attribute of the administration command queue (SYSTEM.ADMIN.COMMAND.QUEUE) has been changed so that MQGET calls are now allowed. This means the command server can get messages from the queue.

User Response: None required.

AMQ3801   Connection broken; there is a problem with the queue manager.

Explanation: The connection is stopped because there is a problem with its queue manager.

User Response: For more information, use the MQSeries Properties dialog to view the status of the queue manager. Then restart the connection.

AMQ3802   Connection broken; there is a problem with the channel group.

Explanation: The connection is stopped because there is a problem with its channel group.

User Response: For more information, use the MQSeries Properties dialog to view the status of the channel group. Then restart the connection.

AMQ3803   Connection broken; there is a problem with the phonebook entry.

Explanation: The connection is stopped because there is a problem with its phonebook entry.

User Response: For more information, use the MQSeries Properties dialog to view the status of the phonebook entry. Then restart the connection.

AMQ3804   Channel group broken; there is a problem with the listener.

Explanation: The connection is stopped because there is a problem with the listener in its channel group.

User Response: For more information, use the MQSeries Properties dialog to view the status of the listener. Then restart the connection.

AMQ3805   Channel group broken; there is a problem with one or more channels.

Explanation: The connection is stopped because there is a problem with one or more of the channels in its channel group.

User Response: For more information, use the MQSeries Properties dialog to view the status of the channels. Then restart the connection.

Chapter 3. AMQ4000-AMQ4999 MQSeries for Windows NT User Interface messages

> **Reading variables in the message**
>
> Some messages display text or numbers that vary according to the circumstances giving rise to the message; these are known as *message variables*. The message variables are indicated in this book by the use of the '&' symbol and a number: &1, &2, and so on.
>
> In some cases a message may have variables in the Explanation or User action. Find the values of the message variables by looking in the error log. The complete message, including the Explanation and the User action, is recorded there.

AMQ4000 **Default object not found. Cannot create a new object because the default object for the object type could not be found.**

Explanation: When you create a new object, the initial values of the object's attributes are copied from an object called a default object. Each type of MQSeries object has a separate default object. The MQSeries Explorer could not obtain the attributes of the default object for the type of object that you were trying to create.

User Response: Use the MQSeries administration tools on the queue manager experiencing the problem to ensure that all of the default objects are intact.

AMQ4001 **The queue manager specified is already shown in the console.**

AMQ4002 **Are you sure you want to delete this object?**

AMQ4003 **MQSeries system objects are used internally by MQSeries. You are advised not to delete them. Do you want to keep the MQSeries system object?**

AMQ4004 **Clear all messages from the queue?**

AMQ4005 **The object has been replaced or deleted. The properties could not be applied.**

AMQ4006 **The channel was pinged successfully. MQSeries successfully sent data to the remote queue manager and then received the data returned.**

AMQ4007 **The message sequence number for the channel was reset.**

AMQ4008 **The request to start the channel was accepted.**

AMQ4009 **The request to stop the channel was accepted.**

AMQ4010 **The in-doubt state was resolved.**

AMQ4011 **The queue has been cleared of messages.**

AMQ4012 **The object was created successfully but it is not visible in the current view with the settings for visible objects.**

Explanation: None

User Response: None

AMQ4013 **You cannot run the MQSeries Explorer snap-in because you do not have authority to administer MQSeries.**

Explanation: You must be in the Administrators group, in the mqm group, or logged on with the SYSTEM ID to administer MQSeries.

User Response: None

AMQ4014 **The character '&1' was not valid.**

AMQ4015 **Supply a non-blank name.**

AMQ4016 **The test message was put successfully.**

Explanation: None

User Response: None

AMQ4017	**The IBM MQSeries Explorer snap-in requires Microsoft Management Console Version 1.1 or later. Install a suitable version of the MMC and retry the operation.**

AMQ4018	**The IBM MQSeries Explorer snap-in failed to initialize properly.**

Explanation: The MQSeries Explorer experienced a problem during initialization when it was loaded by the Microsoft® Management Console. This normally indicates that a piece of the underlying software could not be initialized.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved.

AMQ4019	**An object called &1 of the same type already exists. Do you want to replace the definition of the existing object?**

AMQ4020	**The changes you are making to the attributes of object &1 will affect the operation of the queue manager or another program currently using the object. Would you like to force the change to the object's attributes?**

AMQ4021	**Failed to access one or more MQSeries objects. The objects' icons have been marked to indicate the objects in error.**

Explanation: An operation which accessed multiple MQSeries objects was only partially successful. Some of the objects could not be accessed but the errors on the individual objects were not reported separately. An example of such an error is insufficient authority to obtain the attributes of an object.

User Response: To establish the cause of the problem on a particular object, select the object and refresh it.

AMQ4022	**You cannot specify the same name for the initiation queue as the name of the queue itself. Specify another queue name.**

AMQ4023	**The queue manager '&1' does not exist on this computer.**

AMQ4024	**The object cannot be replaced. Delete the object and retry the operation.**

AMQ4025	**The changes made to the cluster attributes of the queue will take effect once they have propagated across the network. You must refresh any views containing the cluster queues in the affected clusters to show the changes.**

AMQ4026	**You have created a queue which is shared in one or more clusters. The queue will be available as a cluster queue once its definition has propagated across the network. You must refresh any views containing the cluster queues in the affected clusters to show the cluster queue.**

AMQ4027	**An error occurred connecting to the queue manager. Are you sure you want to show this queue manager in the console?**

AMQ4028	**Platform not supported.**

Explanation: This queue manager cannot be administered by the MQSeries Explorer because it is running on an unsupported platform. The value &1 for the Platform attribute of the queue manager is not supported by the MQSeries Explorer.

User Response: None

AMQ4029	**Command level too low.**

Explanation: This queue manager cannot be administered by the MQSeries Explorer because the command level of the queue manager is less than &1. You will have to upgrade the queue manager to a newer version of MQSeries.

User Response: None

AMQ4030	**Codepage conversion table not found. This queue manager cannot be administered by the MQSeries Explorer because a codepage conversion table was not found. Install a codepage conversion table from CCSID &1 to CCSID &2 on the computer on which the MQSeries Explorer is running.**

Explanation: None
User Response: None

AMQ4031	**CCSID not found.**

Explanation: This queue manager cannot be administered by the MQSeries Explorer because CCSID &1 cannot be found in the CCSID table. The MQSeries Explorer cannot convert character data to or from the unrecognized CCSID.

User Response: None

AMQ4032   Command server not available.
Explanation: If the command server is not running, start the command server and retry. Ensure that the queue called SYSTEM.ADMIN.COMMAND.QUEUE is configured to enable programs to get messages off it.
User Response: None

AMQ4033   Gets inhibited for the queue. Cannot get messages from the queue.

AMQ4034   Message too long.
Explanation: You tried to put a message on a queue that was larger than the maximum allowed for the queue or queue manager.
User Response: None

AMQ4035   No message available. The response message did not arrive within a reasonable amount of time.
Explanation: The MQSeries Explorer was expecting a response message from the command server. The message did not arrive within 30 seconds. This may indicate a transient problem that caused an unusual delay in receiving a response. Alternatively, it may indicate a problem with the command server of the queue manager being administered.
User Response: Retry the operation. If the error persists, examine the problem determination information on the queue manager's computer to see if the command server recorded any problem determination information.

AMQ4036   Access not authorized. You are not authorized to perform this operation.

AMQ4037   Object definition changed since opened.

AMQ4038   Object damaged. The object was damaged and could not be accessed. The object may be deleted. Alternatively, it may be possible to recover it from a media image or backup.
Explanation: None
User Response: None

AMQ4039   Object in use.
Explanation: The object was already open from another application. To change the attributes of an object, specify the Force option as 'Yes' to apply the changes. If you do this, any applications using the object must close and reopen the object to proceed.
User Response: None

AMQ4040   Puts inhibited for this queue. Cannot put messages to this queue.

AMQ4041   Queue is deleted.

AMQ4042   Queue full. The queue contains the maximum number of messages.

AMQ4043   Queue manager not available for connection.
Explanation: Ensure that the queue manager is running. If the queue manager is running on another computer, ensure it is configured to accept remote connections.
User Response: None

AMQ4044   Queue manager stopping.

AMQ4045   Queue not empty. The queue contains one or more messages or uncommitted put or get requests.

AMQ4046   Insufficient system resources available.
Explanation: The operation failed due to a resource shortage. Possible causes include shortage of disk space or virtual memory.
User Response: Examine the problem determination information on the queue manager's computer to establish the type of resource which has been exhausted. Rectify the problem and retry the operation.

AMQ4047   Insufficient storage available.

AMQ4048   Unexpected error.
Explanation: An unexpected error occurred during the operation.
User Response: Examine the problem determination information on this computer and also on the queue manager's computer to establish the cause of the error.

AMQ4049   Unknown object name.

AMQ4050   Allocation failed. An attempt to allocate a conversation to a remote system failed. The error may be due to an invalid entry in the channel definition or it may be that the listening program at the remote system was not running.

AMQ4051   Bind failed. The bind to a remote system during session negotiation failed.
Explanation: An attempt to bind the TCP/IP socket to the listener port was unsuccessful.
User Response: The failure could be due to another

program using the same port number. Examine the problem determination information on this computer and also on the queue manager's computer to establish the cause of the error.

AMQ4052   Coded character-set ID error. Cannot convert a command message to the CCSID of the target queue manager.

AMQ4053   Channel in doubt. The operation could not complete because the channel was in doubt.
Explanation: The operation cannot be completed because the channel is in doubt. This is a normal but transitory state for an MQSeries channel.
User Response: Examine the status of the channel, and either restart the channel to resolve the in-doubt state, or resolve the channel manually from the sending end of the channel.

AMQ4054   Channel in use.
Explanation: An attempt was made to perform an operation on a channel, but the channel is currently active.
User Response: None

AMQ4055   Channel status not found.
Explanation: No channel status is available for this channel. This may indicate that the channel has not been used.
User Response: None

AMQ4056   Command failed.
Explanation: The MQSeries Explorer issued a command to the queue manager's command server. The command failed but the reason could not be diagnosed.
User Response: Examine the problem determination information on this computer and also on the queue manager's computer to establish the cause of the error.

AMQ4057   Configuration error.
Explanation: A configuration error was detected in the channel definition or communication subsystem; allocation of a conversation was not possible.
User Response: None

AMQ4058   Connection closed.
Explanation: An error occurred while receiving data from a remote system. The connection to the remote system has unexpectedly terminated.
User Response: None

AMQ4059   Connection refused.
Explanation: Could not establish a connection to a remote system. The remote system might not have been configured to allow a connection from this system.
User Response: None

AMQ4060   Dynamic queue scope error.
Explanation: The Scope attribute of the queue was set to MQSCO_CELL but this is not allowed for a dynamic queue.
User Response: None

AMQ4061   Remote system not available.
Explanation: Could not allocate a conversation to a remote system. The error may be transitory so the allocate may succeed later.
User Response: None

AMQ4062   The MQINQ call failed.
Explanation: The queue manager issued an MQINQ call to inquire about an MQSeries object. The call failed.
User Response: Check the queue manager's error log for more information about the error.

AMQ4063   The MQOPEN call failed.
Explanation: The queue manager issued an MQOPEN call to open an MQSeries object. The call failed.
User Response: Check the queue manager's error log for more information about the error.

AMQ4064   The MQSET call failed.
Explanation: The queue manager issued an MQSET call to set the values of the attributes of an MQSeries object. The call failed.
User Response: Check the queue manager's error log for more information about the error.

AMQ4065   Message sequence number error.
Explanation: The message sequence number parameter was not valid.
User Response: None

AMQ4066   Message truncated.
Explanation: The command server received a message that was larger than its maximum valid message size.
User Response: None

AMQ4067   Communications manager not available. The communications subsystem is not available.
Explanation: An attempt was made to use the communications subsystem but it has not been started.
User Response: Start the communications subsystem and retry the operation.

AMQ4068   Queue is not a transmission queue.
Explanation: The queue specified in the channel definition was not a transmission queue.
User Response: None

AMQ4069   Object already exists.
Explanation: Could not create object because the object already existed.
User Response: None

AMQ4070   Object is open.
Explanation: An attempt was made to delete or change an object that was in use.
User Response: Wait until the object is not in use and retry.

AMQ4071   Object has wrong type.
Explanation: Could not replace a queue object of a different type.
User Response: None

AMQ4072   Queue already exists in cell.
Explanation: Could not define a queue with cell scope or change the scope of an existing queue from queue-manager scope to cell scope because a queue with that name already existed in the cell.
User Response: None

AMQ4073   Ping error.
Explanation: You can only ping a sender or server channel.
User Response: If the local channel is a receiver channel, ping from the remote queue manager.

AMQ4074   Receive failed.
Explanation: The receive operation failed. This may be caused by a communications failure.
User Response: None

AMQ4075   Error while receiving data.
Explanation: An error occurred while receiving data from a remote system. This may be caused by a communications failure.
User Response: None

AMQ4076   Remote queue manager terminating.
Explanation: The channel stopped because the remote queue manager was terminating.
User Response: None

AMQ4077   Remote queue manager not available.
Explanation: The channel could not be started because the remote queue manager was not available. Ensure that the remote queue manager is started and that it is configured to accept incoming communication requests.
User Response: None

AMQ4078   Send failed.
Explanation: An error occurred while sending data to a remote system. This may be caused by a communications failure.
User Response: None

AMQ4079   Channel terminated by security exit.
Explanation: A channel security exit terminated the channel.
User Response: None

AMQ4080   Remote channel not known.
Explanation: There is no definition of this channel on the remote system.
User Response: None

AMQ4081   User exit not available.
Explanation: The channel was terminated because the user exit specified does not exist.
User Response: None

AMQ4082   Unexpected MQSeries error (&1).
Explanation: An unexpected error occurred during the operation. The MQSeries reason code reported was &1.
User Response: Read the description of the reason code in the MQSeries Application Programming Reference or Programmable System Administration manual. Examine the problem determination information on this computer and also on the queue manager's computer to establish the cause of the error.

AMQ4083   Queue manager name not known.
Explanation: If the queue manager is remote, this may indicate that another queue manager is using the same connection name. Queue managers using TCP/IP on the same computer must listen on different port numbers. This means that they will also have different connection names.
User Response: None

AMQ4084   Cell directory is not available.
Explanation: The Scope attribute of the queue was set to MQSCO_CELL but no name service supporting a cell directory has been configured.
User Response: None

AMQ4085   No name supplied for transmission queue.
Explanation: None
User Response: You must supply a non-blank transmission queue name for this channel type.

AMQ4086   No connection name supplied.
Explanation: None
User Response: Supply a non-blank connection name for this channel type.

AMQ4087   An error occurred while trying to use a cluster resource.
Explanation: None
User Response: Check that the queues whose names start with 'SYSTEM.CLUSTER.' are not full and that messages are allowed to be put on them.

AMQ4088   Cannot share transmission queue in cluster.
Explanation: The queue is a transmission queue and cannot be shared in a cluster.
User Response: None

AMQ4089   Puts inhibited for system command queue. Cannot put messages to the queue called 'SYSTEM.ADMIN.COMMAND.QUEUE'.

AMQ4090   The MQSeries Explorer uses the queue called 'SYSTEM.ADMIN.COMMAND.QUEUE'. Are you sure you want to continue?
Explanation: If you inhibit puts or gets for this queue, you will no longer be able to administer the queue manager using the MQSeries Explorer.
User Response: None

AMQ4091   Cannot connect to remote queue manager.
Explanation: The remote queue manager is accepting connections on an unsupported protocol. The MQSeries Explorer only supports connections to remote queue managers using the TCP/IP protocol.
User Response: None

AMQ4092   Leaving cluster requires change to a namelist.
Explanation: The queue manager's membership of the cluster is defined using a namelist '&2'.
User Response: To remove the queue manager from the cluster, remove '&1' from the namelist ensuring that you do not inadvertently affect other the definitions of other objects using the namelist.

AMQ4093   The cluster specified is already shown in the console.

AMQ4094   An error occurred adding this cluster to the console. Are you sure you want to show this cluster in the console?

AMQ4095   Queue manager '&1' is not a repository queue manager for cluster '&2'.
Explanation: The MQSeries Explorer needs a connection to a repository queue manager to administer the cluster.
User Response: None

AMQ4096   Are you sure you want to clear the password for this channel?

AMQ4097   Unmatched quotation mark. The attribute can contain a list of values.
Explanation: Each value may be enclosed in a pair of single or double quotation marks. A quotation mark may be included in a value by using a pair of quotation marks.
User Response: None

AMQ4098   Invalid list format. The attribute can contain a list of values. Each pair of values must be separated by a space or a comma. Values may be enclosed in single or double quotation marks. Quotation marks are only required for values which contain spaces, commas or quotation marks.

AMQ4099   Cannot communicate with one or more repository queue managers.
Explanation: Cluster '&1' is configured to use one or more repository queue managers which communicate using a protocol other than TCP/IP. The MQSeries Explorer can only establish connections to remote queue managers using TCP/IP.
User Response: To complete removal of the queue manager from the cluster, you must issue the RESET CLUSTER ACTION(FORCEREMOVE) command from the repository queue managers.

AMQ4100   The MMC document file <filename> could not be created. The IBM MQSeries Services console has failed to initialize.
Explanation: The service program has attempted to create a default MSC document file in the MQM "\bin" directory, but could not.
User Response: Check that the disk is not full, and that the user has create and write permissions in the MQM bin directory.

AMQ4101 **The IBM MQSeries Services could not be stopped. The error message is: <error message>.**
Explanation: The task bar program has attempted to stop the IBM MQSeries service. The error message supplied has been generated by the windows NT service control manager.
User Response: None

AMQ4102 **Shutting down IBM MQSeries will terminate all running queue managers and MQSeries processes. Are you sure you want to continue?**
Explanation: None
User Response: None

AMQ4106 **IBM MQSeries Services has been installed successfully.**

AMQ4107 **The Service Control handler could not be registered.**
Explanation: There is a problem installing or running the IBM MQSeries service.
User Response: None

AMQ4108 **A general error has occurred: <message>.**

AMQ4109 **A COM error has occurred:<message>.**

AMQ4110 **This queue manager is already stopped.**
Explanation: The user has tried to stop a queue manager that is already in the stopped state, or has disabled the service from the hardware profiles in the Windows NT control panel Services applet.
User Response: None, or if the service has been disabled from the hardware profile, enable the service.

AMQ4111 **This service is already stopped.**
Explanation: The user has tried to stop a service that is already in the stopped state.
User Response: None

AMQ4112 **This service is already running.**
Explanation: The user has tried to start a queue manager that is already in the started state.
User Response: None

AMQ4113 **This queue manager is already running.**
Explanation: The user has tried to start a service that is already in the started state.
User Response: None

AMQ4114 **Invalid parameter specified. Please re-enter this value.**
Explanation: The highlighted parameter is not within the valid range for this field.
User Response: Correct the field and retry.

AMQ4115 **Invalid parameter specified. Please enter a value between min and max.**
Explanation: The highlighted parameter is not within the valid range for that field.
User Response: Correct the field and retry.

AMQ4116 **The queue manager must be stopped before it can be deleted.**
Explanation: The user has tried to delete a queue manager that is still running.
User Response: Stop the queue manager and try again.

AMQ4117 **This action cannot be undone. Are you sure you want to delete the MQSeries queue manager 'queue manager' from your system?**
Explanation: None
User Response: Select "Yes" to confirm deletion of the queue-manager service, or "No" to continue.

AMQ4118 **Could not access the MQSeries Alert Monitor.**
Explanation: The snap-in has tried to access or launch the MQSeries alert monitor, but has failed. The COM error is appended to the message.
User Response: None

AMQ4119 **Could not initialize MQSeries Services utility library AMQMDUIN.**
Explanation: The snap-in has tried to access or launch the MQSeries Services utility library, but has failed. The COM error is appended to the message.
User Response: None.

AMQ4120 **Could not initialize Microsoft Foundation Classes for this thread.**
Explanation: The snap-in has tried to initialize the MFC library but has failed. Additional error text is appended to the message.
User Response: None.

AMQ4121 **Could not access MQSeries Services Channel Initiator.**
Explanation: The snap-in has tried to access or launch the MQSeries COM object, but has failed. The COM error is appended to the message.
User Response: None

AMQ4122   A general error has occurred. <message>

AMQ4123   Could not access MQSeries Services Listener.
Explanation: The snap-in has tried to access or launch the MQSeries COM object, but has failed. The COM error is appended to the message.
User Response: None

AMQ4124   Could not access MQSeries Services Trigger Monitor.
Explanation: The snap-in has tried to access or launch the MQSeries COM object, but has failed. The COM error is appended to the message.
User Response: None

AMQ4125   Could not access MQSeries Services COM object.
Explanation: The snap-in has tried to access or launch the MQSeries COM object, but has failed. The COM error is appended to the message.
User Response: None

AMQ4126   Could not release advice connections from MQSeries Services objects.
Explanation: The snap-in has tried to release its connections to the Service objects, but a COM error has occurred. Additional error text is appended to the message.
User Response: None

AMQ4127   Could not initialize MQSeries Services objects.
Explanation: The snap-in has tried to access or launch the MQSeries COM object, but has failed. The COM error is appended to the message.
User Response: None

AMQ4128   The service could not be started. <message>
Explanation: The IBM MQSeries service has tried to start a particular component service of a queue manager, but has failed. The returned error text from the specific service is appended to the message.
User Response: None

AMQ4200   There is a problem with the default configuration. Unable to display the Default Configuration window.
Explanation: There is a problem with MQSeries.
User Response: Use the 'Details>>' button to show further details about the problem and contact your systems administrator.

AMQ4201   Unable to check if the computer exists on this domain.
Explanation: MQSeries was unable to check if the computer name you entered exists on your computer's domain.
User Response: Retry the operation, if the problem persists contact your systems administrator.

AMQ4202   Unable to find the computer name 'name' on this TCP/IP domain 'domain'.
Explanation: MQSeries was unable to locate a computer with this name on your computer's TCP/IP domain.
User Response: Enter a different computer name.

AMQ4203   Unable to set up the default configuration.
Explanation: MQSeries was unable to set up the default configuration. This error may occur if MQSeries is busy with another operation.
User Response: Retry the operation. If the problem persists, use the 'Details>>' and 'Print' buttons to record further details about the problem and contact your systems administrator.

AMQ4204   Unable to join the default cluster.
Explanation: MQSeries was unable to join your computer to the default cluster. This error may occur if MQSeries is busy with another operation.
User Response: Retry the operation. If the problem persists, use the 'Details>>' and 'Print' buttons to record further details about the problem and contact your systems administrator.

AMQ4205   Unable to allow remote administration of the queue manager.
Explanation: MQSeries was unable change the configuration of your queue manager to allow it to be remotely administered. This error may occur if MQSeries is busy with another operation.
User Response: Retry the operation. If the problem persists, use the 'Details>>' and 'Print' buttons to record further details about the problem and contact your systems administrator.

AMQ4206   Unable to prevent remote administration of the queue manager.
Explanation: MQSeries was unable change the configuration of your queue manager to prevent it from being remotely administered. This error may occur if MQSeries is busy with another operation.
User Response: Retry the operation. If the problem persists, use the 'Details>>' and 'Print' buttons to record further details about the problem and contact your systems administrator.

AMQ4208   **Show this panel again the next time the queue manager is started?**
Explanation: You can choose whether you want the same panel to be shown the next time this queue manager is started, and the default configuration is not complete.
User Response: Select whether you want the panel to be shown next time.

AMQ4209   **The TCP/IP name of the remote computer must not be your own computer name.**
Explanation: You have selected that the repository queue manager is on another computer, but you have entered the name of your own computer.
User Response: Enter the correct name of the repository queue manager.

AMQ4210   **The command server must be active to complete this operation. Use the MQSeries Services to start it, then retry the operation.**
Explanation: The operation you requested needs the command server to be running.
User Response: Use MQSeries Services to start the command server, then retry the operation.

AMQ4212   **Unable to complete this task because you do not have authority to administer MQSeries. You must be in the Administrators group, in the mqm group or logged in with the SYSTEM ID to administer MQSeries.**
Explanation: Your userid is not authorized to carry out the operation you requested.
User Response: Retry the operation on a userid with the required authority, or contact your systems administrator.

AMQ4213   **Unable to delete queue manager &1 because it is being used by another program.**
Explanation: MQSeries was unable to delete the old default configuration queue manager because another program is using the queue manager.
User Response: Close the programs that are using the queue manager, and click Retry.

AMQ4214   **The computer 'name' is not known on the network.**
Explanation: MQSeries is unable to locate a computer with this name on your network.
User Response: Enter a different computer name.

AMQ4215   **Upgrade of the default configuration was cancelled.**
Explanation: You pressed 'Cancel' while running the default configuration wizard to upgrade the default configuration.
User Response: None

AMQ4251   **Cannot Initialize WinSock - TCP/IP may not be installed. Please install TCP/IP and try again.**
Explanation: Postcard was not able to initialize the interface to TCP/IP.
User Response: Check that TCP/IP has been installed successfully. If the problem persists, refer to your systems administrator.

AMQ4252   **Cannot Find WinSock - TCP/IP may not be installed. Please install TCP/IP and try again.**
Explanation: Postcard was not able to find the interface to TCP/IP.
User Response: Check that TCP/IP has been installed successfully. If the problem persists, refer to your systems administrator.

AMQ4253   **Cannot get fully qualified TCP/IP domain name - Please ensure that the TCP/IP protocol is configured.**
Explanation: Postcard was not able to determine the TCP/IP domain name for your computer.
User Response: Check that TCP/IP has been installed successfully. If the problem persists, refer to your systems administrator.

AMQ4254   **Failed to Allocate System Memory - Please contact your system administrator.**
Explanation: Postcard was not able to allocate enough memory to run correctly.
User Response: Close other programs to release system memory. If the problem persists, refer to your systems administrator.

AMQ4257   **The call MQCONN failed while preparing for a Put operation, with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**
Explanation: An error occurred when Postcard tried to connect to the queue manager in order to send the postcard. This error may occur if MQSeries is busy with another operation.
User Response: Try to send the postcard again. If the problem persists contact your systems administrator.

AMQ4258 **The call MQOPEN failed while preparing for a Put operation, with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to open a queue in order to send the postcard. This error may occur if MQSeries is busy with another operation.

User Response: Try to send the postcard again. If the problem persists contact your systems administrator.

AMQ4259 **The call MQCLOSE failed while preparing for a Put operation, with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to close the queue after sending the postcard. This error may occur if MQSeries is busy with another operation.

User Response: If the problem persists contact your systems administrator.

AMQ4260 **The call MQDISC failed while preparing for a Put operation, with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to disconnect from the queue manager after sending the postcard. This error may occur if MQSeries is busy with another operation.

User Response: If the problem persists contact your systems administrator.

AMQ4261 **The call MQPUT failed with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to send the postcard by putting its data to the queue. This error may occur if MQSeries is busy with another operation.

User Response: Try to send the postcard again. If the problem persists contact your systems administrator.

AMQ4262 **The call MQCONN failed while preparing for a Get operation, with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to connect to the queue manager in order to receive postcards. This error may occur if MQSeries is busy with another operation.

User Response: Restart Postcard. If the problem persists contact your systems administrator.

AMQ4263 **The call MQOPEN failed while preparing for a Get operation, with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to open a queue in order to send the postcard. This error may occur if MQSeries is busy with another operation.

User Response: Restart Postcard. If the problem persists contact your systems administrator.

AMQ4264 **The call MQCLOSE failed while preparing for a Get operation, with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to close the queue after receiving postcards. This error may occur if MQSeries is busy with another operation.

User Response: If the problem persists contact your systems administrator.

AMQ4265 **The call MQDISC failed while preparing for a Get operation, with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to disconnect from the queue manager after receiving postcards. This error may occur if MQSeries is busy with another operation.

User Response: If the problem persists contact your systems administrator.

AMQ4267 **The call MQGET failed with Completion Code 'completion-code (number)', Reason Code 'reason-code (number)'.**

Explanation: An error occurred when Postcard tried to receive a postcards by getting its data from the queue. This error may occur if MQSeries is busy with another operation.

User Response: Restart Postcard. If the problem persists contact your systems administrator.

AMQ4269 **Unable to run Postcard because you do not have authority to use MQSeries.**

Explanation: Your user Id is not authorized to use Postcard. You must be in the Administrator's group, in the mqm group, or logged in with the SYSTEM ID to use MQSeries.

User Response: Use Postcard on a user Id with the required authority, or contact your systems administrator.

AMQ4271   **Unable to open a local queue called '&1' on the mailbox queue manager '&2'.**

Explanation: Postcard was unable to automatically create the queue it uses on the queue manager.

User Response: Use MQSeries Explorer to create the queue, and restart Postcard.

AMQ4272   **The mailbox queue manager '&1' does not exist on this computer.**

Explanation: The mailbox queue manager name specified after the '-m' parameter to Postcard does not exist on this computer.

User Response: Restart Postcard specifying the name of a queue manager that does exist on this computer.

AMQ4273   **Unable to contact the target mailbox 'name'.**

Explanation: Postcard was unable send the message as it could not contact the target mailbox.

User Response: Click 'Retry' to attempt to send the message again, otherwise click 'Cancel'.

AMQ4274   **Postcard has detected that 'mailbox' is the name of a computer and a queue manager.**

Explanation: Postcard has detected that the destination mailbox name is the name of a computer and of a queue manager.

User Response: Select whether you want to send the message to the computer or the queue manager with this name, then click OK.

AMQ4300   ″**Please supply some text in order for the MQPUT(1) operation to succeed. (AMQ4300)**″

Explanation: No text has been supplied for the user so that the MQPUT or MQPUT1 operation can proceed.

User Response: Supply some text in the editable area so that the MQPUT or MQPUT1 operation can proceed.

AMQ4301   ″**Please supply some text in order for the MQPUT operation to succeed. (AMQ4301)**″

Explanation: No text has been supplied for the user so that the MQPUT operation may proceed.

User Response: Supply some text in the editable area so that the MQPUT may proceed.

AMQ4302   ″**Please supply some text in order for the MQPUT1 operation to succeed. (AMQ4302)**″

Explanation: No text has been supplied for the user so that the MQPUT1 operation may proceed.

User Response: Supply some text in the editable area so that the MQPUT1 may proceed.

AMQ4303   ″**The command server for the queue manager &s is not started. Start the command server and try again. (AMQ4303)**″

Explanation: In order for the API Exerciser to function, a command server must be running.

User Response: Either start the command server from the MQServices application or run strmqcsv <Queue Manager> from the command line.

AMQ4304   ″**Problem enumerating queues with reason &s, code (&ld). (AMQ4304)**″

Explanation: The API Exerciser encountered a problem trying to enumerate queues.

User Response: Ensure that the command server is running (from the Service application) and that there are queues configured for the queue manager.

AMQ4305   ″**There are no queue managers present in the system. Please create one and try again. (AMQ4305)**″

Explanation: The API Exerciser could not find any queue managers on the system.

User Response: Use the Services application to create one or run crtmqm <Queue Manager>.

AMQ4306   ″**Memory allocation failure. Stop some other applications and try again. (AMQ4306)**″

Explanation: There are not sufficient system resources available in the system to satisfy the running of API Exerciser.

User Response: Shut some other applications down and try running the API Exerciser again.

AMQ4307   **API Exerciser encountered a COM failure and cannot continue.**

Explanation: When the API Exerciser started, it was unable to make a COM connection to MQSeries Services.

User Response: Ensure that MQSeries has been correctly installed and configured, and that your user ID is a member of the mqm group. If the problem persists, refer to your systems administrator.

AMQ4717   **Cannot start dialog '&s'.**

Explanation: An internal error occurred while trying to display a dialog.

User Response: Review the installation log for details of any problems. If the error persists, contact your systems administrator.

AMQ4718   **Cannot load performance counters**

Explanation: An error occurred when loading the MQSeries performance counter information.

User Response: Review the installation log for details

of any problems. If the error persists, contact your systems administrator.

AMQ4719   Error migrating queue manager command files.
Explanation: An error occurred while migrating queue manager command files to the registry.
User Response: Review the installation log for details of any problems. If the error persists, contact your systems administrator.

AMQ4720   Error initializing security environment.
Explanation: Setup was unable to successfully initialize the security environment.
User Response: Review the installation log for details of any problems. If the error persists, contact your systems administrator.

AMQ4722   Setup cannot continue without VGA or better screen resolution.
Explanation: Setup was run using a monitor resolution less than VGA resolution.
User Response: Use a monitor with resolution equal to or better than 640 x 480 pixels.

AMQ4724   Error restoring queue manager configuration.
Explanation: Setup was unable to restore the queue manager configuration information to the registry.
User Response: Review the installation log for details of any problems. If the error persists, contact your systems administrator.

AMQ4725   The Server component cannot be installed without 800 x 600 pixels or better screen resolution.
Explanation: The server component cannot be installed if the monitor's resolution is less than 800 x 600 pixels.
User Response: Use a monitor with resolution equal to or better than 800 x 600 pixels.

AMQ4726   The 'Internet Gateway' component requires the 'Windows NT Client' component.
Explanation: You cannot install the Internet Gateway component without installing the Windows NT Client component.
User Response: Select both components, the Windows NT Client component only, or neither component.

AMQ4727   Uninstallation of the 'Windows NT Client' component requires uninstallation of the 'Internet Gateway' component.
Explanation: You cannot uninstall the Windows NT Client component without uninstalling the Internet Gateway component.
User Response: Select both components, the Internet Gateway component only, or neither component.

AMQ4729   You cannot install the Windows NT Client from the client CD-ROM because MQSeries server components are already installed on this computer. To install the Windows NT Client on this computer, use the server CD-ROM.
Explanation: If one or more components from the server CD-ROM are installed, you must use the server CD-ROM to install further components.
User Response: Use the server CD-ROM.

AMQ4730   Java support is now separately available; see the installation notes. Setup will delete existing MQSeries V5.1 Java files.
Explanation: MQSeries V5.2 installation removes the Java™ support files that were included in MQSeries V5.1.
User Response: If you require MQSeries Java support, obtain the code from IBM and install it separately.

AMQ4350   Setup cannot continue; a later version of MQSeries for Windows NT is installed.
Explanation: Installation detected that a version of MQSeries for Windows later than version 5.1 is already installed on the computer.
User Response: Do not attempt to install version 5.1 when a later version is already installed.

AMQ4351   Uninstallation cannot continue; uninstallation is already running.
Explanation: An attempt was made to run two copies of uninstallation at once.
User Response: Run only one copy of uninstallation at a time.

AMQ4352   Setup cannot continue; Windows NT Version 4 is required.
Explanation: A version of Windows NT earlier than version 4 is installed.
User Response: Do not attempt to install MQSeries for Windows NT on a version of Windows NT earlier than version 4.

AMQ4353 **Setup cannot continue; '&s' is not an Administrator.**
Explanation: The user running installation does not have administrator authority.
User Response: Log off and log back on using a user ID with administrator authority.

AMQ4354 **No repository computer name entered.**
Explanation: The required computer name of the repository was not entered in the dialog.
User Response: Enter the repository computer name.

AMQ4355 **Repository computer name is not valid.**
Explanation: The entered computer name is not a valid computer name.
User Response: Enter a valid computer name.

AMQ4356 **Enter a remote computer name.**
Explanation: The computer name entered is the name of the computer that is running installation.
User Response: Enter the name of a remote computer.

AMQ4357 **Registration failed for file '&s' (code &d).**
Explanation: An error occurred during the self-registering process for an MQSeries executable file.
User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4358 **Unregistration failed for file '&s' (code &d).**
Explanation: An error occurred during the self-unregistering process for an MQSeries executable file.
User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4359 **Unable to register file '&s'.**
Explanation: Installation was unable to invoke the system utility regsvr32.exe, or, if the file is a .exe file, was unable to invoke the .exe file.
User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4360 **Unable to unregister file '&s'.**
Explanation: Installation was unable to invoke the system utility regsvr32.exe, or, if the file is a .exe file, was unable to invoke the .exe file.
User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4361 **Uninstall cannot continue; Administrator logon required.**
Explanation: The user running uninstallation does not have administrator authority.
User Response: Log off and log back on using a user ID with administrator authority.

AMQ4362 **Failed to create the default configuration.**
Explanation: An error occurred creating the default configuration.
User Response: Review the installation log file for details of any problems.

AMQ4363 **Setup could not detect the Windows NT Service Pack level (Service Pack 3 or later is required). Is Service Pack 3 or later installed?**
Explanation: Setup was unable to detect which Service Pack is installed on the computer.
User Response: Reply Yes if you know that the indicated Service Pack or a later version is installed, otherwise reply No. Install the Service and then retry.

AMQ4364 **Setup could not detect the Windows NT Service Pack level (Service Pack 4 or later is required). Is Service Pack 4 or later installed?**
Explanation: Setup was unable to detect which Service Pack is installed on the computer.
User Response: Reply Yes if you know that the indicated Service Pack or a later version is installed, otherwise reply No. Install the Service and then retry.

AMQ4365 **Setup cannot continue because Service Pack 3 is not installed.**
Explanation: The required Service Pack level is not installed.
User Response: Install the Service Pack and retry.

AMQ4366 **Setup cannot continue because Service Pack 4 is not installed.**
Explanation: The required Service Pack level is not installed.
User Response: Install the Service Pack and retry.

AMQ4367 **Setup cannot continue because Internet Explorer Version 4 is not installed.**
Explanation: MQSeries requires some system files that are part of Internet Explorer 4.01.
User Response: Install Internet Explorer 4.01 and then retry. You do not have to use Internet Explorer as an Internet browser; just the files are required.

AMQ4368   Select at least one component to continue.

Explanation: No MQSeries components were selected for installation.

User Response: Select one or more components.

AMQ4369   The 'Web Administration Server' component requires the 'Server' component.

Explanation: You cannot install the Web Administration Server component without installing the Server component.

User Response: Select both components, just the Server component or neither of these.

AMQ4370   Uninstallation of the 'Server' component requires uninstallation of the 'Web Administration Server' component.

Explanation: You cannot uninstall the Server component without uninstalling the Web Administration Server component.

User Response: Select both components, just the Web Administration Server component or neither of these.

AMQ4371   The 'Documentation in Other Languages' component requires the 'Documentation in English' component.

Explanation: You cannot select one of the Documentation in Other Languages subcomponents without selecting one of the Documentation in English subcomponents.

User Response: Select one or more of the Documentation in English subcomponents, one or more of both sets of subcomponents, or none of these.

AMQ4372   Uninstallation of the 'Documentation in English' component requires uninstallation of the 'Other Documentation Languages' component.

Explanation: You cannot uninstall the Documentation in English component without uninstalling the Documentation in Other Laguuages component.

User Response: Uninstall both components, just the Documentation in Other Languages component, or neither of these.

AMQ4373   There is not enough space on drive &s (program files) to install these components. Please free up some disk space or modify your selections.

Explanation: There is not enough space on the indicated drive to install the components you selected.

User Response: Select less components or increase the space available on the drive, or go back and select another drive for the program files top-level folder.

AMQ4374   There is not enough space on drive &s (data files) to install these components. Please free up some disk space or modify your selections.

Explanation: There is not enough space on the indicated drive to install the components you selected. User action: Select less components or increase the space available on the drive, or go back and select another drive for the data files top-level folder.

AMQ4375   The program files top-level folder is not valid.

Explanation: The program files top-level folder is not a valid path.

User Response: Enter a valid path.

AMQ4376   The data files top-level folder is not valid. (AMQ4376)

Explanation: The data files top-level folder is not a valid path.

User Response: Enter a valid path.

AMQ4377   The log files folder is not valid. (AMQ4377)

Explanation: The log files folder name is not a valid path.

User Response: Enter a valid path.

AMQ4378   A root folder is not allowed for the program files top-level folder. (AMQ4378)

Explanation: MQSeries cannot be installed in a root folder, for example 'c:\'.

User Response: Enter a non-root folder.

AMQ4379   A root folder is not allowed for the data files top-level folder. (AMQ4379)

Explanation: MQSeries cannot be installed in a root folder, for example 'c:\'.

User Response: Enter a non-root folder.

AMQ4380   A root folder is not allowed for the log files folder. (AMQ4380)

Explanation: MQSeries cannot be installed in a root folder, for example 'c:\'.

User Response: Enter a non-root folder.

AMQ4381   There is not enough space on drive &s (log files) to install these components. Please free up some disk space or modify your selections.

Explanation: There is not enough space on the indicated drive to install the components you selected. User action: Select less components or increase the space available on the drive, or go back and select another drive for the log files folder.

AMQ4382   Unable to create or replace folder '&s'.
Explanation: An error occurred creating the indicated folder.
User Response: Review the installation log file for details of any problems.

AMQ4383   Uninstallation cannot continue; failed to save queue manager configuration.
Explanation: An error occurred while saving the current queue manager configuration to a file.
User Response: Check that the registry keys under 'HKEY_LOCAL_MACHINE\SOFTWARE\IBM\MQSeries' are readable by an administrator. Check that there is enough space on the drive containing the data files folder (where the configuration is being saved in file \config\config.reg). If the error persists, contact your systems administrator.

AMQ4385   Unknown language specified ('&s').
Explanation: The -l parameter passed to setup.exe specified a language that was unrecognized.
User Response: Specify one of the language names supported by MQSeries Setup.

AMQ4386   Codepage (&d) for specified language not available.
Explanation: The codepage for the specified language is not installed on the computer.
User Response: Install the required language support. Additional language packages for Windows NT 4.0 can be found in the \Langpack folder on the Windows NT CD-ROM. The file 'readme.txt' in that directory explains how to install additional support. If the error persists, contact your systems administrator.

AMQ4387   Before Setup can display help, this computer's help system needs upgrading to HTML Help 1.1. Would you like to upgrade now? (You might need to restart the computer.)
Explanation: The help system on this computer needs upgrading.
User Response: Click Yes to upgrade the help system now (you might be asked to reboot), or No to cancel the help request.

AMQ4388   MQSeries Setup or uninstallation is already running.
Explanation: An attempt was made to run more than one instance of Setup or uninstallation at once.
User Response: Run only one copy of Setup or installation at a time.

AMQ4389   Setup could not create a local 'mqm' group (code &d).
Explanation: An error occurred creating a local user group called 'mqm'.
User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4390   Setup could not create a global 'mqm' group (code &d).
Explanation: An error occurred creating a local user group called 'mqm'.
User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4391   Setup could not find the global 'mqm' group. (AMQ4391)
Explanation: The global 'mqm ' group was created, but could not then be found.
User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4392   Setup could not add the global 'mqm' group to the local 'mqm' group (code &d).
Explanation: An error occurred adding the global 'mqm' group to the local 'mqm' group.
User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4393   No ports were specified; no listeners will be created.
Explanation: No TCP/IP ports were entered in the dialog, so no listeners will be created for the listed queue managers.
User Response: If this is what was intended then continue, otherwise go back to the dialog and enter one or more port numbers.

AMQ4394   No queue managers are selected for remote administration.
Explanation: No queue managers were selected from the list, so no queue managers will be enabled for remote administration.
User Response: If this is what was intended then continue, otherwise go back to the dialog and select one or more queue managers.

AMQ4395   One or more 'Server' component prerequisites were not selected; the component cannot be installed.
Explanation: One or more prerequisite software items for the Server component were not selected to be installed.

User Response: If the Server component is not required then continue, otherwise return to the dialog and select the prerequisite components, or exit Setup, install the prerequisite items, and restart Setup.

AMQ4396 **One or more prerequisite upgrades were not selected; MQSeries will not operate correctly.**

Explanation: One or more prerequisite software items for MQSeries were not selected to be installed.
User Response: Either return to the dialog and select the items, or exit Setup, install the prerequisite items, and restart Setup.

AMQ4397 **Cannot install on a network drive (drive &s).**

Explanation: One of the drives for the installation folders is a network drive.
User Response: Select installation folders that are on local drives.

AMQ4398 **Cannot start the selected application as TCP/IP is not installed on this machine.**

Explanation: In order to run either MQSeries Explorer, Postcard or Default Configuration applications, it is necessary to have TCP/IP installed and configured on your machine. All other First Steps application can still be run.
User Response: If you need to run these applications, you must install and configure TCP/IP on your machine.

AMQ4399 **Cannot start the selected application. Return code = &d.**

Explanation: There has been a general problem in starting the selected application. The return code is &d.
User Response: Retry the operation. If the problem persists, look up the return code in the NT documentation, or contact the systems administrator.

AMQ4500 **You are about to forcibly remove queue manager '&1' from cluster '&2'. Are you sure you want to forcibly remove the queue manager from the cluster?**

Explanation: None
User Response: You should only do this to remove a queue manager from a cluster when it has already been deleted and cannot be removed from the cluster in the normal way.

AMQ4501 **Queue manager successfully removed from cluster.**

Explanation: The queue manager may still appear as a member of the cluster until the configuration changes have propagated across the network.
User Response: None

AMQ4502 **You have shared the queue in cluster '&1'.**

Explanation: The queue manager is not a member of this cluster.
User Response: To make the queue available to the members of this cluster, you must join the queue manager to the cluster.

AMQ4503 **List of values too long. The list of values which you have entered is too long. The total number of significant characters allowed for this value is &1.**

AMQ4504 **Value too long. You have entered a value containing too many characters. The maximum number of characters allowed for each value of this attribute is &1.**

AMQ4505 **Too many entries in list. You have entered too many values in the list. The maximum number of values is &1.**

AMQ4506 **Cannot connect to queue manager '&1'.**

Explanation: The queue manager cannot be removed from the cluster in the normal way.
User Response: Retry the operation when the queue manager is available. If the queue manager no longer exists, you may choose to forcibly remove the queue manager from the cluster.

AMQ4507 **Remote queue manager not using TCP/IP.**

Explanation: The connection information available for the remote queue manager uses a communication protocol other than TCP/IP. The MQSeries Explorer cannot connect to the queue manager to remove it from the cluster in the normal way. If the queue manager no longer exists, you may choose to forcibly remove the queue manager from the cluster.
User Response: None

AMQ4508 **Queue manager successfully left cluster. The queue manager may still appear as a member of the cluster until the configuration changes have propagated across the network.**

Explanation: None
User Response: None

AMQ4509 **The request to suspend membership of the cluster has been accepted.**

AMQ4510 **The request to resume membership of the cluster has been accepted.**

AMQ4511 **The queue manager is not a member of the cluster.**

AMQ4512 **The MQSeries DLL '&1' cannot be loaded.**
Explanation: The MQSeries Explorer failed to load the DLL '&1'. This indicates that the MQSeries installation on this computer has been corrupted.
User Response: Re-install MQSeries on this computer to re-instate the DLL.

AMQ4513 **The request to refresh the information about the cluster has been accepted.**

AMQ4514 **The queue manager is not a member of cluster '&1'.**
Explanation: The object which you have shared in the cluster will not be available to other members of the cluster until you make this queue manager a member of the cluster.
User Response: None

AMQ4550 **The two queue manager names must be different.**
Explanation: You entered the same name for the two queue managers that are to be made repositories for the cluster. The two repositories must be made on different queue managers.
User Response: Enter a different name for one of the repository queue managers.

AMQ4551 **Unable to connect to the queue manager called 'name'. Please check the correct queue manager has been chosen and that the queue manager is started.**
Explanation: The queue manager you choose to be made a repository for the cluster must already exist, and be started.
User Response: Enter the name of an existing queue manager that is started.

AMQ4552 **Queue manager 'name' is already a repository queue manager. Choose another queue manager and continue with the Wizard, otherwise refer to the MQSeries Queue Manager Clusters manual.**
Explanation: The Wizard is only designed to work with queue managers that are not already a repository queue manager.
User Response: Choose a queue manager that is not already a repository queue manager. If you want to work with a queue manager that is already a repository, then you must enter the necessary

commands yourself, refer to the MQSeries Queue Manager Clusters manual for more information.

AMQ4553 **Queue manager 'name' is not a repository queue manager. Please choose another queue manager.**
Explanation: The Wizard needs to know the name of one of the queue managers that is already a repository for the cluster. The queue manager you entered is not a repository queue manager.
User Response: Enter the name and connection name of one of the existing repository queue managers for the cluster.

AMQ4554 **Please enter no more than number characters.**
Explanation: You entered too many characters.
User Response: The name you choose must be no longer than number characters long.

AMQ4555 **The port number 'number' is not valid. Please choose another number.**
Explanation: The number you entered is not a valid port number.
User Response: Enter a valid number.

AMQ4556 **The port number 'number' is already being used by another queue manager. Please choose another number.**
Explanation: The port number you entered is already in use.
User Response: Choose a different port number.

AMQ4557 **Unable to use name 'name', a cluster receiver with this name already exists on queue manager 'queue-manager'. Please choose another name.**
Explanation: Cluster receiver channel names on a queue manager must be unique. A cluster receiver channel with the name you entered already exists.
User Response: Choose a different name.

AMQ4558 **Cluster receiver 'name' does not exist on queue manager ' queue-manager'. Please enter the correct name.**
Explanation: The Wizard needs to know the name of the existing cluster receiver channel on the queue manager. The name you entered does not exist.
User Response: Enter the correct name of the cluster receiver channel.

AMQ4559 **Unable to use name 'name', a channel with this name already exists on queue manager 'queue-manager'. Resolving this situation requires renaming channels or using namelists, which is beyond the scope of this wizard.**

Explanation: To communicate with the repository queue manager, a new cluster sender needs to be created on the joining queue manager 'queue-manager' that matches the name of the cluster receiver channel 'name' on the repository. A channel with that name already exists on the joining queue manager.

User Response: To resolve this situation you need to rename the channels on the cluster, or use a namelist. Refer to the MQSeries Queue Manager Clusters manual for more information.

AMQ4560 **Unable to create a repository on queue manger 'queue-manager' on connection 'connection-name'.**

Explanation: The Wizard was unable to create a repository on the queue manager.

User Response: Use the 'Details>>' button to show further details about the problem and refer to the MQSeries Queue Manager Clusters manual or to your systems administrator.

AMQ4561 **Unable to create cluster sender channel 'name' on queue manager 'queue-manager'.**

Explanation: The Wizard was unable to create a cluster sender channel on the queue manager.

User Response: Use the 'Details>>' button to show further details about the problem and refer to the MQSeries Queue Manager Clusters manual or to your systems administrator.

AMQ4562 **Unable to create cluster receiver channel 'name' on queue manager 'queue-manager'.**

Explanation: The Wizard was unable to create a cluster receiver channel on the queue manager.

User Response: Use the 'Details>>' button to show further details about the problem and refer to the MQSeries Queue Manager Clusters manual or to your systems administrator.

AMQ4563 **Unable to start a listener on queue manager 'queue-manager'.**

Explanation: The Wizard was unable to start a listener on the queue manager.

User Response: Use the 'Details>>' button to show further details about the problem and refer to the MQSeries Queue Manager Clusters manual or to your systems administrator.

AMQ4564 **Unable to use name 'name', a channel with this name already exists on queue manager 'queue-manager'. Please choose another name.**

Explanation: Channel names on a queue manager must be unique. A channel with the name you entered already exists.

User Response: Choose a different name.

AMQ4703 **One or more problems occurred during Setup. Review '&s' for details.**

Explanation: Setup was only partially successful.

User Response: Review the installation log file for details of any problems.

AMQ4704 **If specified, TCP/IP domain must be '&s'.**

Explanation: The computer with the cluster repository must be in the same TCP/IP domain as this computer.

User Response: Enter the name of a computer in the local domain.

AMQ4705 **Current service level is '&s'. Re-apply service after Setup completes.**

Explanation: Some service has been applied to the current installation. Installation or reinstallation of MQSeries components might regress some files.

User Response: Review the instructions that came with the service that was applied. If necessary re-apply the service.

AMQ4706 **Dialog '&s' failed.**

Explanation: An internal error occurred displaying a dialog.

User Response: Retry. If the error persists, contact your systems administrator.

AMQ4707 **Error migrating '&s'.**

Explanation: An error occurred migrating a .ini file to the registry.

User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4708 **Error creating remote administration channel for queue manager '&s'.**

Explanation: A channel for the indicated queue manager could not be created.

User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4709   **Error creating TCP/IP listener for queue manager '&s'.**

Explanation: A listener for the indicated queue manager could not be created.

User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4710   **Error updating '&s' environment variable.**

Explanation: An error occurred updating the indicated environment variable in the registry.

User Response: Review the installation log file for details of any problems. If the error persists, contact your systems administrator.

AMQ4711   **One or more problems occurred during uninstallation. Review '&s' for details.**

Explanation: Uninstallation was only partially successful.

User Response: If the installation log file is available, review it for details of any problems. If the error persists, contact your systems administrator.

Chapter 4. AMQ5000-AMQ5999 Installable services messages

> **Reading variables in the message**
> Some messages display text or numbers that vary according to the circumstances giving rise to the message; these are known as *message variables*. The message variables are indicated in this book by the use of the '&' symbol and a number: &1, &2, and so on.
>
> In some cases a message may have variables in the Explanation or User action. Find the values of the message variables by looking in the error log. The complete message, including the Explanation and the User action, is recorded there.

AMQ5005   Unexpected error
Explanation: An unexpected error occurred in an internal function of the product.
User Response: Save the generated output files and contact your IBM support center.

AMQ5006   Unexpected error: rc = &1
Explanation: An unexpected error occurred in an internal function of the product.
User Response: Save the generated output files and contact your IBM support center.

AMQ5008   An essential MQSeries process &1 cannot be found and is assumed to be terminated.
Explanation: 1) A user has inadvertently terminated the process. 2) The system is low on resources. Some operating systems terminate processes to free resources. If your system is low on resources, it is possible it has terminated the process so that a new process can be created.
User Response: MQSeries will stop all MQSeries processes. Inform your systems administrator. When the problem is rectified MQSeries can be restarted.

AMQ5009   MQSeries agent process &1 has terminated unexpectedly.
Explanation: MQSeries has detected that an agent process has terminated unexpectedly. The queue manager connection(s) that this process is responsible for will be broken.
User Response: Use any previous FFSTs to determine the reason for the failure. Try to eliminate the following reasons before contacting your IBM support center.

1) A user has inadvertently terminated the process.

2) The system is low on resources. Some operating systems terminate processes to free resources. If your system is low on resources, it is possible that the operating system has terminated the process so that a new process can be created.

AMQ5010   The system is restarting the WorkLoad Management Server process.
Explanation: The system has detected that the WorkLoad Management server process (amqzlwa0, pid:&1) has stopped and is restarting it.
User Response: Save the generated output files which may indicate the reason why the WorkLoad Management process stopped.

If the reason the WorkLoad Management Server process stopped is a problem in a WorkLoad Management user exit correct the problem, otherwise contact your IBM support center.

AMQ5011   MQSeries repository process &1 has ended unexpectedly.
Explanation: MQSeries has detected that the repository process has ended unexpectedly.
User Response: Use any previous FFSTs to determine the reason for the failure. Try to eliminate the following reasons before contacting your IBM support center.

1.A user has inadvertently ended the process.

2.The system is low on resources. Some operating systems end processes to free resources. If your system is low on resources, it is possible that the operating system has ended the process in order to create a new process.

AMQ5203   An error occurred calling the XA interface.
Explanation: The error number is &2 where a value of

1 indicates the supplied flags value of &1 was invalid,

2 indicates that there was an attempt to use threaded and non-threaded libraries in the same process,

3 indicates that there was an error with the supplied queue manager name '&3',

4 indicates that the resource manager id of &1 was invalid,

5 indicates that an attempt was made to use a second queue manager called '&3' when another queue manager was already connected,

6 indicates that the Transaction Manager has been called when the application isn't connected to a queue manager, and

7 indicates that the XA call was made while another call was in progress.
User Response: Correct the error and try the operation again.

AMQ5204   A non-threaded application tried to run as a Trusted application.
Explanation: Only applications linked with the threaded MQ libraries can run as Trusted applications.
User Response: Make sure that the application is relinked with the threaded MQ libraries, or set the environment variable MQ_CONNECT_TYPE to STANDARD.

AMQ5358   MQSeries could not load AX support module '&3'.
Explanation: An error has occurred loading the AX support module &3. This module needs to be loaded so that dynamically-registering resource managers, such as DB2®, can participate in global units of work.
User Response: Look for a previous message outlining the reason for the load failure. Message AMQ6175 should have been issued if the load failed because of a system error. If this is the case then follow the guidance given in message AMQ6175 to resolve the problem. In the absence of prior messages or FFST™ information related to this problem check that the AX support module and the mqmax library have been correctly installed on your system.

AMQ5501   There was not enough storage to satisfy the request
Explanation: An internal function of the product attempted to obtain storage, but there was none available.
User Response: Stop the product and restart it. If this does not resolve the problem, save the generated output files and contact your IBM support center.

AMQ5502   The CDS directory name '&3' is not in the correct format.
Explanation: An internal function of the DCE Naming service found a CDS directory name in the wrong format. The name was expected to start with either '/...' for a fully qualified name (from global root), or '/.:' for
User Response: Save the generated output files and

contact your IBM support center.

AMQ5503   The name of the local DCE cell cannot be determined, status = &1
Explanation: The DCE Naming Service attempted to determine the name of the local DCE cell by calling 'dce_cf_get_cell_name()', which returned a nonzero return code.
User Response: Save the generated output files and contact your IBM support center.

AMQ5504   DCE error. No value for the XDS attribute found.
Explanation: The DCE Naming service called om_get() to get the entry from the object returned by ds_read(). Although the status was correct, no objects were returned.
User Response: Save the generated output files and contact your IBM support center.

AMQ5505   DCE error. No value for the XDS attribute number &1 found.
Explanation: The DCE Naming service called om_get() to get the entry from the object returned by ds_read(). Although the status was correct, no objects were returned.
User Response: Save the generated output files and contact your IBM support center.

AMQ5506   DCE error. '&3' returned &1 for attribute number &2.
Explanation: The DCE Naming service queried an object by calling '&3' which returned a nonzero return code.
User Response: Save the generated output files and contact your IBM support center.

AMQ5507   DCE error. '&3' failed for an unknown reason.
Explanation: An unexpected error occurred in an internal function of the DCE Naming service.
User Response: Save the generated output files and contact your IBM support center.

AMQ5508   DCE error. The requested attribute is not present.
Explanation: The DCE Naming service was attempting to extract the value from an attribute, but the attribute cannot be found in the XDS object.
User Response: Save the generated output files and contact your IBM support center.

AMQ5509   DCE error. The XDS workspace cannot be initialized.
Explanation: The DCE Naming service called 'ds_initialize()' to initialize the XDS workspace, but 'ds_initialize()' returned a nonzero return code.

User Response: Save the generated output files and contact your IBM support center.

AMQ5510   DCE error. '&3' returned with problem &1.
Explanation: The DCE Naming service found an unexpected XDS error.
User Response: Save the generated output files and contact your IBM support center.

AMQ5511   Installable service component '&3' returned '&4'.
Explanation: The internal function, that adds a component to a service, called the component initialization process. This process returned an error.
User Response: Check the component was installed correctly. If it was, and the component was supplied by IBM, then save the generated output files and contact your IBM support center. If the component was not supplied by IBM, save the generated output files and follow the support procedure for that component.

AMQ5512   Installable service component '&3' returned '&4' for queue manager name = '&5'.
Explanation: An installable service component returned an unexpected return code.
User Response: Check the component was installed correctly. If it was, and the component was supplied by IBM, then save the generated output files and contact your IBM support center. If the component was not supplied by IBM, save the generated output files and follow the support procedure for that component.

AMQ5513   '&3' returned &1.
Explanation: An unexpected error occurred.
User Response: Save the generated output files and contact your IBM support center.

AMQ5519   Bad DCE identity. Status = &1, auth = &2, keytab file = '&3', principal = '&4'.
Explanation: The keytab file was not installed correctly, or the MQSeries user ID has a different password from that used to create the keytab file.
User Response: Make sure that the MQSeries user ID defined when the product was installed has the same password as that defined by the keytab file, and that the keytab file has been installed correctly.

AMQ5520   The system could not load the module '&5' for the installable service '&3' component '&4'. The system return code was &1. The Queue Manager is continuing without this component.
Explanation: The queue manager configuration data included a stanza for the installable service '&3' component '&4' with the module '&5'. The system returned &1 when it tried to load this module. The

Queue Manager is continuing without this component.
User Response: Make sure that the module can be loaded. Put the module into a directory where the system can load it, and specify its full path and name in the configuration data . Then stop and restart the queue manager.

AMQ5521   The system could not open "&3".
Explanation: The system failed to open the default object "&3" at connect time for reason &4. This may be because "&3" has been deleted or changed.
User Response: Recreate the default objects by running "strmqm -c <qmgr>" (where <qmgr> is the name of the queue manager) and retry the application.

AMQ5522   An MQSeries installable service component could not be initialized.
Explanation: An installable service component returned an unexpected return code.
User Response: Check the queue manager error logs for messages explaining which installable service could not be initialized and why that service could not be initialized. Check the component was installed correctly. If it was, and the component was supplied by IBM, then save the generated output files and contact your IBM support center. If the component was not supplied by IBM, save the generated output files and follow the support procedure for that component.

AMQ5524   The MQSeries Object Authority Manager has failed to migrate authority data
Explanation: The MQSeries Object Authority Manager has attempted to migrate existing queue manager authority data from a previous version of an MQSeries Object Authority Manager and failed.
User Response: Check this log for any previous related messages, follow their recommendations then restart the queue manager.

AMQ5525   The MQSeries Object Authority Manager has failed.
Explanation: The MQSeries Object Authority Manager has failed to complete an MQSeries request.
User Response: Check the queue manager error logs for messages explaining the failure and try to correct the problem accordingly.

AMQ5526   The MQSeries Object Authority Manager has failed with reason &1
Explanation: The MQSeries Object Authority Manager has failed an operation on the Object Authority Manager's data queue '&3' with reason &1.
User Response: Investigate why the error has occurred and correct the problem.

| **AMQ5527** | **The MQSeries Object Authority Manager has failed to locate an essential authority file** |

Explanation: The MQSeries Object Authority Manager has failed to locate the authority file '&3'. The migration of authority data cannot continue until the file has been restored. The queue manager will shutdown.
User Response: Restore the authority file mentioned above and restart the queue manager.

| **AMQ5528** | **The MQSeries Object Authority Manager has failed to locate an object's authority file** |

Explanation: The MQSeries Object Authority Manager has failed to locate the authority file for the object '&3' of type &1. The authority access to this object will initially be limited to members of the mqm group.
User Response: To extend access to this object use the setmqaut command, see the MQSeries System Administration documentation for details.

| **AMQ5600** | **Usage: crtmqm [-z] [-q] [-c Text] [-d DefXmitQ] [-h MaxHandles] [-g ApplicationGroup]** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5601** | **[-t TrigInt] [-u DeadQ] [-x MaxUMsgs] [-lp LogPri] [-ls LogSec]** |

Explanation:
User Response: None.

| **AMQ5602** | **[-lc | -ll] [-lf LogFileSize] [-ld LogPath] QMgrName** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5603** | **Usage: dltmqm [-z] QMgrName** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5604** | **Usage: dspmqaut [-m QMgrName] [-n ObjName] -t ObjType [-p Principal | -g Group] [-s ServiceName]** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5605** | **Usage: endmqm [-z] [-c | -w | -i | -p] QMgrName** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5606** | **Usage: setmqaut -m QMgrName [-n ObjName] -t ObjType [-p Principal | -g Group] [-s ServiceName] Authorizations** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5607** | **Usage: strmqm [-z][-c] [QMgrName]** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5608** | **Usage: dspmqtrn [-m QMgrName] [-e] [-i]** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5609** | **Usage: rsvmqtrn -m QMgrName (-a | ((-b | -c | -r RMId) Transaction,Number))** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5610** | **Usage: strmqtrc [-m QMgrName] [-e] [-t TraceType] [-l MaxFileSize]** |

Explanation: This applies to UNIX systems. MaxFileSize is the maximum size of a trace file in millions of bytes.
User Response: None.

| **AMQ5610** | **Usage: strmqtrc [-t TraceType] [-l MaxFileSize]** |

Explanation: This applies to Windows NT and Windows 2000 systems only. MaxFileSize is the maximum size of a trace file in millions of bytes.
User Response: None.

| **AMQ5611** | **Usage: endmqtrc** |

Explanation: This applies to Windows NT and Windows 2000 systems only.
User Response: None.

| **AMQ5611** | **Usage: endmqtrc [-m QMgrName] [-e] [-a]** |

Explanation: This applies to UNIX systems.
User Response: None.

| **AMQ5612** | **Usage: dspmqtrc [-t TemplateFile] [-h] [-o OutputFileName] InputFileName** |

Explanation: This shows the correct usage.
User Response: None.

| **AMQ5613** | **Usage: scmmqm (-a -s Service command file | -d) [-z] QMgrName** |

Explanation: This shows the correct usage.
User Response: None.

AMQ5614   Usage: setmqtry
Explanation: This shows the correct usage.
User Response: None.

AMQ5615   Default objects cannot be created:
CompCode = &1 Reason = &2.
Explanation: During the creation of a queue manager,
using the crtmqm command, the default objects could
not be created. Possible reasons for this include another
command, issued elsewhere, quiescing or stopping the
queue manager, or insufficient storage being available.
User Response: Use the Completion and Reason codes
shown in the message to determine the cause of the
failure, then re-try the command.

AMQ5616   Usage: setmqprd [LicenseFile]
Explanation: This shows the correct usage.
User Response: None.

AMQ5617   Default objects cannot be created.
Explanation: During the creation of a queue manager
using the crtmqm command, the default objects could
not be created. The most likely reason for this error is
that the queue manager was started before the crtmqm
command had completed.
User Response: Ensure that the queue manager being
created is not started before the create request
completes. Stop the queue manager if it is already
running. Restart the queue manager using the strmqm
command with the '-c' option to request that the
default objects are created.

AMQ5635   The argument supplied with the -l flag
is not valid.
Explanation: The argument supplied with the -l flag
must be in the range 1 - 4293.
User Response: Submit the command again with a
valid argument.

AMQ5637   The environment variable EXTSHM is
set to "ON". This is incompatible with
the way MQSeries uses shared memory.
Reset the environment variable
EXTSHM and retry the command.
Explanation: On AIX the environment variable
EXTSHM causes shared memory segments to be fixed
size. MQSeries expects to be able to extend shared
memory segments.
User Response: Reset the environment variable
EXTSHM and retry the command.

AMQ5700   Queue manager name '&3', work queue
name '&4'.
Explanation: These are the values of the parameters
with which the add-in task was started.
User Response: None.

AMQ5701   Checking mail-in database &3
Explanation: The add-in task is performing a periodic
check for mail memos that have arrived in the mail-in
database called &3.
User Response: None.

AMQ5702   Checking for replies.
Explanation: The add-in task is checking the reply
queues for responses from MQSeries applications.
User Response: None

AMQ5703   MQSeries add-in task ended.
Explanation: Termination of the MQSeries add-in task
has completed.
User Response: None.

AMQ5704   Terminating.
Explanation: The add-in task is terminating, either
due to a user request or an error.
User Response: None.

AMQ5705   Initializing.
Explanation: The add-in task is initializing. It
processes the link database and connects to the queue
manager in preparation to receive requests.
User Response: None.

AMQ5706   Mail-in database '&3', link database
'&4', wait time &1 seconds.
Explanation: These are the values of the parameters
with which the add-in task was started.
User Response: None.

AMQ5707   Add-in task initialization complete.
Modification date &3.
Explanation: The add-in task has finished reading the
link database and is now ready to process requests. The
Modification date is the last change date of MQLINK.
User Response: None.

AMQ5708   Only two-byte integer values are
supported for S/390 format.
Explanation: The add-in task supports conversion of
two-byte integers from S/390® systems.
User Response: Ensure that the entry in the link
database uses fields of only two bytes in length if they
are in the S/390 format.

AMQ5710   Text of user document causing previous
message: '&3'.
Explanation: The add-in task generated the previous
message in response to an error. This message contains
the text of the user note associated with the error.
User Response: None.

AMQ5711   An error occurred in reading the link database.

Explanation: The add-in task detected an error while reading the link database.

User Response: Use the information in previous error messages to diagnose the error. Then, correct the contents of the link database and restart the add-in task.

AMQ5712   An error occurred while setting field '&5' in user document, return code &3.

Explanation: The add-in task was trying to update a document in response to a reply from an MQSeries application. An error was encountered during the update of the field '&5'. The link database entry '&4' was being used to perform the update.

User Response: Make sure that the entry in the link database matches the description of the form being used for the update.

AMQ5714   Field '&4' not found in link database entry.

Explanation: The add-in task could not find a field called '&4' during processing of the link database. This field is a required field.

User Response: Examine the definition of the link database being used to ensure that all of the required fields are supplied. Refer to the IBM-supplied sample link database for an example of a valid link database.

AMQ5715   Data type '&4' not supported.

Explanation: The add-in task does not support the data type '&4'.

User Response: Consult the MQSeries documentation for a description of the list of supported data types. Update the entry in the link database using the unsupported data type. Then, stop and restart the add-in task.

AMQ5716   An error occurred connecting to MQSeries queue manager '&4', MQSeries reason code &3.

Explanation: The add-in task could not connect to MQSeries queue manager '&4'. The reason code from MQCONN was &3.

User Response: Look up the reason code in the MQSeries documentation to establish the cause of the error. Ensure that the queue manager exists and is running. If the add-in task is running as an MQSeries client, ensure that it can communicate with the server queue manager.

AMQ5717   An error occurred disconnecting from MQSeries queue manager '&4', MQSeries reason code &3.

Explanation: The add-in task encountered an error disconnecting from the MQSeries queue manager '&4'. The reason code from MQDISC was &3.

User Response: Look up the reason code in the MQSeries documentation to establish the cause of the error.

AMQ5718   An error occurred during processing of a request in the mail-in database.

Explanation: The add-in task encountered an error during processing of a request in the mail-in database. The processing involves transforming the contents of the mail memo into a message which is placed on an MQSeries queue. If the message has a reply, an additional message is formatted and placed on the internal work queue.

User Response: Use the information in previous error messages to diagnose the error.

AMQ5720   Errors detected in response message from MQSeries application.

Explanation: The response from an MQSeries application to a message sent by the add-in task satisfied the error conditions specified in the corresponding link database entry. The error data is '&4'.

User Response: Examine the error conditions in the link database entry to establish why the error conditions were satisfied. If an invalid request message was sent to the MQSeries application, correct the request messages being sent. If the problem was due to an error encountered by the MQSeries application, correct the cause of the error and retry the request.

AMQ5721   An error occurred opening internal work file '&4'.

Explanation: The add-in task could not open the internal work file used to hold the contents of a mail memo during processing. Possible causes include more than one program trying to use the same file.

User Response: Ensure that there is only one copy of the MQSeries add-in task running.

AMQ5723   Memory allocation failed.

Explanation: The add-in task was unable to allocate storage.

User Response: Try to free up some system memory and retry the operation.

AMQ5725   Empty mail memo received from mail-in database.

Explanation: The add-in task found a mail memo with an empty body in the mail-in database. Mail memos in the mail-in database must contain the information required to generate a message to place on an MQSeries queue.

User Response: Ensure that all entries placed in the mail-in database have the expected contents.

AMQ5727 **Link database entry '&4' cannot be found.**

Explanation: The add-in task received a request without a corresponding entry in the link database. The name of the required entry is '&4'.

User Response: Either add an entry of the correct name to the link database or change the request being generated to use an existing entry in the link database. If you add an entry to the link database, you will have to stop and restart the add-in task before the change takes effect.

AMQ5729 **An error was encountered by the add-in task. Check the mail for details.**

Explanation: This message is inserted into the error_field_msg field of a user document if an error is encountered by the add-in task during the processing of the document's associated mail memo.

User Response: None.

AMQ5730 **Error encountered by MQSeries add-in task**

Explanation: This is the subject line of mail memos sent by the add-in task.

User Response: None.

AMQ5731 **Idle.**

Explanation: The add-in task is waiting for the configured time interval to elapse before checking the mail-in database for new requests and checking the reply queues for new replies.

User Response: None.

AMQ5732 **LOAD MQLINK [-t] [-q WorkQName] [-w WaitTime] [-d MailInDB] [-l LinkDB] [QMgrName]**

Explanation: This is a summary of the correct syntax for invoking the MQSeries add-in task in Lotus® Notes™. If you specify a queue manager name, it must the last parameter. The order of the other parameters is not significant.

User Response: None.

AMQ5733 **MQSeries add-in task loading.**

Explanation: The add-in task has been started and is accessing the link database in preparation to receive requests.

User Response: None.

AMQ5734 **An error occurred opening the database '&4'. The error code was &3.**

Explanation: The add-in task could not open the named database. This could be because the database does not exist.

User Response: Refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5735 **An error occurred opening the mail file '&4'. The error code was &3.**

Explanation: The add-in task could not open the named mail file.

User Response: Refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5736 **An error occurred searching the database '&4'. The error code was &3.**

Explanation: The add-in task could not search the named database.

User Response: Refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5737 **An error occurred deleting an entry from the database '&4'. The error code was &3.**

Explanation: The add-in task could not delete an entry from the named database.

User Response: Refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5738 **An error occurred extracting the contents of a mail memo in the mail-in database '&5' to the file called '&4'. The error code was &3.**

Explanation: The add-in task could not extract the body of a mail memo into the named file. Possible causes include being unable to create the file or another program already using the file.

User Response: Ensure that there is only one copy of the MQSeries add-in task running. If the problem was due to the configuration in which you are operating Lotus Notes, refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5739 **An error occurred opening a mail memo in the mail-in database '&4'. The error code was &3.**

Explanation: The add-in task could not open a mail memo in the named mail-in database.

User Response: Refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5740 **An error occurred opening an entry in the link database '&4'. The Notes error code was &3.**

Explanation: The add-in task could not open an entry in the link database.

User Response: Refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5741 **An error occurred creating a mail memo. The error code was &3.**

Explanation: The add-in task could not create a mail memo. This is probably due to a shortage of resources.

User Response: Refer to the Lotus Notes

AMQ5742 • AMQ5747

documentation for information to resolve the problem.

AMQ5742   Could not send a mail memo to user '&4'. The error code was &3.
Explanation: The add-in task could not send a mail memo to the named user to report an error condition.
User Response: Refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5743   Could not find entry with ID '&5' in database '&4'. The error code was &3.
Explanation: The add-in task could not find an entry in the database '&4' which it was to update in response to a reply from an MQSeries application. This may indicate that the entry has been manually deleted or that another application has already updated the entry.
User Response: Refer to the Lotus Notes documentation for information to resolve the problem.

AMQ5744   Could not update an entry in database '&4'. The error code was &3.
Explanation: The add-in task could not update an entry in the database '&4' in response to a reply from an MQSeries application.
User Response: If the 'error code' is -1 (65535) the reply buffer is too short to be able to update the Notes document. If the 'error code' is greater than 0 the value has been returned from a Lotus Notes API function. Refer to the Lotus Notes documentation for further information.

If the 'error code' is 0 refer to previous messages for an explanation.

AMQ5745   An error occurred opening MQSeries queue '&4', reason code &3. &5
Explanation: The add-in task could not open MQSeries queue '&4'. The reason code from MQOPEN was &3. The reason may be explained by &5.
User Response: Look up the reason code in the MQSeries documentation to establish the cause of the error. If a version of MQLINK is running with the same work queue, for example with 'SYSTEM.NOTES.WORKQUEUE', then you will get this message and reason code 2042.

AMQ5746   An error occurred putting a message on MQSeries queue '&4', reason code &3.
Explanation: The add-in task could not put a message on MQSeries queue '&4'. The reason code from MQPUT was &3.
User Response: Look up the reason code in the MQSeries documentation to establish the cause of the error.

AMQ5747   An error occurred getting a message from MQSeries queue '&4', reason code &3.
Explanation: The add-in task could not get a message from MQSeries queue '&4'. The reason code from MQGET was &3.
User Response: Look up the reason code in the MQSeries documentation to establish the cause of the error.

Chapter 5. AMQ6000-AMQ6999 Common services messages

> **Reading variables in the message**
>
> Some messages display text or numbers that vary according to the circumstances giving rise to the message; these are known as *message variables*. The message variables are indicated in this book by the use of the '&' symbol and a number: &1, &2, and so on.
>
> In some cases a message may have variables in the Explanation or User action. Find the values of the message variables by looking in the error log. The complete message, including the Explanation and the User action, is recorded there.

AMQ6004   **An error occurred during MQSeries initialization or ending.**
Explanation: An error was detected during initialization or ending of MQSeries. The MQSeries error recording routine has been called.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6015   **The operating system is either too busy or has insufficient resources to complete a system request.**
Explanation: A system request '&3' was rejected by the operating system with return code '&1'. MQSeries retried the request, but it continued to fail. This failure may indicate that the operating system is either too busy or has insufficient resources to complete the request.
User Response: Investigate whether the system is constrained by the workload on this system or by the workload on a server that it is using, and reduce the workload.

AMQ6025   **Program not found.**
Explanation: MQSeries is unable to start program &3 because it was not found.
User Response: Check the program name is correctly specified and rerun the program.

AMQ6026   **A resource shortage prevented the creation of an MQSeries process.**
Explanation: An attempt to create an MQSeries process was rejected by the operating system due to a process limit (either the number of processes for each user or the total number of processes running system wide), or because the system does not have the resources necessary to create another process.
User Response: Investigate whether a process limit is preventing the creation of the process and if so why the system is constrained in this way. Consider raising this limit or reducing the workload on the system.

AMQ6035   **MQSeries failed, no storage available.**
Explanation: An internal function of the product attempted to obtain storage, but there was none available.
User Response: Stop the product and restart it. If this does not resolve the problem, save the generated output files and contact your IBM support center.

AMQ6037   **MQSeries was unable to obtain enough storage.**
Explanation: The product is unable to obtain enough storage. The product's error recording routine may have been called.
User Response: Stop the product and restart it. If this does not resolve the problem see if a problem has been recorded. If a problem has been recorded, use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6047   **Conversion not supported.**
Explanation: MQSeries is unable to convert string data tagged in CCSID &1 to data in CCSID &2.
User Response: Check the appropriate National Language Support publications to see if the CCSIDs are supported by your system.

AMQ6048   **DBCS error**
Explanation: MQSeries is unable to convert string data due to a DBCS error. Conversion is from CCSID &1 to CCSID &2.
User Response: Check the appropriate National Language Support publications to see if the CCSIDs are supported by your system.

AMQ6049   DBCS-only string not valid.
Explanation: MQSeries is unable to convert string data in CCSID &1 to data in CCSID &2. Message descriptor data must be in single-byte form. CCSID &2 is a DBCS-only CCSID.
User Response: Check the CCSID of your job or system and change it to one supporting SBCS or mixed character sets. Refer to the appropriate National Language Support publications for character sets and CCSIDs supported.

AMQ6050   CCSID error.
Explanation: MQSeries is unable to convert string data in CCSID &1 to data in CCSID &2.
User Response: Check the appropriate National Language Support publications to see if the CCSIDs are supported by your system.

AMQ6051   Conversion length error.
Explanation: MQSeries is unable to convert string data in CCSID &1 to data in CCSID &2, due to an input length error.
User Response:

AMQ6052   Conversion length error.
Explanation: MQSeries is unable to convert string data in CCSID &1 to data in CCSID &2.
User Response:

AMQ6053   CCSID error
Explanation: MQSeries is unable to convert string data in CCSID &1 to data in CCSID &2.
User Response: One of the CCSIDs is not supported by the system. Check the appropriate National Language Support publications to see if the CCSIDs are supported by your system.

AMQ6064   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6090   MQSeries was unable to display an error message &6.
Explanation: MQSeries has attempted to display the message associated with return code hexadecimal '&6'. The return code indicates that there is no message text associated with the message. Associated with the request are inserts &1 : &2 : &3 : &4 : &5.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM

support center. Do not discard these files until the problem has been resolved.

AMQ6091   An internal MQSeries error has occurred.
Explanation: Private memory has detected an error, and is abending due to &3. The error data is &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6092   Manual conversion required for CCSID: &1
Explanation: CCSID &1 exists in new format but could not be reconciled against your old format.
User Response: Manually edit CCSID entry &1 in conv\table\ccsid.tbl if you wish to retain your old conversion. For assistance call your Service Representative.

AMQ6100   An internal MQSeries error has occurred.
Explanation: MQSeries has detected an error, and is abending due to &3. The error data is &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6107   CCSID not supported.
Explanation: MQSeries is unable to convert string data in CCSID &1 to data in CCSID &2, because one of the CCSIDs is not recognized.
User Response: Check the appropriate National Language Support publications to see if the CCSIDs are supported by your system.

AMQ6109   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6110   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM

support center. Do not discard these files until the problem has been resolved.

AMQ6115   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6118   An internal MQSeries error has occurred (&1)
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6119   An internal MQSeries error has occurred (&3)
Explanation: MQSeries detected an unexpected error when calling the operating system. The MQSeries error recording routine has been called.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6120   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6121   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: MQSeries has detected a parameter count of &1 that is not valid. Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6122   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: MQSeries has detected parameter &1 that is not valid, having value &2&3. Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6125   An internal MQSeries error has occurred.
Explanation: An internal error has occurred with identifier &1. This message is issued in association with other messages.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6148   An internal MQSeries error has occurred.
Explanation: MQSeries has detected an error, and is abending due to &3. The error data is &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6150   MQSeries semaphore is busy.
Explanation: MQSeries was unable to acquire a semaphore within the normal timeout period of &1 minutes.
User Response: MQSeries will continue to wait for access. If the situation does not resolve itself and you suspect that your system is locked then investigate the process which owns the semaphore. The PID of this process will be documented in the accompanying FFST.

AMQ6162   An error occurred when reading the configuration data.
Explanation: An error has occurred when reading the configuration data.
User Response: If you have changed the configuration data, check and correct the change. If you have not changed the configuration data, use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6163   An error has occurred locking the configuration data.
Explanation: An error has occurred locking the configuration data.
User Response: If you have changed the registry permissions, check and correct the change. If you have not changed the registry, use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6164   An expected stanza in the configuration data is missing or contains errors.
Explanation: An expected stanza is missing from the configuration data or the stanza contains errors.
User Response: If you have changed the configuration data, check and correct the change.

AMQ6165   Unable to access the configuration data.
Explanation: Access to the configuration data is denied.
User Response: If you have changed the configuration data permissions, check and correct the changes.

AMQ6166   An entry in the configuration data is missing.
Explanation: A required entry in the configuration data is missing.
User Response: If you have changed the configuration data, recover the previous configuration data and retry the operation.

AMQ6172   No codeset found for current locale.
Explanation: No codeset could be determined for the current locale. Check that the locale in use is supported.
User Response: None.

AMQ6173   No CCSID found for codeset &3.
Explanation: Codeset &3. has no supported CCSID. Check that the locale in use is supported. CCSIDs can be added by updating the file /var/mqm/conv/table/ccsid.tbl.
User Response: None.

AMQ6174   The library &3 was not found. The queue manager will continue without this module.
Explanation: The dynamically loadable file &3 was not found.
User Response: Check that the file exists and is either fully qualified or is in the appropriate directory.

AMQ6175   The system could not dynamically load the library &3. The system return code was &1. The queue manager will continue without this module.
Explanation: This message applies to Windows NT and Windows 2000 systems only. The dynamically loadable file &3 failed to load correctly due to an internal error. and the MQSeries error recording routine has been called.
User Response: Check that the file has not been corrupted then use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6175   The system could not dynamically load the library &3. The error message was &4. The Queue Manager will continue without this module.
Explanation: This message applies to Sun Solaris systems. The dynamically loadable file &3 failed to load correctly due to an internal error. and the MQSeries error recording routine has been called.
User Response: Check that the file has not been corrupted then use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6175   The system could not dynamically load the library &3. The error message was &4. The errno was &1, check the errno in sys/errno.h. The queue manager will continue without this module.
Explanation: This message applies to AIX, HP-UX and Linux systems. The dynamically loadable file &3 failed to load correctly due to an internal error. and the MQSeries error recording routine has been called.
User Response: Check that the file has not been corrupted then use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6177   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called.
User Response: Details of the error have been stored at &3. A synopsis is given in the data section below. Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6179   Load of dynamic module failed.
Explanation: MQSeries is unable to load dynamic module &3 because &4.
User Response:

AMQ6180   Default conversion not supported.
Explanation: MQSeries is unable to convert string data tagged in CCSID &1 to data in CCSID &2.
User Response: Check the default CCSIDs specified in the ccsid.tbl file and make sure that conversion is supported between these CCSIDs.

AMQ6182   Error found in line &1 of ccsid.tbl
Explanation: Line &1 contains and error. The content of the line is &3. Processing continues but the line in error is ignored.
User Response: Correct the line and rerun the program or command giving this message.

AMQ6183   An internal MQSeries error has occurred.
Explanation: An error has been detected, and the MQSeries error recording routine has been called. The failing process is process &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6184   An internal MQSeries error has occurred on queue manager &3.
Explanation: An error has been detected, and the MQSeries error recording routine has been called. The failing process is process &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6187   User is not authorized for RestrictedMode queue manager.
Explanation: All users must be in the RestrictedMode application_group.
User Response: None.

AMQ6188   The system could not dynamically load the library &3 due to a problem with the dll. The errno was &1. The queue manager will continue without this module.
Explanation: This message applies to Windows NT and Windows 2000 systems only. The dynamically loadable file &3 failed to load correctly due to a problem with the dll.
User Response: Check that the dll is in the correct place with the correct file permissions etc. and has not been corrupted.

AMQ6188   The system could not dynamically load the shared library '&3' due to a problem with the library. The errno was &1 and the error message was '&4'. The queue manager will continue without this library.
Explanation: This message applies to AIX, HP-UX, and Linux systems. The shared library '&3' failed to load correctly due to a problem with the library.
User Response: Check that the library exists and is in the correct place with the correct file permissions etc. and has not been corrupted.

AMQ6188   The system could not dynamically load the shared library '&3' due to a problem with the library. The error message was '&4'. The queue manager will continue without this library.
Explanation: This message applies to Sun Solaris systems. The shared library &3 failed to load correctly due to a problem with the library.
User Response: Check that the library exists and is in the correct place with the correct file permissions etc. and has not been corrupted.

AMQ6189   Unable to open error log &3.
Explanation: The error log file &3 cannot be opened. The DosOpen error number is &1. Messages cannot be logged. They may be displayed on your console, following this message.
User Response: Check that the path to the error log exists, and that, if the file exists, it is writable. If the file exists, save its content in another file and delete it. The error log process creates the file if the path exists and the file is missing.

AMQ6190   Program '&3' not found.
Explanation: The program '&3' cannot be found.
User Response: Check that the program specified is available on your system. If the program name is not fully qualified, ensure that the PATH environment variable includes the directory where the program is located.

AMQ6191   Program '&3' failed to start, return code &1.
Explanation: The program '&3' was invoked, but failed to start. The failure reason code is &1.
User Response: Check that the program specified is available on your system, and that sufficient system resources are available. Where applicable, verify that the user is authorized to run the program.

AMQ6193   The registry entry &3 was not found.
Explanation: MQSeries for Windows NT and Windows 2000 sets the registry entry &3 when the product is installed, but the entry is now missing.
User Response: If the registry has been edited, restore the previous version. If the product is newly installed, check whether the installation was successful, and reinstall the product if necessary.

AMQ6196   An error has occurred whilst processing a temporary INI file &3
Explanation: An error has occurred when creating a backup of an INI file. The backup file &4 already exists
User Response: You may have created a backup of the INI file with the name &4, or an earlier operation may have failed. Move or delete the file &4 and reattempt the operation. If you have not changed the INI file, use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ6207   Invalid IPCCBaseAddress parameter (&1), default used (8).
Explanation: IPCCBaseAddress must be one of the following values 4, 5, 8, 9, 10, 11, or 12.
User Response: Stop the Queue Manager. Correct the value in the QueueManager stanza of the mqs.ini file and restart the Queue Manager.

AMQ6209   An unexpected asynchronous signal (&1) has been received and ignored.
Explanation: Process &2 received an unexpected asynchronous signal and ignored it. This has not caused an error, but the source of the signal should be determined, as it is likely that the signal has been generated externally to MQSeries. Use the UNIX system command kill, with option -l, to determine the symbolic name for signal &1.
User Response: Determine the source of the signal and prevent it from occurring again.

AMQ6708   A disk full condition was encountered when formatting a new log file in location &3.
Explanation: The queue manager attempted to format a new log file in directory &3. The drive or file system containing this directory did not have sufficient free space to contain the new log file.
User Response: Increase the amount of space available for log files and retry the request.

AMQ6709   The log for the Queue manager is full.
Explanation: This message is issued when an attempt to write a log record is rejected because the log is full. The queue manager will attempt to resolve the problem.
User Response: This situation may be encountered during a period of unusually high message traffic. However, if you persistently fill the log, you may have to consider enlarging the size of the log. You can either increase the number of log files by changing the values in the queue manager configuration file. You will then have to stop and restart the queue manager. Alternatively, if you need to make the log files themselves bigger, you will have to delete and recreate the queue manager.

AMQ6710   Queue manager unable to access directory &3.
Explanation: The queue manager was unable to access directory &3 for the log. This could be because the directory does not exist, or because the queue manager does not have sufficient authority.
User Response: Ensure that the directory exists and that the queue manager has authority to read and write to it. Ensure that the LogPath attribute in the queue manager's configuration file matches the intended log path.

AMQ6767   Log file &3 could not be opened for use.
Explanation: Log file &3 could not be opened for use. Possible reasons include the file being missing, the queue manager being denied permission to open the file or the contents of the file being incorrect.
User Response: If the log file was required to start the queue manager, ensure that the log file exists and that the queue manager is able to read from and write to it. If the log file was required to recreate an object from its media image and you do not have a copy of the required log file, delete the object instead of recreating it.

AMQ6774   Log file &3 did not contain the requested log record.
Explanation: Log file &3 does not contain the log record whose LSN is &4. This is because the log file numbers have wrapped and the log file name &3 has been reused by a newer file. Once a log file name has been reused, it is not possible to access the data in the previous versions of the file to use this name. The operation which requested this log record cannot be completed.
User Response: None

AMQ6782   The log file numbers have wrapped.
Explanation: Each log file formatted is assigned a number which makes up part of its file name. The numbers are allocated sequentially and consist of seven digits giving a maximum of 10 million different log file names. Once all available numbers have been allocated, the queue manager again starts allocating numbers starting from zero. Once a file number has been re-allocated, you can no longer access data in the previous log files allocated the same number. The file numbers wrapped at log sequence number &3.
User Response: You should periodically take media

images of all MQSeries objects. You must ensure that
media images of all objects which you may need to
recreate do not span more than 10 million log files.

Chapter 6. AMQ7000-AMQ7999 MQSeries product messages

> **Reading variables in the message**
> Some messages display text or numbers that vary according to the circumstances giving rise to the message; these are known as *message variables*. The message variables are indicated in this book by the use of the '&' symbol and a number: &1, &2, and so on.
>
> In some cases a message may have variables in the Explanation or User action. Find the values of the message variables by looking in the error log. The complete message, including the Explanation and the User action, is recorded there.

AMQ7001   The location specified for the creation of the queue manager is not valid.
Explanation: The directory under which the queue managers are to be created is not valid. It may not exist, or there may be a problem with authorization.
User Response: The location is specified in the configuration data. Correct the configuration data and submit the request again.

AMQ7002   An error occurred manipulating a file.
Explanation: An internal error occurred while trying to create or delete a queue manager file.

In the case of a failure to delete a file, a common reason for this error is that a non MQSeries process, for example, the Windows explorer or a virus checker, is accessing the file. In the case where the object that cannot be deleted is a directory, a non-MQSeries process may be accessing a file within the directory or one of its subdirectories.

It is also possible that the error was caused by a disk having insufficient space, or by problems with authorization to the underlying filesystem.
User Response: Identify the file that caused the error, using problem determination techniques. For example, check if there are any FFST files which may identify the queue manager file causing the error.

This error can also be caused if users have created, renamed, or deleted that file. Correct the error in the filesystem and submit the request again.

AMQ7005   The queue manager is running.
Explanation: You tried to perform an action that requires the queue manager stopped, however, it is currently running. You probably tried to delete or start a queue manager that is currently running.
User Response: If the queue manager should be stopped, stop the queue manager and submit the failed command again.

AMQ7006   Missing attribute &5 from configuration data.
Explanation: The &4 stanza in the configuration data is missing the required &5 attribute.
User Response: Check the contents of the configuration data and retry the operation.

AMQ7008   The queue manager already exists.
Explanation: You tried to create a queue manager that already exists.
User Response: If you specified the wrong queue manager name, correct the name and submit the request again.

AMQ7010   The queue manager does not exist.
Explanation: You tried to perform an action against a queue manager that does not exist. You may have specified the wrong queue manager name.
User Response: If you specified the wrong name, correct it and submit the command again. If the queue manager should exist, create it, and then submit the command again.

AMQ7011   The queue manager files have not been completely deleted.
Explanation: While deleting the queue manager, an error occurred deleting a file or directory. The queue manager may not have been completely deleted.
User Response: Follow problem determination procedures to identify the file or directory and to complete deletion of the queue manager.

AMQ7012   The specified trigger interval is not valid.
Explanation: You specified a value for the trigger interval that is not valid. The value must be not less than zero and not greater than 999 999 999.
User Response: Correct the value and resubmit the request.

45

AMQ7013   There is an error in the name of the specified dead-letter queue.
Explanation: You specified a name for the dead-letter queue that is not valid.
User Response: Correct the name and resubmit the request.

AMQ7014   There is an error in the name of the specified default transmission queue.
Explanation: You specified a name for the default transmission queue that is not valid.
User Response: Correct the name and submit the command again.

AMQ7015   There is an error in the maximum number of open object handles specified.
Explanation: You specified a value for the maximum number of open object handles to be allowed that is not valid. The value must be not less than zero and not greater than 999 999 999.
User Response: Correct the value and submit the command again.

AMQ7016   There is an error in the maximum number of uncommitted messages specified.
Explanation: You specified a value for the maximum number of uncommitted messages to be allowed that is not valid. The value must be not less than 1 and not greater than 999 999 999.
User Response: Correct the value and submit the command again.

AMQ7017   Log not available.
Explanation: The queue manager was unable to use the log. This could be due to a log file being missing or damaged, or the log path to the queue manager being inaccessible.
User Response: Ensure that the LogPath attribute in the queue manager configuration file is correct. If a log file is missing or otherwise unusable, restore a backup copy of the file, or the entire queue manager.

AMQ7018   The queue manager operation cannot be completed.
Explanation: An attempt has been made to perform an operation on a queue manager. Resources required to perform the operation are not available.
User Response:

AMQ7019   An error occurred while creating the directory structure for the new queue manager.
Explanation: During creation of the queue manager an error occurred while trying to create a file or directory.
User Response: Identify why the queue manager files

cannot be created. It is probable that there is insufficient space on the specified disk, or that there is a problem with access control. Correct the problem and submit the command again.

AMQ7020   The operation was carried out, but one or more transactions remain in-doubt.
Explanation: The queue manager tried to resolve all internally coordinated transactions which are in-doubt. In-doubt transactions still remain after the queue manager has attempted to deliver the outcome of these transactions to the resource managers concerned. Transactions remain in-doubt when the queue manager cannot deliver the outcome of the transaction to each of the participating resource managers. For example, a resource manager may not be available at this time.
User Response: Use the DSPMQTRN command to display the remaining in-doubt transactions.

AMQ7021   An error occurred while deleting the directory structure for the queue manager.
Explanation: While deleting the queue manager, an error occurred deleting a file or directory. The queue manager may not have been completely deleted.
User Response: Follow problem determination procedures to identify the file or directory and to complete deletion of the queue manager.

AMQ7022   The resource manager identification number is not recognized.
Explanation: The identification number of the resource manager you supplied was not recognized.
User Response: Ensure that you entered a valid resource manager identification number. Use the DSPMQTRN command to display a list of resource managers and their identification numbers.

AMQ7023   The resource manager was in an invalid state.
Explanation: The resource manager, the identification number of which you supplied, was in an invalid state.
User Response: Ensure that you entered the correct resource manager identification number. Use the DSPMQTRN command to display a list of resource managers and their identification numbers. A resource manager is in an invalid state, if it is still available to resolve the transaction, use the -a optional flag to resolve this and all other internally coordinated in-doubt transactions.

AMQ7024   Arguments supplied to a command are not valid.
Explanation: You supplied arguments to a command that it could not interpret. It is probable that you specified a flag not accepted by the command, or that you included extra flags.
User Response: Correct the command and submit it

again. Additional information on the arguments causing the error may be found in the error logs for the queue, or queue manager, referenced in the command.

AMQ7025   **Error in the descriptive text argument (-c parameter) of the crtmqm command.**
Explanation: The descriptive text you supplied to the crtmqm command was in error.
User Response: Correct the descriptive text argument and submit the command again.

AMQ7026   **A principal or group name was invalid.**
Explanation: You specified the name of a principal or group which does not exist.
User Response: Correct the name and resubmit the request.

AMQ7027   **Argument &3 supplied to command &4 is invalid.**
Explanation: The argument &3 was supplied to the command &4 which could not be interpreted. This argument is either not accepted by the command, or an extra flag has been included.
User Response: Correct the command and submit it again.

AMQ7028   **The queue manager is not available for use.**
Explanation: You have requested an action that requires the queue manager running, however, the queue manager is not currently running.
User Response: Start the required queue manager and submit the command again.

AMQ7030   **Quiesce request accepted. The queue manager will stop when all outstanding work is complete.**
Explanation: You have requested that the queue manager end when there is no more work for it. In the meantime, it will refuse new applications that attempt to start, although it allows those already running to complete their work.
User Response: None.

AMQ7031   **The queue manager is stopping.**
Explanation: You issued a command that requires the queue manager running, however, it is currently in the process of stopping. The command cannot be run.
User Response: None

AMQ7041   **Object already exists.**
Explanation: A Define Object operation was performed, but the name selected for the object is already in use by an object that is unknown to MQSeries. The object name selected by MQSeries was &3, in directory &4, of object type &5.
User Response: Remove the conflicting object from

the MQSeries system, then try the operation again.

AMQ7042   **Media image not available for object &3 of type &4.**
Explanation: The media image for object &3, type &4, is not available for media recovery. A log file containing part of the media image cannot be accessed.
User Response: A previous message indicates which log file could not be accessed. Restore a copy of the log file and all subsequent log files from backup. If this is not possible, you must delete the object instead.

AMQ7044   **Media recovery not allowed.**
Explanation: Media recovery is not possible on a queue manager using a circular log. Damaged objects must be deleted on such a queue manager.
User Response: None.

AMQ7047   **An unexpected error was encountered by a command.**
Explanation: An internal error occurred during the processing of a command.
User Response: Follow problem determination procedures to identify the cause of the error.

AMQ7048   **The queue manager name is either not valid or not known**
Explanation: Either the specified queue manager name does not conform to the rules required by MQSeries or the queue manager does not exist. The rules for naming MQSeries objects are detailed in the MQSeries Command Reference.
User Response: Correct the name and submit the command again.

AMQ7053   **The transaction has been committed.**
Explanation: The prepared transaction has been committed.
User Response: None.

AMQ7054   **The transaction has been backed out.**
Explanation: The prepared transaction has been backed out.
User Response: None.

AMQ7055   **The transaction number is not recognized.**
Explanation: The number of the transaction you supplied was not recognized as belonging to an in-doubt transaction.
User Response: Ensure that you entered a valid transaction number. It is possible that the transaction number you entered corresponds to a transaction which was committed or backed out before you issued the command to resolve it.

AMQ7056   Transaction number &1,&2.
Explanation: This message is used to report the
number of an in-doubt transaction.
User Response: None.

**AMQ7059   An error occurred when reading the
configuration data.**
Explanation: An error has occurred when reading the
configuration data.
User Response: If you have changed the configuration
data, check and correct the change. If you have not
changed the configuration data, use the standard
facilities supplied with your system to record the
problem identifier, and to save the generated output
files. Contact your IBM support center. Do not discard
these files until the problem has been resolved.

**AMQ7060   An error has occurred locking the
configuration data.**
Explanation: An error has occurred locking the
configuration data.
User Response: If you have changed the configuration
data permissions, check and correct the change. If you
have not changed the configuration data, use the
standard facilities supplied with your system to record
the problem identifier, and to save the generated
output files. Contact your IBM support center. Do not
discard these files until the problem has been resolved.

**AMQ7061   An expected stanza in the configuration
data is missing or contains errors.**
Explanation: An expected stanza is missing from the
configuration data or the stanza contains errors.
User Response: If you have changed the configuration
data, check and correct the change.

AMQ7062   Unable to access the configuration data.
Explanation: Access to the configuration data is
denied.
User Response: If you have changed the configuration
data permissions, check and correct the change.

AMQ7063   Configuration data is missing.
Explanation: The configuration data for MQSeries is
missing.
User Response: If you have changed the configuration
data, recover the previous configuration data and retry
the operation.

AMQ7064   Log path not valid or inaccessible.
Explanation: The supplied log path could not be used
by the queue manager. Possible reasons for this include
the path not existing, the queue manager not being able
to write to the path, or the path residing on a remote
device.
User Response: Ensure that the log path exists and
that the queue manager has authority to read and write

to it. If the queue manager already exists, ensure that
the LogPath attribute in the queue manager's
configuration file matches the intended log path.

AMQ7065   Insufficient space on disk.
Explanation: The operation cannot be completed due
to shortage of disk space.
User Response: Either make more disk space
available, or reduce the disk requirements of the
command you issued.

AMQ7066   There are no prepared transactions.
Explanation: There are no prepared transactions to be
resolved.
User Response: None.

**AMQ7068   Authority file contains an authority
stanza that is not valid.**
Explanation:
User Response: Correct the contents of the incorrect
authorization file by editing it.

**AMQ7069   The queue manager was created
successfully, but cannot be made the
default.**
Explanation: The queue manager was defined to be
the default queue manager for the machine when it
was created. However, although the queue manager
has been created, an error occurred trying to make it
the default. There may not be a default queue manager
defined for the machine at present.
User Response: There is probably a problem with the
machine-wide ini file. Verify the existence of the file, its
access permissions, and its contents. If its backup file
exists, reconcile the contents of the two files and then
delete the backup. Finally, either update the
machine-wide ini file by hand to specify the desired
default queue manager, or delete and recreate the
queue manager.

**AMQ7069   The queue manager was created
successfully, but cannot be made the
default.**
Explanation: The queue manager was defined to be
the default queue manager for the machine when it
was created. However, although the queue manager
has been created, an error occurred trying to make it
the default. There may not be a default queue manager
defined for the machine at present.
User Response: There is probably a problem with the
configuration data. Update the configuration data to
specify the desired default queue manager, or delete
and recreate the queue manager.

AMQ7072   Stanza not valid. Refer to the error log for more information.
Explanation: A stanza that is not valid was found. Refer to the error log for more information.
User Response: Correct the error and retry the operation.

AMQ7073   Log size not valid.
Explanation: Either the number of log files or the size of the log files was outside the accepted values.
User Response: Make sure that the log parameters you enter lie within the valid range.

AMQ7074   Unknown stanza key &4 at &3 in the configuration data.
Explanation: Key &3 contained a stanza called &4. This stanza is not recognized.
User Response: Check the contents of the configuration data and retry the operation.

AMQ7075   Unknown attribute in ini file.
Explanation: Line &1 of the configuration file &3 contained an attribute called &4 that is not valid. This attribute is not recognized in this context.
User Response: Check the contents of the file and retry the operation.

AMQ7075   Unknown attribute &4 at &3 in the configuration data.
Explanation: Key &3 in the configuration data contained an attribute called &4 that is not valid. This attribute is not recognized in this context.
User Response: Check the contents of the configuration data and retry the operation.

AMQ7076   Invalid value for attribute in ini file.
Explanation: Line &1 of the configuration file &3 contained value &5 that is not valid for the attribute &4.
User Response: Check the contents of the file and retry the operation.

AMQ7076   Value &5 not valid for attribute &4 at &3 in the configuration data.
Explanation: Key &3 in the configuration data contained value &5 that is not valid for the attribute &4.
User Response: Check the contents of the configuration data and retry the operation.

AMQ7077   You are not authorized to perform the requested operation.
Explanation: You tried to issue a command for the queue manager. You are not authorized to perform the command.
User Response: Contact your system administrator to perform the command for you. Alternatively, request

authority to perform the command from your system administrator.

AMQ7080   No objects processed.
Explanation: No objects were processed, either because no objects matched the criteria given, or because the objects found did not require processing.
User Response: None.

AMQ7081   Object &3, type &4 recreated.
Explanation: The object &3, type &4 was recreated from its media image.
User Response: None.

AMQ7082   Object &3, type &4 is not damaged.
Explanation: Object &3, type &4 cannot be recreated since it is not damaged.
User Response: None

AMQ7083   A resource problem was encountered by a command.
Explanation: The command failed due to a resource problem. Possible causes include the log being full or the command running out of memory.
User Response: Look at the previous messages to diagnose the problem. Rectify the problem and retry the operation.

AMQ7084   Object &3, type &4 damaged.
Explanation: The object &3, type &4 was damaged. The object must be deleted or, if the queue manager supports media recovery, recreated from its media image.
User Response: Delete the object or recreate it from its media image.

AMQ7085   Object &3, type &4 not found.
Explanation: Object &3, type &4 cannot be found.
User Response: None.

AMQ7086   Media image for object &3, type &4 recorded.
Explanation: The media image for object &3, type &4, defined in Queue Manager &5, has been recorded.
User Response: None.

AMQ7087   Object &3, type &4 is a temporary object
Explanation: Object &3, type &4 is a temporary object. Media recovery operations are not permitted on temporary objects.
User Response: None.

AMQ7088 Object &3, type &4 in use.
Explanation: Object &3, type &4 is in use. Either an application has it open or, if it is a local queue, there are uncommitted messages on it.
User Response: Ensure that the object is not opened by any applications, and that there are no uncommitted messages on the object, if it is a local queue. Then, retry the operation.

AMQ7089 Media recovery already in progress.
Explanation: Another media recovery operation is already in progress. Only one media recovery operation is permitted at a time.
User Response: Wait for the existing media recovery operation to complete and retry the operation.

AMQ7090 The queue manager CCSID is not valid.
Explanation: The CCSID to be used by the QMGR is not valid, because: 1) It is a DBCS CCSID. 2) The CCSID encoding is not ASCII or ASCII related. EBCDIC or UCS2 encodings are not valid on this machine. 3) The CCSID encoding is unknown.
User Response: Check the CCSID is valid for the machine on which you are working.

AMQ7091 You are performing authorization for the queue manager, but you specified an object name.
Explanation: Modification of authorizations for a queue manager can be performed only from that queue manager. You must not specify an object name.
User Response: Correct the command and submit it again.

AMQ7092 An object name is required but you did not specify one.
Explanation: The command needs the name of an object, but you did not specify one.
User Response: Correct the command and submit it again.

AMQ7093 An object type is required but you did not specify one.
Explanation: The command needs the type of the object, but you did not specify one.
User Response: Correct the command and submit it again.

AMQ7094 You specified an object type that is not valid, or more than one object type.
Explanation: Either the type of object you specified was not valid, or you specified multiple object types on a command which supports only one.
User Response: Correct the command and submit it again.

AMQ7095 An entity name is required but you did not specify one.
Explanation: The command needs one or more entity names, but you did not specify any. Entities can be principals or groups.
User Response: Correct the command and submit it again.

AMQ7096 An authorization specification is required but you did not provide one.
Explanation: The command sets the authorizations on MQSeries objects. However you did not specify which authorizations are to be set.
User Response: Correct the command and submit it again.

AMQ7097 You gave an authorization specification that is not valid.
Explanation: The authorization specification you provided to the command contained one or more items that could not be interpreted.
User Response: Correct the command and submit it again.

AMQ7098 The command accepts only one entity name. You specified more than one.
Explanation: The command can accept only one principal or group name. You specified more than one.
User Response: Correct the command and submit it again.

AMQ7099 Entity &3 has the following authorizations for object &4:
Explanation: Informational message. The list of authorizations follows.
User Response: None.

AMQ7099 Entity &3 has the following authorizations for object &4:
Explanation: Informational message. The list of authorizations follows.
User Response: None.

AMQ7104 Resource manager &1 has prepared.
Explanation: This message reports the state of a resource manager with respect to an in-doubt transaction.
User Response: None.

AMQ7105 Resource manager &1 has committed.
Explanation: This message reports the state of a resource manager with respect to an in-doubt transaction.
User Response: None.

AMQ7106   Resource manager &1 has rolled back.
Explanation: This message reports the state of a resource manager with respect to an in-doubt transaction.
User Response: None.

AMQ7107   Resource manager &1 is &3.
Explanation: This message reports the identification number and name of a resource manager.
User Response: None.

AMQ7108   Any in-doubt transactions have been resolved.
Explanation: All, if there were any, of the internally coordinated transactions which were in-doubt, have now been resolved. This message reports successful completion of the RSVMQTRN command when the -a option is used.
User Response: None.

AMQ7109   A decision on behalf of the unavailable resource manager has been delivered.
Explanation: A decision for an internally coordinated transaction which was in-doubt, has now been delivered on behalf of the unavailable resource manager. This message reports successful completion of the RSVMQTRN command when the -r option is used.
User Response: None.

AMQ7110   Media image for the syncfile recorded.
Explanation: The media image for the syncfile has been recorded.
User Response: None.

AMQ7111   Resource manager &1 has participated.
Explanation: This message reports the state of a resource manager with respect to an in-doubt transaction.
User Response: None.

AMQ7112   Transaction number &1,&2 has encountered an error.
Explanation: This message is used to report the number of an in-doubt transaction which has encountered an error with one or more resource managers.
User Response: Refer to the queue manager error log for more information about which resource managers are in error. Ensure that the resource managers that were in error, are working correctly, restart the queue manager. If the problem persists, use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ7153   A license could not be installed for this copy of MQSeries.
Explanation: A Production or Trial Period license could not be installed for this copy of MQSeries. This is because the 'nodelock' file in the 'qmgrs/@SYSTEM' directory could not be created or updated.
User Response: Check the ownership and permissions of the 'qmgrs/@SYSTEM' directory.

AMQ7159   A FASTPATH application has ended unexpectedly.
Explanation: A FASTPATH application has ended in a way that did not allow the queue manager to clean up the resources owned by that application. Any resources held by the application can only be released by stopping and restarting the queue manager.
User Response: Investigate why the application ended unexpectedly. Avoid ending FASTPATH applications in a way which prevents MQSeries from releasing resources held by the application.

AMQ7196   By installing this product, you accept the terms of the International Program License Agreement and the License Information supplied with the product.

AMQ7206   Group name has been truncated.
Explanation: MQSeries only supports group names up to 12 characters long. The operating system is attempting to return a group longer than this.
User Response: Reduce the group name to 12 characters or less.

AMQ7207   User ID longer than 12 characters.
Explanation: MQSeries only supports user names up to 12 characters long. This operation is being attempted from a user name longer than this.
User Response: Reduce the user name to 12 characters or less.

AMQ7208   The queue manager failed to pass a PCF message to another queue manager.
Explanation: The queue manager attempted to put a PCF message to &3 to start the channel &4 to cluster queue manager &5. The put failed with reason &1. When the queue manager resolves a cluster queue to a remote cluster queue manager, the message is put to the SYSTEM.CLUS.TRANSMIT.QUEUE. If the channel to the remote cluster queue manager is not running, the queue manager attempts to start the channel by sending a PCF message to &3.
User Response: Resolve the problem with &3 and if necessary start the channel manually.

AMQ7209 The queue manager attempted to open SYSTEM.CHANNEL.INITQ which failed with reason &3

Explanation: When the queue manager resolves a cluster queue to a remote cluster queue manager, the message is put to the SYSTEM.CLUS.TRANSMIT.QUEUE. If the channel to the remote cluster queue manager is not running, the queue manager attempts to start the channel by sending a PCF message to the SYSTEM.CHANNEL.INITQ

User Response: Resolve the problem with the SYSTEM.CHANNEL.INITQ and if necessary start the channels manually.

AMQ7210 The Cluster Workload exit module could not be loaded.

Explanation: The Cluster Workload exit module '&3' could not be loaded for reason &4.

User Response: Correct the problem with the Cluster Workload exit module '&3'

AMQ7211 The Queue Manager is still waiting for a reply from the Cluster Workload Exit server process.

Explanation: The Queue Manager is configured to run the Cluster Workload Exit in SAFE mode. This means that the Cluster Workload Exit is run by a server process (amqzlw0). The Queue Manager has been waiting &1 seconds for this server process to reply to a request to run the Cluster Workload Exit. It is possible that the exit is hung or is looping.

User Response: End the Queue Manager, resolve the problem with the Cluster Workload Exit and restart the Queue Manager

AMQ7212 The address of the Cluster exit function could not be found.

Explanation: The address of the Cluster exit function '&4' could not be found in module '&3' for reason &1 &5.

User Response: Correct the problem with the Cluster exit function '&4' in the module '&3'

AMQ7305 Trigger message could not be put on an initiation queue.

Explanation: The attempt to put a trigger message on queue &4 on queue manager &5 failed with reason code &1. The message will be put on the dead-letter queue.

User Response: Ensure that the initiation queue is available, and operational.

AMQ7306 The dead-letter queue must be a local queue.

Explanation: An undelivered message has not been put on the dead-letter queue &4 on queue manager &5, because the queue is not a local queue. The message will be discarded.

User Response: Inform your system administrator.

AMQ7307 A message could not be put on the dead-letter queue.

Explanation: The attempt to put a message on the dead-letter queue &4 on queue manager &5 failed with reason code &1. The message will be discarded.

User Response: Ensure that the dead-letter queue is available and operational.

AMQ7308 Trigger condition &1 was not satisfied.

Explanation: At least one of the conditions required for generating a trigger message was not satisfied, so a trigger message was not generated. If you were expecting a trigger message, consult the MQSeries Application Programming Guide for a list of the conditions required. (Note that arranging for condition &1 to be satisfied might not be sufficient because the conditions are checked in an arbitrary order, and checking stops when the first unsatisfied condition is discovered.)

User Response: If a trigger message is required, ensure that all the conditions for generating one are satisfied.

AMQ7310 Report message could not be put on a reply-to queue.

Explanation: The attempt to put a report message on queue &4 on queue manager &5 failed with reason code &1. The message will be put on the dead-letter queue.

User Response: Ensure that the reply-to queue is available and operational.

AMQ7463 The log for queue manager &3 is full.

Explanation: This message is issued when an attempt to write a log record is rejected because the log is full. The queue manager will attempt to resolve the problem.

User Response: This situation may be encountered during a period of unusually high message traffic. However, if you persistently fill the log, you may have to consider enlarging the size of the log. You can either increase the number of log files by changing the values in the queue manager configuration file. You will then have to stop and restart the queue manager. Alternatively, if you need to make the log files themselves bigger, you will have to delete and recreate the queue manager.

AMQ7464   The log for queue manager &3 is no longer full.

Explanation: This message is issued when a log was previously full, but an attempt to write a log record has now been accepted. The log full situation has been resolved.

User Response: None

AMQ7465   The log for queue manager &3 is full.

Explanation: An attempt to resolve a log full situation has failed. This is due to the presence of a long-running transaction.

User Response: Try to ensure that the duration of your transactions is not excessive. Commit or roll back any old transactions to release log space for further log records.

AMQ7466   There is a problem with the size of the logfile.

Explanation: The log for queue manager &3 is too small to support the current data rate. This message is issued when the monitoring tasks maintaining the log cannot keep up with the current rate of data being written.

User Response: The number of primary log files configured should be increased to prevent possible log full situations.

AMQ7467   The oldest log file required to start queue manager &3 is &4.

Explanation: The log file &4 contains the oldest log record required to restart the queue manager. Log records older than this may be required for media recovery.

User Response: You can move log files older than &4 to an archive medium to release space in the log directory. If you move any of the log files required to recreate objects from their media images, you will have to restore them to recreate the objects.

AMQ7468   The oldest log file required to perform media recovery of queue manager &3 is &4.

Explanation: The log file &4 contains the oldest log record required to recreate any of the objects from their media images. Any log files prior to this will not be accessed by media recovery operations.

User Response: You can move log files older than &4 to an archive medium to release space in the log directory.

AMQ7469   Transactions rolled back to release log space.

Explanation: The log space for the queue manager is becoming full. One or more long-running transactions have been rolled back to release log space so that the queue manager can continue to process requests.

User Response: Try to ensure that the duration of

your transactions is not excessive. Consider increasing the size of the log to allow transactions to last longer before the log starts to become full.

AMQ7472   Object &3, type &4 damaged.

Explanation: Object &3, type &4 has been marked as damaged. This indicates that the queue manager was either unable to access the object in the file system, or that some kind of inconsistency with the data in the object was detected.

User Response: If a damaged object is detected, the action performed depends on whether the queue manager supports media recovery and when the damage was detected. If the queue manager does not support media recovery, you must delete the object as no recovery is possible. If the queue manager does support media recovery and the damage is detected during the processing performed when the queue manager is being started, the queue manager will automatically initiate media recovery of the object. If the queue manager supports media recovery and the damage is detected once the queue manager has started, it may be recovered from a media image using the rcrmqmobj command or it may be deleted.

AMQ7601   Duplicate XA resource manager is not valid.

Explanation: Line &1 of the configuration file &3 contained a duplicate XA resource manager '&5'. This is not valid for attribute &4. Each XA resource manager must be given a unique name.

User Response: Check the contents of the file and retry the operation.

AMQ7601   Duplicate XA resource manager '&5' not valid for attribute &4 at &3 in the configuration data.

Explanation: Key &3 in the configuration data contained a duplicate XA resource manager '&5'. This is not valid for attribute &4. Each XA resource manager must be given a unique name.

User Response: Check the contents of the configuration data and retry the operation.

AMQ7603   MQSeries has been configured with resource manager '&3' that is not valid.

Explanation: The XA switch file &4 for resource manager &3 indicates that an attempt has been made to configure another queue manager as an external resource manager. This is not allowed, so the queue manager will terminate.

User Response: Remove the offending XAResourceManager stanza from the configuration data and restart the queue manager.

AMQ7604 **The XA resource manager '&3' was not available when called for &4. The queue manager is continuing without this resource manager.**

Explanation: The XA resource manager '&3' has indicated that it is not available, by returning XAER_RMERR on an xa_open request or XAER_RMFAIL when called for something else. Normally this indicates that the resource manager has been shut down. In this case the resource manager cannot participate in any new transactions. Any in-flight transactions in which it was involved will be backed out, and any transactions in which it is in-doubt will only be resolved when contact with the resource manager is re-established. A further message will be issued when the queue manager has been able to do this. If the problem occurred on an xa_open request, and the resource manager should be available, then there may be a configuration problem.

User Response: Try to establish the reason why the resource manager is unavailable. It may be that an invalid XAOpenString has been defined for the resource manager in the 'qm.ini' configuration file. If this is the case, stop and then restart the queue manager so that any change will be picked up. Alternatively, the queue manager may be reaching a resource constraint with this resource manager. For example, the resource manager may not be able to accommodate all of the queue manager processes being connected at one time, you may need to alter one of its tuning parameters.

AMQ7605 **The XA resource manager &3 has returned an unexpected return code &1, when called for &4.**

Explanation: MQSeries received an unexpected return code when calling XA resource manager &3 at its &4 entry point. This indicates an internal error, either within MQSeries or the resource manager.

User Response: Try to determine the source of the error. A trace of the failure could be used to look at the XA flows between MQSeries and the resource manager. MQSeries has allocated an RMId of &2 to this resource manager. This will be useful when isolating the flows associated with the resource manager concerned. If the error occurs on an xa_commit or xa_rollback request, the queue manager will not attempt to redeliver the commit or rollback instruction for this transaction, until after the queue manager has been restarted. The transaction indoubt is identified by the following XID of X'&5'. If you think that the error lies within the queue manager, contact your IBM support center. Do not discard any information describing the problem until after the problem has been resolved.

AMQ7606 **A transaction has been committed but one or more resource managers have backed out.**

Explanation: MQSeries was processing the commit operation for a transaction involving external resource managers. One or more of these resource managers failed to obey the commit request and instead rolled back their updates. The outcome of the transaction is now mixed and the resources owned by these resource managers may now be out of synchronization. MQSeries will issue further messages to indicate which resource managers failed to commit their updates.

User Response: The transaction with the mixed outcome is identified by the following XID of X'&3'. The messages which identify the failing resource managers will also contain this same XID. If the transaction has completed it won't be displayed by the dspmqtrn command and all other transaction participants will have committed their updates. If the transaction is displayed by the dspmqtrn command then there are some participants still in prepared state. In order to preserve data integrity you will need to perform recovery steps local to the failing resource managers.

AMQ7607 **A transaction has been rolled back but one or more resource managers have committed.**

Explanation: MQSeries was rolling back a transaction involving external resource managers. One or more of these resource managers failed to obey the rollback request and instead committed their updates. The outcome of the transaction is now mixed and the resources owned by these resource managers may now be out of synchronization. MQSeries will issue further messages to indicate which resource managers failed to roll back their updates.

User Response: The transaction with the mixed outcome is identified by the following XID of X'&3'. The messages which identify the failing resource managers will also contain this same XID. If the transaction has completed it won't be displayed by the dspmqtrn command and all other transaction participants will have rolled back their updates. If the transaction is displayed by the dspmqtrn command then there are some participants still in prepared state. In order to preserve data integrity you will need to perform recovery steps local to the failing resource managers.

AMQ7608 **XA resource manager returned a heuristic return code.**

Explanation: This message is associated with an earlier AMQ7606 message reporting a mixed transaction outcome. It identifies one of the resource managers ('&4') that failed to commit its updates. The transaction associated with this failure is identified by the following XID of X'&3'.

User Response: Use the return code &1 returned by the resource manager to determine the effects of the

failure. The return code indicates that the resource manager made a heuristic decision about the outcome of the transaction which disagrees with the commit decision of the queue manager. In order to preserve data integrity you will need to perform recovery steps local to this resource manager.

AMQ7609   XA resource manager returned a heuristic return code.
Explanation: This message is associated with an earlier AMQ7607 message reporting a mixed transaction outcome. It identifies one of the resource managers ('&4') that failed to rollback its updates. The transaction associated with this failure is identified by the following XID of X'&3'.
User Response: Use the return code &1 returned by the resource manager to determine the effects of the failure. The return code indicates that the resource manager made a heuristic decision about the outcome of the transaction which disagrees with the rollback decision of the queue manager. In order to preserve data integrity you will need to perform recovery steps local to this resource manager.

AMQ7612   Switch call exception
Explanation: Exception number &1 occurred when calling resource manager switch &3.
User Response: Check the resource manager switch has not been corrupted.

AMQ7622   MQSeries could not load the XA switch load file for resource manager '&3'.
Explanation: An error has occurred loading XA switch file &4. If the error occurred during startup then the queue manager will terminate. At all other times the queue manager will continue without this resource manager meaning that it will no longer be able to participate in global transactions. The queue manager will also retry the load of the switch file at regular intervals so that the resource manager will be able to participate again should the load problem be resolved.
User Response: Look for a previous message outlining the reason for the load failure. Message AMQ6175 is issued if the load failed because of a system error.

If this is the case then follow the guidance given in message AMQ6175 to resolve the problem. In the absence of prior messages or FFST information related to this problem check that the name of the switch load file is correct and that it is present in a directory from which it can be dynamically loaded by the queue manager. The easiest method of doing this is to define the switch load file as a fully-qualified name. Note that if the queue manager is still running it will need to be restarted in order that any changes made to its configuration data can be picked up.

AMQ7623   MQSeries has not been configured with XA resource manager.
Explanation: The queue manager has noticed that XA resource manager '&3' was removed from the qm.ini file of the queue manager. However, it was logged as being involved in &1 transactions that are still in-doubt. The queue manager cannot resolve these transactions. The queue manager is continuing without this resource manager.
User Response: First check that the qm.ini configuration file of the queue manager concerned hasn't been mistakenly altered resulting in an 'XAResourceManager' stanza being removed, or the 'Name' of any the resource managers being changed. If the qm.ini file was changed by mistake then you will need to reinstate resource manager '&3' in the qm.ini file before stopping and then restarting the queue manager in order that the change will be picked up.

If you have intentionally removed a resource manager from the qm.ini file, consider the integrity implications of your action since the resource manager concerned may be in an in-doubt state. If you are sure that is not the case then you can use the 'rsvmqtrn' command to deliver an outcome on behalf of the resource manager in order that the queue manager can forget about the transactions concerned. If you cannot be sure that such an action will not cause an integrity problem then you should consider re-instating the resource manager in the qm.ini file so that the queue manager can contact the resource manager and automatically resolve the transactions concerned next time the queue manager is restarted.

AMQ7623   MQSeries has not been configured with XA resource manager '&3' which may be involved in in-doubt transactions. The queue manager is continuing without this resource manager.
Explanation: The queue manager has recognized that XA resource manager '&3' was removed from the registry entry of the queue manager. However, it was logged as being involved in &1 transactions that are still in-doubt. The queue manager cannot resolve these transactions.
User Response: Check that the configuration data entry of the queue manager concerned has not been altered by mistake, resulting in an 'XAResourceManager' stanza being removed, or the 'Name' of any the resource managers being changed.

If the configuration data entry was changed by mistake, you need to reinstate resource manager '&3' in the configuration data before stopping, and then restarting the queue manager to access the change.

If you have intentionally removed a resource manager from the configuration data, consider the integrity implications of your action because the resource manager concerned may be in an in-doubt state.

If you are sure that this is not the case, you can use the

| 'rsvmqtrn' command to instruct the resource manager
| to inform the queue manager that it can forget about
| the transactions concerned.

| If using the 'rsvmqtrn' command could result in an
| integrity problem, you should consider reinstating the
| resource manager in the configuration data, so that the
| queue manager can contact the resource manager and
| automatically resolve the transactions concerned next
| time the queue manager is restarted.

AMQ7624   **An exception occurred during an &4 call to XA resource manager '&3'.**

Explanation: An exception has been detected during a call to an XA resource manager. The queue manager will continue after assuming a return code of XAER_RMERR from the call.

User Response: An FFST should have been produced which documents the exception. Use this and any further FFSTs to try and determine the reason for the failure. A trace of the problem will be useful to identify the XA flows between the queue manager and the resource manager concerned. MQSeries has allocated an RMId of &1 to this resource manager. Use this to isolate the flows concerned. First contact the supplier of the resource manager for problem resolution. If however you think that the problem lies within the queue manager then contact your IBM support center. Do not discard any information describing the problem until after it has been resolved.

AMQ7625   **The XA resource manager '&3' has become available again.**

Explanation: MQSeries has managed to regain contact with a resource manager that had become unavailable. Any in-doubt transactions involving this resource manager will be resolved. The resource manager will now be able to participate in new transactions.

User Response: None.

AMQ7626   **XA resource manager initialization failure. Refer to the error log for more information.**

Explanation: The queue manager has failed to initialize one or more of the XA resource managers defined in the configuration data.

User Response: Correct the error and restart the queue manager.

AMQ7701   **DMPMQLOG command is starting.**

Explanation: You have started the DMPMQLOG command and it is processing your request.

User Response: None.

AMQ7702   **DMPMQLOG command has finished successfully.**

Explanation: The DMPMQLOG command has finished processing your request and no errors were detected.

User Response: None.

AMQ7703   **DMPMQLOG command has used option '&3' with an invalid value '&4'.**

Explanation: You started the DMPMQLOG command specifying an invalid option value. The '&4' value for option '&3' is either missing or of an incorrect format.

User Response: Refer to the command syntax, and then try the command again.

AMQ7704   **DMPMQLOG command has used an invalid option '&3'.**

Explanation: You started the DMPMQLOG command specifying an invalid option of '&3'.

User Response: Refer to the command syntax and then try the command again.

AMQ7705   **Usage: dmpmqlog [-b | -s StartLSN | -n ExtentNumber] [-e EndLSN] [-f LogFilePath] [-m QMgrName]**

Explanation:

User Response: None.

| **AMQ7706**   **DMPMQLOG command has used an incorrect queue manager name '&3' or path '&4'.**

| **Explanation:**   The DMPMQLOG command has used '&3' as the queue manager name and, if shown, '&4' as the directory path for '&3'. Either '&3' and/or '&4' is incorrect; if '&4' is not shown then it is '&3' which is incorrect.

| Possible reasons for the error include:
| • '&3' is not an existing queue manager name.
| • The entries for '&3' in the MQSeries system initialization (INI) file are incorrect.
| • '&4' is not a correct path for '&3'.

| If you started the command specifying option -m (queue manager name option) with a value then this value will have been used as the queue manager name, otherwise the default queue manager name will have been used.

| **User Response:**   Check that '&3' is an existing queue manager name. Check your MQSeries system's initialization (INI) file to ensure that '&3' and its associated entries are correct. If '&4' is shown, check that it is a correct MQSeries system directory path for '&3'.

AMQ7707 DMPMQLOG command has failed: CompCode = 0x&1.

Explanation: The DMPMQLOG command has detected an error and the MQSeries recording routine has been called. Possible reasons for this include a damaged log file, a problem during initialization for the queue manager or an internal MQSeries failure.

User Response: Check that the queue manager being used by DMPMQLOG, as specified by you using the -m command option or defaulted, exists and is not currently running. If it does not exist, try the command again specifying an existing queue manager. If it is running, stop the queue manager and then try the command again. Otherwise, use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Note the completion code (CompCode) and then contact your IBM support center.

AMQ7708 DMPMQLOG command has used an invalid default queue manager name.

Explanation: You started the DMPMQLOG command without specifying option -m (queue manager name option) and so your MQSeries default queue manager name has been used. However, this default name either could not be found or is invalid.

User Response: Check that the default queue manager name exists and is valid, and then try the command again.

AMQ7709 DMPMQLOG command has used an invalid combination of options.

Explanation: You started the DMPMQLOG command specifying an invalid combination of the options -b (base LSN option), -s (start LSN option) and -n (extent number option). Only 1 or none of these options may be specified.

User Response: Refer to the command syntax and then try the command again.

AMQ7710 DMPMQLOG command has used option -n which is invalid for circular logging.

Explanation: You started the DMPMQLOG command specifying option -n (extent number option) but this is not valid when your MQSeries log is defined as circular.

User Response: Use a different option and then try the command again.

AMQ7711 DMPMQLOG command has used option -m with a value that is too long.

Explanation: You started the DMPMQLOG command specifying option -m (queue manager name option) with a value that is more than &1 characters.

User Response: Specify a shorter queue manager name and then try the command again.

AMQ7712 DMPMQLOG command has used option -f with a value which is too long.

Explanation: You started the DMPMQLOG command specifying option -f (log file path option) with a value which is more than &1 characters.

User Response: Specify a shorter log file path name and then try the command again.

AMQ7713 DMPMQLOG command was unable to allocate sufficient storage.

Explanation: The DMPMQLOG command has been unable to allocate some storage.

User Response: Free some storage and then try the command again.

AMQ7714 DMPMQLOG command has reached the end of the log.

Explanation: The DMPMQLOG command has processed any log data and has now reached the end of the log.

User Response: None.

AMQ7715 DMPMQLOG command cannot open file '&3'.

Explanation: The DMPMQLOG command was unable to open file '&3' for reading.

User Response: Check that the file exists, can be opened for reading, and that you have authority to access it, and then try the command again.

AMQ7716 DMPMQLOG command has finished unsuccessfully.

Explanation: The DMPMQLOG command has finished with your request but an error has been detected. The previous message issued by the command can be used to identify the error.

User Response: Refer to the previous message issued by the command.

AMQ7717 DMPMQLOG command has failed to initialize: CompCode = 0x&1.

Explanation: The DMPMQLOG command has failed during its initialization and the MQSeries recording routine has been called. Possible reasons for this include that your queue manager is already running. The completion code can be used to identify the error.

User Response: Check that the queue manager being used by DMPMQLOG, as specified by you using the -m command option or defaulted, exists and is not currently running. If it is running, stop the queue manager and then try the command again. Otherwise, use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ7718 **DMPMQLOG command is using a default of '&3' for the queue manager name.**

Explanation: You have started the DMPMQLOG command without specifying option -m (queue manager name option) and so a default value of '&3' is being used. This value is obtained from your MQSeries default queue manager name.

User Response: None.

AMQ7719 **DMPMQLOG command is using a default of '&3' for the starting dump location.**

Explanation: You have started the DMPMQLOG command without specifying option -b (base LSN option), option -s (start LSN option) or option -n (extent number option), and so a default value of '&3' is being used. This value is the Log Sequence Number (LSN) of the first record in the active part of the log, and will be used as the location from which to start dumping.

User Response: None.

AMQ7720 **DMPMQLOG command is using extent '&1' but the current extent is '&2'.**

Explanation: You have started the DMPMQLOG command specifying option -n (extent number option) with a value of '&1' but this value is greater than '&2', which represents the extent currently being used.

User Response: When using option -n, specify its value as being less than or equal to the extent number currently being used.

AMQ7721 **DMPMQLOG command has not found any log records in extent number '&1'.**

Explanation: During its normal processing, the DMPMQLOG command did not find any log records in this extent.

User Response: None.

AMQ7722 **DMPMQLOG command cannot find the object catalogue for queue manager '&3'.**

Explanation: The DMPMQLOG command is using the queue manager named '&3' but cannot find the manager's object catalogue file. This file should have been created at the time the queue manager was created.

User Response: Refer to the "System Management Guide" for a description of the location and name of the object catalogue file. Check that the file exists and is available for use by this command. If it does not exist then you will need to re-create the queue manager.

AMQ7723 **DMPMQLOG command cannot find the requested Log Sequence Number (LSN).**

Explanation: The DMPMQLOG command has been started with an LSN but it cannot be found in the log.

User Response: Check for an existing LSN and then try the command again.

AMQ7724 **DMPMQLOG command cannot use the requested extent number.**

Explanation: The DMPMQLOG command has been started with an extent number but it is beyond the end of the log.

User Response: Check for an existing extent number and then try the command again.

AMQ7725 **DMPMQLOG command cannot find an old Log Sequence Number (LSN).**

Explanation: The DMPMQLOG command has been started specifying an LSN which is older than the log's base LSN. However, the specified LSN could not be found.

User Response: Check for an existing LSN and then try the command again.

AMQ7726 **DMPMQLOG command has used option -s with an incorrect value for circular logging.**

Explanation: You started the DMPMQLOG command specifying option -s (start LSN option) with a value which is less than the base LSN of a log which is defined as circular. LSN values less than the base LSN can only be specified when using a linear log.

User Response: When using option -s with a circular log, specify an option value which is equal or greater to the log's base LSN, and then try the command again.

AMQ7806 **Could not start MQSeries web administration server: &1.**

Explanation: An unsuccessful attempt was made to start the MQSeries web administration server on port &1.

User Response: Check the product is installed correctly; the required registry keys and values are correct and the web server port is not already in use. If the problem persists contact your service representative.

AMQ7807 **MQSeries web administration server running.**

Explanation: MQSeries web administration server running. Listening on port &4, root directory is &5.

User Response: None.

AMQ7808   **Internal run-time error in MQSeries web administration: &4.**

Explanation: MQSeries web administration had the following internal run-time error: &4.

User Response: Check that: the product is installed correctly and that the required registry keys and values are correct. If the problem persists contact your service representative.

AMQ7809   **MQSeries web administration user limit reached.**

Explanation: The maximum number of concurrent MQSeries web administration users has been reached (&4).

User Response: Use the MQServices 'Web Administration Server' properties page in the Microsoft Management Console to increase the value of the web administration 'MaxClients' parameter.

AMQ7810   **Failed to create class, reason code: &1.**

Explanation: While trying to create class &3 on &4 error code &1 was encountered. The associated error message generated by the operating system is: &5

User Response: Check the system documentation to determine the course of action required to rectify the problem.

AMQ7880   **Error code &1 starting &4/&3 MQSeries service.**

Explanation: The MQSeries service was unable to start &4/&3. The error message reported was as follows: &5

User Response: Use the MQSeries MMC Services Manager to investigate why the service could not begin. If recovery for this service is active, MQSeries will attempt to recover.

AMQ7882   **Attempting to recover &4/&3 MQSeries service.**

Explanation: The MQSeries service has detected that &4/&3 has failed, and is attempting to restart it.

User Response: No Action Required.

AMQ7883   **&4/&3 MQSeries service started from recovery.**

Explanation: The MQSeries service has successfully recovered &4/&3.

User Response: No Action Required.

AMQ7884   **Unable to recover &4/&3 MQSeries service.**

Explanation: The MQSeries service has attempted to recover &4/&3, but all attempts have failed. There will be no more attempts to recover this service.

User Response: Use the MQSeries MMC Services Manager to investigate why the service failed and could not be restarted.

AMQ7884   **Unable to recover &4/&3 MQSeries service.**

Explanation: The MQSeries service has attempted to recover &4/&3, but all attempts have failed. There will be no more attempts to recover this service.

User Response: Use the MQSeries MMC Services Manager to investigate why the service failed and could not be restarted.

AMQ7886   **Unable to create queue manager &4.**

Explanation: Queue manager &4 could not be created. MQSeries returned error &1: &5

User Response: No action required.

AMQ7890   **Unable to open mapped file containing MQSeries performance data.**

Explanation: The MQSeries extensible counter dll was unable to open a mapped file used to collect queue performance data. Your system may be running short on virtual memory.

User Response: No action required. Performance statistics for MQSeries queues will not be displayed.

AMQ7891   **Unable to create a mutex to access MQSeries performance data.**

Explanation: The MQSeries extensible counter dll was unable to create a mutex required to synchronize collection of queue performance data

User Response: No action required. Performance statistics for MQSeries queues will not be displayed.

AMQ7892   **Unable to map to shared memory file containing MQSeries performance data.**

Explanation: The MQSeries extensible counter dll was unable to map the shared memory file required for collection of queue performance data.

User Response: No action required. Performance statistics for MQSeries queues will not be displayed.

AMQ7893   **Unable to open "Performance" key for MQSeries services. Status code: &1**

Explanation: The MQSeries extensible counter dll was unable to obtain performance counter values from the "Performance" key for MQSeries services. Status code is the return value from the Windows registry call RegOpenKeyEx.

User Response: No action required. Performance statistics for MQSeries queues will not be displayed.

AMQ7894   **Unable to read the "Performance\First Counter" value for MQSeries services. Status code: &1**

Explanation: The MQSeries extensible counter dll was unable to obtain performance counter values from the "Performance\First Counter" key for MQSeries services. Status code is the return value from the Windows registry call RegOpenKeyEx.

User Response: No action required. Performance statistics for MQSeries queues will not be displayed.

AMQ7895   Unable to read the ″Performance\First Help″ value for MQSeries services. Status code: &1

Explanation: The MQSeries extensible counter dll was unable to obtain performance counter values from the ″Performance\First Counter″ key for MQSeries services. Status code is the return value from the Windows registry call RegOpenKeyEx.

User Response: No action required. Performance statistics for MQSeries queues will not be displayed.

AMQ7901   The data-conversion exit &3 has not loaded.

Explanation: The data-conversion exit program, &3, failed to load. The internal function gave exception &4.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ7902   The data-conversion exit &3 was not loaded. The operating system call &4 returned &1.

Explanation:

User Response: Specify REPLACE to over-write the existing file, or choose a different output file name.

AMQ7903   The data-conversion exit &3 cannot be found.

Explanation: Message data conversion has been requested for an MQSeries message with a user-defined format, but the necessary data-conversion exit program, &3, cannot be found. The internal function gave exception &4.

User Response: Check that the necessary data-conversion exit &3 exists.

AMQ7904   The data-conversion exit &3 cannot be found, or loaded.

Explanation: Message data conversion was requested for an MQSeries message with a user-defined format, but the necessary data conversion exit program, &3, was not found, or loaded. The &4 function call gave a return code of &1.

User Response: Check that the necessary data conversion exit routine exists in one of the standard directories for dynamically loaded modules. If necessary, inspect the generated output to examine the message descriptor (MQMD structure) of the MQSeries message for the conversion which was requested. This may help you to determine where the message originated.

AMQ7905   Unexpected exception &4 in data-conversion exit.

Explanation: The data-conversion exit program, &3, ended with an unexpected exception &4. The message has not been converted.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ7907   Unexpected exception in data-conversion exit.

Explanation: The data-conversion exit routine, &3, ended with an unexpected exception. The message has not been converted.

User Response: Correct the error in the data-conversion exit routine.

AMQ7921   An incorrect eye-catcher field in an MQDXP structure has been detected.

Explanation: The MQDXP structure passed to the Internal Formats Conversion routine contains an incorrect eye-catcher field.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ7922   A PCF message is incomplete.

Explanation: Message data conversion cannot convert a message in Programmable Command Format (PCF) because the message is only &1 bytes long and does not contain a PCF header. The message has either been truncated, or it contains data that is not valid.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7923   A message had an unrecognized integer encoding - &1.

Explanation: Message data conversion cannot convert a message because the integer encoding value of the message, &1, was not recognized.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7924   **Bad length in the PCF header (length = &1).**

Explanation: Message data conversion cannot convert a message in Programmable Command Format (PCF) because the PCF header structure contains an incorrect length field. Either the message has been truncated, or it contains data that is not valid.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7925   **Message version &1 is not supported.**

Explanation: Message data conversion cannot convert a message because the Version field of the message contains an incorrect value.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7926   **A PCF message has an incorrect parameter count value &1.**

Explanation: Message data conversion cannot convert a message in Programmable Command Format (PCF) because the parameter count field of the PCF header is incorrect.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7927   **Bad type in PCF structure number &1 (type = &2).**

Explanation:

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7928   **Bad length in PCF structure number &1 (length = &2).**

Explanation:

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these

files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7929   **A PCF structure is incomplete.**

Explanation: Message data conversion cannot convert a message in Programmable Command Format (PCF) because structure number &1, of Type value &2, within the message is incomplete. The message has either been truncated, or it contains data that is not valid.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7930   **Bad CCSID in PCF structure number &1 (CCSID = &2).**

Explanation:

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7931   **Bad length in PCF structure number &1 (length = &2).**

Explanation: Message data conversion cannot convert a message in Programmable Command Format (PCF) because one of the structures of the message contains an incorrect length field.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7932   **Bad count in PCF structure number &1 (count = &2).**

Explanation: Message data conversion cannot convert a message in Programmable Command Format (PCF) because a StringList structure of the message contains an incorrect count field.

User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor, the headers of the message, and the incorrect structure to determine the source of the message, and to see how data that is not valid became included in the message.

AMQ7933   Bad string length in PCF structure.
Explanation: Message data conversion cannot convert a message in Programmable Command Format (PCF) because structure number &1 of the message contains an incorrect string length value &2.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor, the headers of the message, and the incorrect structure to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7934   Wrong combination of MQCCSI_DEFAULT with MQCCSI_EMBEDDED.
Explanation: Message data conversion could not convert a message in Programmable Command Format (PCF) because structure &1 of the message contained a CodedCharSetId field of MQCCSI_DEFAULT while the message itself had a CodedCharSetId of MQCCSI_EMBEDDED. This is an incorrect combination.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor, the headers of the message and the incorrect structure to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7935   Bad CCSID in message header (CCSID = &1).
Explanation: Message data conversion could not convert a message because the Message Descriptor of the message contained an incorrect CodedCharSetId field.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Do not discard these files until the problem has been resolved. Use the file containing the Message Descriptor of the message to determine the source of the message and to see how data that is not valid became included in the message.

AMQ7936   The file &3 already exists.
Explanation: The output file already exists, but REPLACE has not been specified.
User Response: Specify REPLACE to over-write the existing file, or select a different output file name.

AMQ7937   Structure length &1 in MQFMT_IMS_VAR_STRING format message is not valid.
Explanation: This error is detected when attempting data conversion. The valid range for the length is 4

(with no string data) to 32767. The message is returned unconverted with a reason code of MQRC_CONVERTED_STRING_TOO_BIG.
User Response: Check the content of the message before data conversion and correct the message format. When converting data using two or more bytes per character, remember that the number of bytes in each character can change during data conversion. This causes the message lengths to change.

AMQ7943   Usage: crtmqcvx SourceFile TargetFile
Explanation: None.
User Response: None.

AMQ7953   One structure has been parsed.
Explanation: The crtmqcvx command has parsed one structure.
User Response: None.

AMQ7954   &1 structures have been parsed.
Explanation: The crtmqcvx command has parsed &1 structures.
User Response: None.

AMQ7955   Unexpected field: &1.
Explanation: The field within the structure is of a type that is not recognized.
User Response: Correct the field and retry the command.

AMQ7956   Bad array dimension.
Explanation: An array field of the structure has an incorrect dimension value.
User Response: Correct the field and retry the command.

AMQ7957   Warning at line &1.
Explanation: The structure contains another field after a variable length field.
User Response: Correct the structure and retry the command.

AMQ7958   Error at line &1 in field &3.
Explanation: Field name '&3' is a field of type 'float'. Fields of type float are not supported by this command.
User Response: Either correct the structure to eliminate fields of type float, or write your own routine to support conversion of these fields.

AMQ7959   Error at line &1 in field &3.
Explanation: Field name '&3' is a field of type 'double'. Fields of type double are not supported by this command.
User Response: Either correct the structure to eliminate fields of type double, or write your own routine to support conversion of these fields.

AMQ7960   Error at line &1 in field &3.
Explanation: Field name '&3' is a 'pointer' field. Fields
of type pointer are not supported by this command.
User Response: Either correct the structure to
eliminate fields of type pointer, or write your own
routine to support conversion of these fields.

AMQ7961   Error at line &1 in field &3.
Explanation: Field name '&3' is a 'bit' field. Bit fields
are not supported by this command.
User Response: Either correct the structure to
eliminate bit fields, or write your own routine to
support conversion of these fields.

AMQ7962   No input file specified.
Explanation: This command requires that an input file
is specified.
User Response: Specify the name of the input file and
retry the command.

AMQ7963   No output file specified.
Explanation: This command requires that an output
file name is specified.
User Response: Specify the name of the output file
and retry the command.

AMQ7964   Unexpected option &3.
Explanation: The option specified is not valid for this
command.
User Response: Retry the command with a valid
option.

AMQ7965   Incorrect number of arguments.
Explanation: The command was passed an incorrect
number of arguments.
User Response: Retry the command, passing it the
correct number of arguments.

AMQ7968   Cannot open file '&3'.
Explanation: You cannot open the file &3.
User Response: Check that you have the correct
authorization to the file and retry the command.

AMQ7969   Syntax error.
Explanation: This line of the input file contains a
language syntax error.
User Response: Correct the syntax error and retry the
command.

AMQ7970   Syntax error on line &1.
Explanation: This message identifies where, in the
input file, a previously reported error was detected.
User Response: Correct the error and retry the
command.

Chapter 7. AMQ8000-AMQ8999 MQSeries administration messages

> **Reading variables in the message**
>
> Some messages display text or numbers that vary according to the circumstances giving rise to the message; these are known as *message variables*. The message variables are indicated in this book by the use of the '&' symbol and a number: &1, &2, and so on.
>
> In some cases a message may have variables in the Explanation or User action. Find the values of the message variables by looking in the error log. The complete message, including the Explanation and the User action, is recorded there.

AMQ8001   **MQSeries queue manager created.**
Explanation: MQSeries queue manager &5 created.
User Response: None.

AMQ8002   **MQSeries queue manager deleted.**
Explanation: MQSeries queue manager &5 deleted.
User Response: None.

AMQ8003   **MQSeries queue manager '&5' started.**
Explanation: MQSeries queue manager '&5' started.
User Response: None.

AMQ8004   **MQSeries queue manager ended.**
Explanation: MQSeries queue manager &5 ended.
User Response: None.

AMQ8005   **MQSeries queue manager changed.**
Explanation: MQSeries queue manager &5 changed.
User Response: None.

AMQ8006   **MQSeries queue created.**
Explanation: MQSeries queue &5 created.
User Response: None.

AMQ8007   **MQSeries queue deleted.**
Explanation: MQSeries queue &5 deleted.
User Response: None.

AMQ8008   **MQSeries queue changed.**
Explanation: MQSeries queue &5 changed.
User Response: None.

AMQ8010   **MQSeries process created.**
Explanation: MQSeries process &5 created.
User Response: None.

AMQ8011   **MQSeries process deleted.**
Explanation: MQSeries process &5 deleted.
User Response: None.

AMQ8012   **MQSeries process changed.**
Explanation: MQSeries process &5 changed.
User Response: None.

AMQ8014   **MQSeries channel created.**
Explanation: MQSeries channel &5 created.
User Response: None.

AMQ8015   **MQSeries channel deleted.**
Explanation: MQSeries channel &5 deleted.
User Response: None.

AMQ8016   **MQSeries channel changed.**
Explanation: MQSeries channel &5 changed.
User Response: None.

AMQ8018   **Start MQSeries channel accepted.**
Explanation: MQSeries channel &5 is being started. The start channel function has been initiated. This involves a series of operations across the network before the channel is actually started. The channel status displays "BINDING" for a short period while communication protocols are negotiated with the channel with whom communication is being initiated.
User Response: None.

AMQ8019   **Stop MQSeries channel accepted.**
Explanation: MQSeries channel &5 has been requested to stop.
User Response: None.

AMQ8020   Ping MQSeries channel complete.
Explanation: Ping MQSeries channel &5 complete.
User Response: None.

AMQ8021   MQSeries Listener program started.
Explanation: The MQSeries channel listener program has been started.
User Response: None.

AMQ8022   MQSeries queue cleared.
Explanation: All messages on MQSeries queue &5 have been deleted.
User Response: None.

AMQ8023   MQSeries channel reset.
Explanation: MQSeries channel &5 has been reset.
User Response: None.

AMQ8024   MQSeries channel initiator started.
Explanation: The channel initiator for MQSeries queue &5 has been started.
User Response: None.

AMQ8025   MQSeries channel resolved.
Explanation: In doubt messages for MQSeries channel &5 have been resolved.
User Response: None.

AMQ8026   End MQSeries queue manager accepted.
Explanation: A controlled stop request has been initiated for MQSeries queue manager &5.
User Response: None.

AMQ8027   MQSeries command server started.
Explanation: The MQSeries command server has been started.
User Response: None.

AMQ8028   MQSeries command server ended.
Explanation: The MQSeries command server has been stopped.
User Response: None.

AMQ8029   MQSeries authority granted.
Explanation: Authority for MQSeries object &5 granted.
User Response: None.

AMQ8030   MQSeries authority revoked.
Explanation: Authority for MQSeries object &5 revoked.
User Response: None.

AMQ8033   MQSeries object recreated.
Explanation: MQSeries object &5 has been recreated from image.
User Response: None.

AMQ8034   MQSeries object image recorded.
Explanation: Image of MQSeries object &5 has been recorded.
User Response: None.

AMQ8041   The queue manager cannot be restarted or deleted because processes, that were previously connected, are still running.
Explanation: Processes, that were connected to the queue manager the last time it was running, are still active. The queue manager cannot be restarted.
User Response: Stop the processes and try to start the queue manager.

AMQ8043   Non runtime application attempted to connect to runtime only queue manager.
Explanation: A non runtime application attempted to connect to a queue manager on a node where support for non runtime applications has not been installed. The connect attempt will be rejected with a reason of MQRC_ENVIRONMENT_ERROR.
User Response: If the node is intended to support only runtime applications, investigate why a non runtime application has attempted to connect to the queue manager. If the node is intended to support non runtime only applications, investigate if the base option has been installed. The base option must be installed if non runtime applications are to run on this node.

AMQ8044   The queue manager was not added to the Service Control Manager.
Explanation: The attempt to add the queue manager to the Windows NT registry failed. This may be because the appropriate entry could not be opened or modified.
User Response: Check that the registry has sufficient space on your disk to grow and that your account has the authority to modify it.

AMQ8045   An error occurred while removing the queue manager from the Service Control Manager.
Explanation: The attempt to remove the queue manager from the Windows registry failed. This may be because the appropriate entry could not be opened or modified.
User Response: Check that the registry has sufficient space on your disk to grow and that your account has the authority to modify it.

AMQ8046 **The queue manager entry in the NT Service Control Manager was successfully updated.**
Explanation: In future the queue manager will be started by the Windows NT service control manager using the specified startup script whenever the system starts.
User Response: In order for the queue manager to be automatically started by the system it is necessary to restart your computer.

AMQ8047 **The function &3 of amqscm.exe on line %1 returned code %2.**
Explanation: An internal error occurred while executing the IBM MQSeries service.
User Response: Use the standard facilities supplied with your system to record the problem identifier, and to save any generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ8048 **Default objects statistics : &1 created. &2 replaced. &3 failed.**
Explanation: Information on the number of objects created or replaced successfully as well as any failures that occurred while creating the default objects.
User Response: None.

AMQ8049 **Object &4. Unable to create or replace.**
Explanation: While creating or replacing the default object &4 for MQSeries queue manager &5 an error occurred. The error was due to improper authorization. The reason code is &1.
User Response: Check this log for more details of what the problem may be. Make sure there are sufficient resources such as disk space and storage. For damaged or corrupted objects, replace these from backup objects. If all else fails, delete the queue manager &5 using dltmqm and create it again using crtmqm.

AMQ8053 **Object &4. Unable to create or replace.**
Explanation: While creating or replacing the default object &4 for MQSeries queue manager &5 an error occurred. The error was due to a broken connection. The reason code is &1.
User Response: Check this log for more details of what the problem may be. Make sure there is sufficient resources such as disk space and storage. For damaged or corrupted objects, replace these from backup objects. If all else fails, delete the queue manager &5 using dltmqm and create it again using crtmqm.

AMQ8054 **Object &4. Unable to create or replace.**
Explanation: While creating or replacing the default object &4 for MQSeries queue manager &5 an error occurred. The error was due to unavailable storage. The reason code is &1.
User Response: Check this log for more details of what the problem may be. Make sure there is sufficient resources such as disk space and storage. For damaged or corrupted objects, replace these from backup objects. If all else fails, delete the queue manager &5 using dltmqm and create it again using crtmqm.

AMQ8055 **Object &4. Unable to create or replace.**
Explanation: While creating or replacing the default object &4 for MQSeries queue manager &5 an error occurred. The error was due to a damaged object. The reason code is &1.
User Response: Check this log for more details of what the problem may be. Make sure there is sufficient resources such as disk space and storage. For damaged or corrupted objects, replace these from backup objects. If all else fails, delete the queue manager &5 using dltmqm and create it again using crtmqm.

AMQ8056 **Object &4. Unable to create or replace.**
Explanation: While creating or replacing the default object &4 for MQSeries queue manager &5 an error occurred. The error was due to a channel definition error. The error code is &1 (X'&2').
User Response: Check this log for more details of what the problem may be. Make sure there is sufficient resources such as disk space and storage. For damaged or corrupted objects, replace these from backup objects. If all else fails, delete the queue manager &5 using dltmqm and create it again using crtmqm.

AMQ8057 **Object &4. Unable to create or replace.**
Explanation: While creating or replacing the default object &4 for MQSeries queue manager &5 an error occurred. The error was due to invalid records in the channel definition file. The error code is &1 (X'&2').
User Response: Check this log for more details of what the problem may be. Make sure there is sufficient resources such as disk space and storage. For damaged or corrupted objects, replace these from backup objects. If all else fails, delete the queue manager &5 using dltmqm and create it again using crtmqm.

AMQ8058 **Object &4. Unable to create or replace.**
Explanation: While creating or replacing the default object &4 for MQSeries queue manager &5 an error occurred. The error was due to not finding the channel definition file. The error code is &1 (X'&2').
User Response: Check this log for more details of what the problem may be. Make sure there is sufficient resources such as disk space and storage. For damaged or corrupted objects, replace these from backup objects. If all else fails, delete the queue manager &5 using

dltmqm and create it again using crtmqm.

AMQ8059   Object &4. Unable to create or replace.
Explanation: While creating or replacing the default object &4 for MQSeries queue manager &5 an error occurred. The error was due to an unexpected error, error code &1 (X'&2').
User Response: Check this log for more details of what the problem may be. Make sure there is sufficient resources such as disk space and storage. For damaged or corrupted objects, replace these from backup objects. If all else fails, delete the queue manager &5 using dltmqm and create it again using crtmqm.

AMQ8060   Cannot open file '&3'.
Explanation: The service command file '&3' for queue manager '&4' cannot be found.
User Response: Check that the correct command file name was specified when the scmmqm command was issued, and that this command file is available.

AMQ8061   Command '&4' is not valid.
Explanation: The command '&4' at line &1 in the MQSeries service command file '&3' for queue manager '&5' is not valid for use in the service command file. The line is ignored.
User Response: Check the contents of the file and retry the operation.

AMQ8062   Unexpected return code, '&1', from command '&3'.
Explanation: An unexpected return code, '&1', was returned by command '&3'. This command was issued by the MQSeries service for queue manager '&4'.
User Response: Verify that the command and parameters are correct.

AMQ8063   Not authorized to issue command '&3'.
Explanation: The current user '&5' is not authorized to issue the command '&3'. The command is ignored.
User Response: Add the user to the local 'mqm' security group and retry the operation.

AMQ8064   Not authorized to start trusted application.
Explanation: The user '&5' is not authorized to start the trusted application '&3'. The application has not started.
User Response: Add the user to the local 'mqm' security group and restart the application.

AMQ8065   Local group '&3' not found.
Explanation: The local group '&3' is unavailable. It is not possible to verify that the user is authorized. The function cannot continue.
User Response: Create the required local group and retry the operation.

AMQ8066   Local mqm group not found.
Explanation: The local mqm group is unavailable. It is not possible to verify that the user is authorized. The function cannot continue.
User Response: Create the local mqm group and retry the operation.

AMQ8069   ApplicationGroup for the crtmqm command does not contain the mqm userid.
Explanation: MQSeries queue manager &5 not created. The ApplicationGroup specified for the crtmqm command must contain the mqm userid when the RestrictedMode option (-g) is specified.
User Response: None.

AMQ8070   ApplicationGroup for crtmqm command is not defined.
Explanation: MQSeries queue manager &5 not created. RestrictedMode option (-g) specified, but the ApplicationGroup does not exist.
User Response: None.

AMQ8071   RestrictedMode option not supported on this platform.
Explanation: MQSeries queue manager &5 not created. The RestrictedMode option was specified but is not supported on this platform.
User Response: None.

AMQ8072   User '&3' not found.
Explanation: The user '&3' was not found in the security database of the local computer, the domain controller for the local computer or the domain controller for the process executing this command. It is not possible to verify that the user is authorized. The function cannot continue.
User Response: Create the required user name.

AMQ8073   Authorization failed because SID: (&3) could not be resolved.
Explanation: The Object Authority Manager was unable to resolve the specified SID into entity and domain information.
User Response: Ensure that the application provides a SID that is recognized on this system, that all necessary domain controllers are available, and that the security policy is set as you required.

AMQ8074   Authorization failed as the SID &3 does not match the entity &4.
Explanation: The Object Authority Manager received inconsistent data - the supplied SID does not match that of the supplied entity information.
User Response: Ensure that the application is supplying valid entity and SID information.

AMQ8075   Authorization failed because the SID for entity &3 cannot be obtained.
Explanation: The Object Authority Manager was unable to obtain a SID for the specified entity.
User Response: Ensure that the entity is valid, and that all necessary domain controllers are available.

AMQ8076   Authorization failed because no SID was supplied for entity &3.
Explanation: The Object Authority Manager was not supplied with SID information for the specified entity, and the security policy is set to 'UseNTSIDs'.
User Response: Ensure that the application is supplying a valid SID, and that the security policy is set as you require.

AMQ8077   Entity &3 has insufficient authority to access object &4.
Explanation: The specified entity is not authorized to access the required object. The following requested permissions are unauthorized: &5
User Response: Ensure that the correct level of authority has been set for this entity against the required object, or ensure that the entity is a member of a privileged group.

AMQ8079   Access was denied when attempting to retrieve group membership information for user '&3'.
Explanation: MQSeries, running with the authority of user '&4', was unable to retrieve group membership information for the specified user.
User Response: Ensure Active Directory™ access permissions allow user '&4' to read group memberships for user '&3'. To retrieve group membership information for a domain user, MQSeries must run with the authority of a domain user.

AMQ8101   Unexpected error (&1).
Explanation: An unexpected reason code with hexadecimal value &1 was received from the MQSeries queue manager during command processing. (Note that hexadecimal values in the range X'07D1'-X'0BB7' correspond to MQI reason codes 2001-2999.) More information might be available in the log. If the reason code value indicates that the error was associated with a particular parameter, the parameter concerned is &4.
User Response: Correct the error and then try the command again.

AMQ8102   MQSeries object name specified in &4 not valid.
Explanation: MQSeries object name &5 specified in &4 is not valid. The length of the name must not exceed 48 characters, or 20 characters if it is a channel name. The name should contain the following characters only: lowercase a-z, uppercase A-Z, numeric 0-9, period (.), forward slash (/), underscore (_) and percent sign (%).

User Response: Change the length of the parameter value or change the parameter value to contain a valid combination of characters, then try the command again.

AMQ8103   Insufficient storage available.
Explanation: There was insufficient storage available to perform the requested operation.
User Response: Free some storage and then try the command again.

AMQ8104   MQSeries directory &3 not found.
Explanation: Directory &3 was not found. This directory is created when MQSeries is installed successfully. Refer to the log for more information.
User Response: Verify that installation of MQSeries was successful. Correct the error and then try the command again.

AMQ8105   Object error.
Explanation: An object error occurred. Refer to the log for more information.
User Response: Correct the error and then try the command again.

AMQ8106   MQSeries queue manager being created.
Explanation: The MQSeries queue manager is being created.
User Response: Wait for the creation process to complete and then try the command again.

AMQ8107   MQSeries queue manager running.
Explanation: The MQSeries queue manager is running.
User Response: None.

AMQ8108   MQSeries queue manager ending.
Explanation: The MQSeries queue manager is ending.
User Response: Wait for the MQSeries queue manager to end and then try the command again.

AMQ8109   MQSeries queue manager being deleted.
Explanation: The MQSeries queue manager is being deleted.
User Response: Wait for the deletion process to complete.

AMQ8110   MQSeries queue manager already exists.
Explanation: MQSeries queue manager &5 already exists.
User Response: None.

AMQ8117   MQSeries queue manager deletion incomplete.
Explanation: Deletion of MQSeries queue manager &5 was only partially successful. An object was not found,

or could not be deleted. Refer to the log for more information.
User Response: Delete any remaining MQSeries queue manager objects.

AMQ8118 **MQSeries queue manager does not exist.**
Explanation: MQSeries queue manager &5 does not exist.
User Response: Create the message queue manager (crtmqm command) and then try the command again.

AMQ8135 **Not authorized.**
Explanation: You are not authorized to perform the requested operation for the MQSeries object &5 specified in &3. Either you are not authorized to perform the requested operation, or you are not authorized to the specified MQSeries object. For a copy command, you may not be authorized to the specified source MQSeries object, or, for a create command, you may not be authorized to the system default MQSeries object of the specified type.
User Response: Obtain the necessary authority from your security officer or MQSeries administrator. Then try the command again.

AMQ8137 **MQSeries queue manager already starting.**
Explanation: The strmqm command was unsuccessful because MQSeries queue manager &5 is already starting.
User Response: Wait for the strmqm command to complete.

AMQ8138 **The MQSeries queue has an incorrect type.**
Explanation: The operation is not valid with MQSeries queue &5 because it is not a local queue.
User Response: Change the QNAME parameter to specify an MQSeries queue of the correct type.

AMQ8139 **Already connected.**
Explanation: A connection to the MQSeries queue manager already exists.
User Response: None.

AMQ8140 **Resource timeout error.**
Explanation: A timeout occurred in the communication between internal MQSeries queue manager components. This is most likely to occur when the system is heavily loaded.
User Response: Wait until the system is less heavily loaded, then try the command again.

AMQ8141 **MQSeries queue manager starting.**
Explanation: MQSeries queue manager &5 is starting.
User Response: Wait for the MQSeries queue manager startup process to complete and then try the command again.

AMQ8142 **MQSeries queue manager stopped.**
Explanation: MQSeries queue manager &5 is stopped.
User Response: Use the strmqm command to start the MQSeries queue manager, and then try the command again.

AMQ8143 **MQSeries queue not empty.**
Explanation: MQSeries queue &5 specified in &2 is not empty or contains uncommitted updates.
User Response: Commit or roll back any uncommitted updates. If the command is DELETE QLOCAL, use the CLEAR QLOCAL command to clear the messages from the MQSeries queue. Then try the command again.

AMQ8144 **Log not available.**
Explanation: The MQSeries logging resource is not available.
User Response: Use the dltmqm command to delete the MQSeries queue manager and then the crtmqm command to create the MQSeries queue manager. Then try the command again.

AMQ8145 **Connection broken.**
Explanation: The connection to the MQSeries queue manager failed during command processing. This may be caused by an endmqm command being issued by another user, or by an MQSeries queue manager error.
User Response: Use the strmqm command to start the message queue manager, wait until the message queue manager has started, and try the command again.

AMQ8146 **MQSeries queue manager not available.**
Explanation: The MQSeries queue manager is not available because it has been stopped or has not been created.
User Response: Use the crtmqm command to create the message queue manager, or the strmqm command to start the message queue manager as necessary. Then try the command again.

AMQ8147 **MQSeries object &3 not found.**
Explanation: If the command entered was Change or Display, the MQSeries object &3 specified does not exist. If the command entered was Copy, the source MQSeries object does not exist. If the command entered was Create, the system default MQSeries object of the specified type does not exist.
User Response: Correct the MQSeries object name and then try the command again or, if you are creating a new MQSeries queue or process object, either specify all parameters explicitly or ensure that the system

| default object of the required type exists. The system default queue names are SYSTEM.DEFAULT.LOCAL.QUEUE, SYSTEM.DEFAULT.ALIAS.QUEUE and SYSTEM.DEFAULT.REMOTE.QUEUE. The system default process name is SYSTEM.DEFAULT.PROCESS.

| **AMQ8148   MQSeries object in use.**
| **Explanation:** MQSeries object &5 specified in &3 is in use by an MQSeries application program.
| **User Response:** Wait until the MQSeries object is no longer in use and then try the command again. If the command is ALTER or CHANGE, specify FORCE to force the processing of the MQSeries object regardless of any application program affected by the change. If the object is the dead-letter queue and the open input count is nonzero, it may be in use by an MQSeries channel. If the object is another MQSeries queue object with a nonzero open output count, it may be in use by an MQSeries channel (of type RCVR or RQSTR). In either case, use the STOP CHANNEL and START CHANNEL commands to stop and restart the channel in order to solve the problem. To alter the queue USAGE the FORCE option must be used if the queue is not empty.

| **AMQ8149   MQSeries object damaged.**
| **Explanation:** The MQSeries object &5 specified in &4 is damaged.
| **User Response:** The MQSeries object contents are not valid. Issue the DISPLAY CHANNEL, DISPLAY QUEUE, or DISPLAY PROCESS command, as required, to determine the name of the damaged object. Issue the DEFINE command, for the appropriate object type, to replace the damaged object, then try the command again.

| **AMQ8150   MQSeries object already exists.**
| **Explanation:** MQSeries object &5 specified for &3 could not be created because it already exists.
| **User Response:** Check that the name is correct and try the command again specifying REPLACE, or delete the MQSeries object. Then try the command again.

AMQ8151   MQSeries object has different type.
Explanation: The type specified for MQSeries object &5 is different from the type of the object being altered or defined.
User Response: Use the correct MQSeries command for the object type, and then try the command again.

AMQ8152   Source MQSeries object has different type.
Explanation: The type of the source MQSeries object is different from that specified.
User Response: Correct the name of the command, or source MQSeries object name, and then try the

command again, or try the command using the REPLACE option.

AMQ8153   Insufficient disk space for the specified queue.
Explanation: The command failed because there was insufficient disk space available for the specified queue.
User Response: Release some disk space and then try the command again.

AMQ8154   API exit load error.
Explanation: The MQSeries queue manager was unable to load the API crossing exit.
User Response: Ensure that the API crossing exit program is valid, and that its name and directory are correctly specified. Correct any error and then try the command again.

AMQ8155   Connection limit exceeded.
Explanation: The queue manager connection limit has been exceeded.
User Response: The maximum limit on the number of MQSeries application programs that may be connected to the MQSeries queue manager has been exceeded. Try the command later.

AMQ8156   MQSeries queue manager quiescing.
Explanation: The MQSeries queue manager is quiescing.
User Response: The queue manager was stopping with -c specified for endmqm. Wait until the queue manager has been restarted and then try the command again.

AMQ8157   Security error.
Explanation: An error was reported by the security manager program.
User Response: Inform your systems administrator, wait until the problem has been corrected, and then try the command again.

AMQ8172   Already disconnected.
Explanation: The MQI reason code of 2018 was returned from the MQSeries queue manager in response to an MQDISC request issued during command processing.
User Response: None.

AMQ8173   No processes to display.
Explanation: There are no matching processes defined on this system.
User Response: Using the DEFINE PROCESS command to create a process.

AMQ8174   No queues to display.
Explanation: There are no matching queues defined on this system.
User Response: Use the appropriate command to define a queue of the type that you require, that is, DEFINE QALIAS, DEFINE QLOCAL, DEFINE QMODEL, or DEFINE QREMOTE.

AMQ8185   Operating system object already exists.
Explanation: The MQSeries object cannot be created because an object that is not known to MQSeries already exists in the MQSeries directory with the name that should be used for the new object. Refer to the log for previous messages.
User Response: Remove the non-MQSeries object from the MQSeries library, and try the command again.

AMQ8186   Image not available for MQSeries object &5.
Explanation: MQSeries object &5 type &3 cannot be recreated because the image is not fully available in the logs that are currently online. Refer to earlier messages in the error log for information about the error logs that need to be brought online for this object to be recreated.
User Response: Bring the relevant error logs online, and try the command again.

AMQ8187   MQSeries object &5 is currently open.
Explanation: MQSeries object &5, type &3, is currently in use, so the &1 command cannot be issued against it. If a generic list was presented to the command, the command is still issued against the other objects in the list.
User Response: Wait until the object is no longer in use, and try the command again.

AMQ8188   Insufficient authorization to MQSeries object &5.
Explanation: You are not authorized to issue the &1 command against MQSeries object &5 type &3. If a generic list was presented to the command, the command is still issued against the other objects in the list.
User Response: Obtain sufficient authorization for the object, and retry the command.

AMQ8189   MQSeries object &5 is damaged.
Explanation: MQSeries object &5 type &4 is damaged and the &3 command cannot be issued against it. If a generic list was presented to the command then the command is still issued against the other objects in the list.
User Response: Issue the appropriate DEFINE command for the object, specifying REPLACE, and then try the command again.

AMQ8190   &3 succeeded on &1 objects and failed on &2 objects.
Explanation: An operation performed on a generic list of objects was not completely successful.
User Response: Examine the log for details of the errors encountered, and take appropriate action.

AMQ8191   MQSeries command server is starting.
Explanation: The MQSeries command server is starting.
User Response: Wait for the strmqcsv command to complete and then try the operation again.

AMQ8192   MQSeries command server already starting.
Explanation: The request to start the MQSeries command server was unsuccessful because the MQSeries command server is already starting.
User Response: Wait for the strmqcsv command to complete.

AMQ8193   MQSeries command server is ending.
Explanation: The MQSeries command server is ending.
User Response: Wait for the endmqcsv command to complete and then try the command again.

AMQ8194   MQSeries command server already ending.
Explanation: The end MQSeries command server request was unsuccessful because the MQSeries command server is already ending.
User Response: Wait for the endmqcsv command to complete.

AMQ8195   MQSeries command server already running.
Explanation: The strmqcsv command was unsuccessful because the MQSeries command server is already running.
User Response: None.

AMQ8196   MQSeries command server already stopped.
Explanation: The request to end the MQSeries command server was unsuccessful because the MQSeries command server is already stopped.
User Response: None.

AMQ8197   Deleted MQSeries queue damaged.
Explanation: The deleted MQSeries queue &5 was damaged, and any messages it contained have been lost.
User Response: None.

AMQ8202   TMQNAME only allowed with channel type *SDR or *SVR.
Explanation: The TMQNAME parameter may only be specified with channel type *SDR or *SVR.
User Response: Remove the TMQNAME parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *SDR or *SVR. Then try the command again.

AMQ8204   MCANAME only allowed with channel type *SDR, *SVR, or *RQSTR.
Explanation: The MCANAME parameter may only be specified with channel type *SDR, *SVR, or *RQSTR.
User Response: Remove the MCANAME parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *SDR, *SVR, or *RQSTR. Then try the command again.

AMQ8205   DSCITV only allowed with channel type *SDR or *SVR.
Explanation: The DSCITV parameter may only be specified with channel type *SDR or *SVR.
User Response: Remove the DSCITV parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *SDR or *SVR. Then try the command again.

AMQ8206   SHORTRTY only allowed with channel type *SDR or *SVR.
Explanation: The SHORTRTY parameter may only be specified with channel type *SDR or *SVR.
User Response: Remove the SHORTRTY parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *SDR or *SVR. Then try the command again.

AMQ8207   SHORTTMR only allowed with channel type *SDR or *SVR.
Explanation: The SHORTTMR parameter may only be specified with channel type *SDR or *SVR.
User Response: Remove the SHORTTMR parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *SDR or *SVR. Then try the command again.

AMQ8208   LONGRTY only allowed with channel type *SDR or *SVR.
Explanation: The LONGRTY parameter may only be specified with channel type *SDR or *SVR.
User Response: Remove the LONGRTY parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *SDR or *SVR. Then try the command again.

AMQ8209   LONGTMR only allowed with channel type *SDR or *SVR.
Explanation: The LONGTMR parameter may only be specified with channel type *SDR or *SVR.
User Response: Remove the LONGTMR parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *SDR or *SVR. Then try the command again.

AMQ8210   PUTAUT only allowed with channel type *RCVR or RQSTR.
Explanation: The PUTAUT parameter may only be specified with channel type *RCVR or RQSTR.
User Response: Remove the PUTAUT parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *RCVR or RQSTR. Then try the command again.

AMQ8211   BATCHINT only allowed with channel type *SDR or *SVR.
Explanation: The BATCHINT parameter may only be specified with channel type *SDR or *SVR.
User Response: Remove the BATCHINT parameter from the command or, if the command is CRTMQMCHL, change the CHLTYPE parameter value to specify *SDR or *SVR. Then try the command again.

AMQ8214   CONNAME parameter required with channel types SDR, RQSTR, CLNTCONN and CLUSSDR. It is also required with channel type CLUSRCVR if TRPTYPE is TCP.
Explanation: The CONNAME parameter must be specified with channel type SDR, RQSTR, CLNTCONN, CLUSSDR and CLUSRCVR.
User Response: Enter a value for parameter CONNAME.

AMQ8219   Command server queue is open, retry later.
Explanation:
User Response: Wait and try again later.

AMQ8226   MQSeries channel already exists.
Explanation: MQSeries channel &3 cannot be created because it already exists.
User Response: Check that the name is correct and try the command again specifying REPLACE, or delete the MQSeries channel and then try the command again.

AMQ8227   Channel &3 not found.
Explanation: The channel could not be found.
User Response: Correct the Channel Name if wrong and then try the command again. For DEFINE CHANNEL check that the Channel Name in error exists.

AMQ8287   No channels with status to display.
Explanation: There are no channels having status
information to display. This indicates either, that the
channel has not been started previously, or, that the
channel has been started but has not yet completed a
transmission sequence.
User Response: None.

AMQ8296   &1 MQSC commands completed
successfully.
Explanation: The &3 command has completed
successfully. The &1 MQSeries commands from &5
have been processed without error and a report written
to the printer spool file.
User Response: None.

AMQ8297   &1 MQSC commands verified
successfully.
Explanation: The &3 command completed
successfully. The &1 MQSeries commands from &5
have been verified and a report written to the printer
spool file.
User Response: None.

AMQ8298   Error report generated for MQSC
command process.
Explanation: The &3 command attempted to process
the sequence of MQSeries commands from &5 and
encountered some errors, however, the operation may
have partially completed. A report has been written to
the printer spool file.
User Response: Examine the spooled printer file for
details of the errors encountered, correct the MQSC
source file, and retry the operation.

AMQ8299   Cannot open &5 for MQSC process.
Explanation: The &1 command failed to open &5 for
MQSeries command processing.
User Response: Check that the intended file exists,
and has been specified correctly. Correct the
specification or create the object, and try the operation
again.

AMQ8302   Internal failure initializing MQSeries
services.
Explanation: An error occurred while attempting to
initialize MQSeries services.
User Response:

AMQ8304   Tracing cannot be started. Too many
traces are already running.
Explanation:
User Response: Stop one or more of the other traces
and try the command again.

AMQ8305   Tracing cannot be started. Too many
traces are already running.
Explanation:
User Response: Stop one or more of the other traces
and try the command again.

AMQ8316   No cluster queue managers to display.
Explanation: There are no matching cluster queue
managers defined on this system.
User Response: None.

AMQ8401   &1 MQSC commands read.
Explanation: The MQSC script contains &1
commands.
User Response: None.

AMQ8402   &1 commands have a syntax error.
Explanation: The MQSC script contains &1 commands
having a syntax error.
User Response: None.

AMQ8403   &1 valid MQSC commands could not be
processed.
Explanation: The MQSC script contains &1 commands
that failed to process.
User Response: None.

AMQ8404   Command failed.
Explanation: An MQSC command has been
recognized, but cannot be processed.
User Response: None.

AMQ8405   Syntax error detected at or near end of
command segment below:-
Explanation: The MQSC script contains &1 commands
having a syntax error.
User Response: None.

AMQ8406   Unexpected 'end of input' in MQSC.
Explanation: An MQSC command contains a
continuation character, but the 'end of input' has been
reached without completing the command.
User Response: None.

AMQ8407   Display Process details.
Explanation: The MQSC DISPLAY PROCESS
command completed successfully, and details follow
this message.
User Response: None.

AMQ8408   Display Queue Manager details.
Explanation: The MQSC DISPLAY QMGR command
completed successfully, and details follow this message.
User Response: None.

AMQ8409   Display Queue details.
Explanation: The MQSC DISPLAY QUEUE command completed successfully, and details follow this message.
User Response: None.

AMQ8410   Parser Error.
Explanation: The MQSC Parser has an internal error.
User Response: None.

AMQ8412   Numeric Range Error.
Explanation: The value assigned to an MQSC command keyword is out of the permitted range.
User Response: None.

AMQ8413   String Length Error.
Explanation: A string assigned to an MQSC keyword is either NULL, or longer than the maximum permitted for that keyword.
User Response: None.

AMQ8414   Display Channel details.
Explanation: The MQSC DISPLAY CHL command completed successfully, and details follow this message.
User Response: None.

AMQ8415   MQSeries commands are active.
Explanation: The MQSC PING QMGR command completed successfully.
User Response: None.

AMQ8416   MQSC timed out waiting for a response from the command server.
Explanation: MQSC did not receive a response message from the remote command server in the time specified.
User Response: None.

AMQ8417   Display Channel Status details.
Explanation: The MQSC DISPLAY CHANNEL STATUS command completed successfully, and details follow this message.
User Response: None.

AMQ8418   &1 command responses received.
Explanation: Running in queued mode, &1 command responses were received from the remote command server.
User Response: None.

AMQ8419   The Queue is already in the DCE cell.
Explanation: The Queue is already in the cell, that is, its SCOPE attribute is already CELL.
User Response: None.

AMQ8420   Channel Status not found.
Explanation: No status was found for the specified channel(s).
User Response: None.

AMQ8421   A required keyword was not specified.
Explanation: A keyword required in this command was not specified.
User Response: None.

AMQ8422   MQSC found the following response to a previous command on the reply q :-
Explanation: MQSC found additional command responses on the reply q. They will fill follow this message.
User Response: None.

AMQ8423   Cell Directory not available.
Explanation: The DCE cell directory is not available, so the requested operation has failed.
User Response: None.

AMQ8424   Error detected in a name keyword.
Explanation: A keyword in an MQSC command contained a name string which was not valid. This may be because it contained characters which are not accepted in MQ names. Typical keywords which can produce this error are QLOCAL (and the other q types), CHANNEL, XMITQ, INITQ, MCANAME etc.
User Response: None.

AMQ8425   Attribute value error .
Explanation: A keyword in an MQSC command contained a value that was not valid.
User Response: None.

AMQ8426   Valid MQSC commands are:
Explanation: The text shows valid MQSC commands.
User Response: None.

AMQ8427   Valid syntax for the MQSC command:
Explanation: The text shown is the valid syntax for the MQSC command.
User Response: None.

AMQ8428   TYPE Keyword has already been specified.
Explanation: The TYPE has already been specified after the DISPLAY verb, for example DISPLAY QUEUE(*) type(QLOCAL) type(QALIAS).
User Response: Delete the second TYPE keyword and run the command again.

AMQ8430   **Remote queue manager name is unknown.**

Explanation: The Remote queue manager name is not known to this queue manager. Check that

User Response: Create a transmission queue of the same name as the remote queue manager if one does not exist.

AMQ8431   **Transmission queue does not exist**

Explanation: The transmission queue does not exist on this queue manager.

User Response: None.

AMQ8432   **You are not allowed to set both the REPOS and REPOSNL fields.**

Explanation: An attempt to set both the REPOS and REPOSNL fields has been made. Only one of these fields can have a value other than blank. Both of the fields may be blank.

User Response: None.

AMQ8433   **You are not allowed to set both the CLUSTER and CLUSNL fields.**

Explanation: An attempt to set both the CLUSTER and CLUSNL fields has been made. Only one of these fields can have a value other than blank. Both of the fields may be blank.

User Response: None.

AMQ8434   **The repository is unavailable.**

Explanation: The repository is unavailable and the data cannot be accessed. Stop and restart the queue manager.

User Response: None.

AMQ8441   **Display Cluster Queue Manager details.**

Explanation: The MQSC DISPLAY CLUSQMGR command completed successfully, and details follow this message.

User Response: None.

AMQ8442   **You are not allowed to set USAGE to XMITQ with either the CLUSTER or CLUSNL fields set.**

Explanation: An attempt has been made to set USAGE to XMITQ when the CLUSTER or CLUSNL field has a value. Change the value of USAGE, or the CLUSTER or CLUSNL fields, and try the command again.

User Response: None.

AMQ8442   **USAGE can not be set to XMITQ with either the CLUSTER or CLUSNL fields set.**

Explanation: An attempt has been made to set USAGE to XMITQ when the CLUSTER or CLUSNL field has a value. Change the value of USAGE, or set the

CLUSTER and CLUSNL fields to blank, and try the command again.

User Response: None.

AMQ8443   **Only the CLUSTER or CLUSNL field may have a value.**

Explanation: An attempt has been made to set both CLUSTER and CLUSNL fields. One and only one of the fields may have a value, the other field must be blank. Change the value of one of the fields to blank and try the command again.

User Response: None.

AMQ8444   **The CLUSTER or CLUSNL fields must have a value.**

Explanation: Both the CLUSTER and CLUSNL fields are blank. One and only one of the fields may be blank, the other field must be a value. Change one of the fields from blank to a value and try the command again.

User Response: None.

AMQ8445   **Program cannot open queue manager object.**

Explanation: An attempt to open a queue manager object has failed.

User Response: See the previously listed messages in the job log.

AMQ8446   **Channel is currently active.**

Explanation: The requested operation failed because the channel is currently active.

User Response: See the previously listed messages in the job log.

AMQ8447   **Requested operation on channel '&5' not valid for this channel type.**

Explanation: The operation requested cannot be performed because channel '&5' is not of a suitable type. For example, only sender, server and cluster-sender channels can be resolved.

User Response: Check that the correct operation was requested. If it was, check that the correct channel name was specified.

AMQ8448   **Channel '&5' is not running.**

Explanation: A request to end channel '&5' has failed because the channel is not running.

User Response: Check that the correct operation was requested. If it was, check that the correct channel name was specified.

AMQ8449   **Queue '&5' inhibited for MQGET.**

Explanation: An MQGET failed because the queue '&5' had been previously inhibited for MQGET.

User Response: None.

AMQ8498   **Starting MQSeries Commands.**
Explanation: The MQSC script contains &1 commands.
User Response: None.

AMQ8499   **Usage: runmqsc [-e] [-v] [-w WaitTime [-x]] QMgrName**
Explanation: None.
User Response: None.

AMQ8500   **MQSeries Display MQ Files**
Explanation:
User Response:

AMQ8501   **Common services initialization failed with return code &1.**
Explanation: A request by the command server to initialize common services failed with return code &1.
User Response: None.

AMQ8502   **Connect shared memory failed with return code &1.**
Explanation: A request by the command server to connect shared memory failed with return code &1.
User Response: None.

AMQ8503   **Post event semaphore failed with return code &1.**
Explanation: A request by the command server to post an event semaphore failed with return code &1.
User Response: None.

AMQ8504   **Command server MQINQ failed with reason code &1.**
Explanation: An MQINQ request by the command server, for the MQSeries queue &3, failed with reason code &1.
User Response: None.

AMQ8505   **Reallocate memory failed with return code &1.**
Explanation: A request by the command server to reallocate memory failed with return code &1.
User Response: None.

AMQ8506   **Command server MQGET failed with reason code &1.**
Explanation: An MQGET request by the command server, for the MQSeries queue &3, failed with reason code &1.
User Response: None.

AMQ8507   **Command server MQPUT1 request for an undelivered message failed with reason code &1.**
Explanation: An attempt by the command server to put a message to the dead-letter queue, using MQPUT1, failed with reason code &1. The MQDLH reason code was &2.
User Response: None.

AMQ8508   **Queue Manager Delete Object List failed with return code &1.**
Explanation: A request by the command server to delete a queue manager object list failed with return code &1.
User Response: None.

AMQ8509   **Command server MQCLOSE reply-to queue failed with reason code &1.**
Explanation: An MQCLOSE request by the command server for the reply-to queue failed with reason code &1.
User Response: None.

AMQ8510   **Command server queue is open, try again later.**

AMQ8511   **Usage: strmqcsv [QMgrName]**

AMQ8512   **Usage: endmqcsv [-c | -i] QMgrName**

AMQ8513   **Usage: dspmqcsv [QMgrName]**

AMQ8514   **No response received after &1 seconds.**
Explanation: The command server has not reported the status of running, to the start request, before the timeout of &1 seconds was reached.
User Response: None.

AMQ8549   **Total string length exceeds the maximum value of 999 characters.**
Explanation: The total length of a channel exit string is 999 characters. The string list assigned to an MQSC keyword is longer than the maximum value of 999 characters permitted for that keyword.
User Response: None.

AMQ8550   **Display namelist details.**
Explanation: The MQSC DISPLAY NAMELIST command completed successfully, and details follow this message.
User Response: None.

AMQ8551   MQSeries namelist changed.
Explanation: MQSeries namelist &5 changed.
User Response: None.

AMQ8552   MQSeries namelist created.
Explanation: MQSeries namelist &5 created.
User Response: None.

AMQ8553   MQSeries namelist deleted.
Explanation: MQSeries namelist &5 deleted.
User Response: None.

AMQ8554   String List String Count Error.
Explanation: The number of strings within the stringlist is greater than the maximum number allowed for the keyword. Reduce the number of strings within the list and try the command again.
User Response: None.

AMQ8555   String List String Length Error.
Explanation: A string in a string list assigned to a keyword is longer than the maximum permitted for that keyword.
User Response: None.

AMQ8556   RESUME QUEUE MANAGER accepted.
Explanation: The RESUME QUEUE MANAGER command has been accepted for processing. The command will be sent to the repository which will process the command and notify all other repositories that this queue manager is now back in the cluster.
User Response: None.

AMQ8557   SUSPEND QUEUE MANAGER accepted.
Explanation: The SUSPEND QUEUE MANAGER command has been accepted for processing. The command will be sent to the repository which will process the command and notify all other repositories that this queue manager is leaving the cluster.
User Response: None.

AMQ8558   REFRESH CLUSTER accepted.
Explanation: The REFRESH CLUSTER command has been accepted for processing. The command will be sent to the Repository which will process the command and notify all other repositories that the Cluster needs refreshing.
User Response: None.

AMQ8559   RESET CLUSTER accepted.
Explanation: The RESET CLUSTER command has been accepted for processing. The command will be sent to the Repository which will process the command and notify all other repositories that the Cluster needs resetting.

User Response: None.

AMQ8560   MQSeries security cache refreshed.
Explanation: The MQSeries Object Authority Manager security cache has been refreshed.
User Response: None.

AMQ8561   Domain controller unavailable.
Explanation: MQSeries was unable to contact the domain controller to obtain information for user '&3'.
User Response: Ensure that a domain controller for the domain on which user '&3' is defined is available. Alternatively, if you are using a computer which is not currently connected to the network and have logged on using a domain user ID, you may wish to log on using a local user ID instead.

AMQ8601   MQSeries trigger monitor started.
Explanation: The MQSeries trigger monitor has been started.
User Response: None.

AMQ8602   MQSeries trigger monitor ended.
Explanation: The MQSeries trigger monitor has ended.
User Response: None.

AMQ8603   Usage: runmqtrm [-m QMgrName] [-q InitQ]
Explanation: None.
User Response: None.

AMQ8604   Use of MQSeries trigger monitor not authorized.
Explanation: The MQSeries trigger monitor cannot be run due to lack of authority to the requested queue manager or initiation queue.
User Response: Obtain the necessary authority from your security officer or MQSeries administrator. Then try the command again.

AMQ8605   Queue manager not available to the MQSeries trigger monitor
Explanation: The queue manager specified for the trigger monitor does not exist, or is not active.
User Response: Check that you named the correct queue manager. Ask your systems administrator to start it, if it is not active. Then try the command again.

AMQ8606   Insufficient storage available for the MQSeries trigger monitor.
Explanation: There was insufficient storage available for the MQSeries trigger monitor to run.
User Response: Free some storage and then try the command again.

AMQ8607 MQSeries trigger monitor connection failed.

Explanation: The trigger monitor's connection to the requested queue manager failed because of MQI reason code &1 from MQCONN.

User Response: Consult your systems administrator about the state of the queue manager.

AMQ8608 MQSeries trigger monitor connection broken.

Explanation: The connection to the queue manager failed while the trigger monitor was running. This may be caused by an endmqm command being issued by another user, or by an MQSeries queue manager error.

User Response: Consult your systems administrator about the state of the queue manager.

AMQ8609 Initiation queue missing or wrong type

Explanation: The named initiation queue could not be found; or the queue type is not correct for an initiation queue.

User Response: Check that the named queue exists, and is a local queue, or that the named queue is an alias for a local queue which exists.

AMQ8610 Initiation queue in use

Explanation: The MQSeries trigger monitor could not open the initiation queue because the queue is open for exclusive use by another application.

User Response: Wait until the queue is no longer in use, and try the command again.

AMQ8611 Initiation queue could not be opened.

Explanation: The MQSeries trigger monitor could not open the initiation queue; reason code &1 was returned from MQOPEN.

User Response: Consult your systems administrator.

AMQ8612 Waiting for a trigger message

Explanation: The MQSeries trigger monitor is waiting for a message to arrive on the initiation queue.

User Response: None.

AMQ8613 Initiation queue changed or deleted

Explanation: The MQSeries trigger monitor is unable to continue because the initiation queue has been deleted or changed since it was opened.

User Response: Retry the command.

AMQ8614 Initiation queue not enabled for input.

Explanation: The MQSeries trigger monitor cannot read from the initiation queue because input is not enabled.

User Response: Ask your systems administrator to enable the queue for input.

AMQ8615 MQSeries trigger monitor failed to get message.

Explanation: The MQSeries trigger monitor failed because of MQI reason code &1 from MQGET.

User Response: Consult your systems administrator.

AMQ8616 End of application trigger.

Explanation: The action to trigger an application has been completed.

User Response: None.

AMQ8617 Not a valid trigger message.

Explanation: The MQSeries trigger monitor received a message that is not recognized as

User Response: Consult your systems administrator.

AMQ8618 Error starting triggered application.

Explanation: An error was detected when trying to start the application identified in a trigger message.

User Response: Check that the application the trigger monitor was trying to start is available.

AMQ8619 Application type &1 not supported.

Explanation: A trigger message was received which specifies application type &1; the trigger monitor does not support this type.

User Response: Use an alternative trigger monitor for this initiation queue.

AMQ8620 Trigger message with warning &1

Explanation: The trigger monitor received a message with a warning. For example, it may have been truncated or it could not be converted to the trigger monitor's data representation. The reason code for the warning is &1.

User Response: None.

AMQ8621 Usage: runmqtmc [-m QMgrName] [-q InitQ]

Explanation: None.

User Response: None.

AMQ8622 Usage: CICS-Transaction-Name [MQTMC2 structure]

Explanation: None.

User Response: None.

AMQ8630 Notes agent not found.

Explanation: The agent &3 was not found in the Notes database &4.

User Response: Check that the agent exists in the Notes database and that its name is correctly specified.

AMQ8701   Usage: rcdmqimg [-z] [-l] [-m QMgrName] -t ObjType [GenericObjName]

Explanation: None.
User Response: None.

AMQ8702   Usage: rcrmqobj [-z] [-m QMgrName] -t ObjType [GenericObjName]

Explanation: None.
User Response: None.

AMQ8703   Usage: dspmqfls [-m QMgrName] [-t ObjType] GenericObjName

Explanation: None.
User Response: None.

AMQ8708   Dead-letter queue handler started to process INPUTQ(&3).

Explanation: The dead-letter queue handler (runmqdlq) has been started and has parsed the input file without detecting any errors and is about to start processing the queue identified in the message.
User Response: None.

AMQ8709   Dead-letter queue handler ending.

Explanation: The dead-letter queue handler (runmqdlq) is ending because the WAIT interval has expired and there are no messages on the dead-letter queue, or because the queue manager is shutting down, or because the dead-letter queue handler has detected an error. If the dead-letter queue handler has detected an error, an earlier message will have identified the error.
User Response: None.

AMQ8710   Usage: runmqdlq [QName[QMgrName]].

Explanation: Syntax for the usage of runmqdlq.
User Response: None.

AMQ8721   Dead-letter queue message not prefixed by a valid MQDLH.

Explanation: The dead-letter queue handler (runmqdlq) retrieved a message from the nominated dead-letter queue, but the message was not prefixed by a recognizable MQDLH. This typically occurs because an application is writing directly to the dead-letter queue but is not prefixing messages with a valid MQDLH. The message is left on the dead-letter queue and the dead-letter queue handler continues to process the dead-letter queue. Each time the dead-letter queue handler repositions itself to a position before this message to process messages that could not be processed on a previous scan it will reprocess the failing message and will consequently re-issue this message.
User Response: Remove the invalid message from the dead-letter queue. Do not write messages to the dead-letter queue unless they have been prefixed by a valid MQDLH. If you require a dead-letter queue handler that can process messages not prefixed by a valid MQDLH, you must change the sample program called amqsdlq to cater for your needs.

AMQ8722   Dead-letter queue handler unable to put message: Rule &1 Reason &2.

Explanation: This message is produced by the dead-letter queue handler when it is requested to redirect a message to another queue but is unable to do so. If the reason that the redirect fails is the same as the reason the message was put to the dead-letter queue then it is assumed that no new error has occurred and no message is produced. The retry count for the message will be incremented and the dead-letter queue handler will continue.
User Response: Investigate why the dead-letter queue handler was unable to put the message to the dead-letter queue. The line number of the rule used to determine the action for the message should be used to help identify to which queue the dead-letter queue handler attempted to PUT the message.

AMQ8741   Unable to connect to queue manager.

Explanation: The dead-letter queue handler (runmqdlq) could not connect to queue manager &3. This message is typically issued when the requested queue manager has not been started or is quiescing, or if the process does not have sufficient authority. The completion code (&1) and the reason (&2) can be used to identify the error. The dead-letter queue handler ends.
User Response: Take appropriate action based upon the completion code and reason.

AMQ8742   Unable to open queue manager: CompCode = &1 Reason = &2.

Explanation: The dead-letter queue handler (runmqdlq) could not open the queue manager object. This message is typically issued because of a resource shortage or because the process does not have sufficient authority. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.
User Response: Take appropriate action based upon the completion code and reason.

AMQ8743   Unable to inquire on queue manager: CompCode = &1 Reason = &2.

Explanation: The dead-letter queue handler (runmqdlq) could not inquire on the queue manager. This message is typically issued because of a resource shortage or because the queue manager is ending. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.
User Response: Take appropriate action based upon the completion code and reason.

AMQ8744   **Unable to close queue manager:**
                CompCode = &1 Reason = &2.
Explanation: The dead-letter queue handler
(runmqdlq) could not close the queue manager. This
message is typically issued because of a resource
shortage or because the queue manager is ending. The
completion code and the reason can be used to identify
the error. The dead-letter queue handler ends.
User Response: Take appropriate action based upon
the completion code and reason.

AMQ8745   **Unable to open dead-letter queue for**
                browse.
Explanation: The dead-letter queue handler
(runmqdlq) could not open the dead-letter queue &3
for browsing. This message is typically issued because
another process has opened the dead-letter queue for
exclusive access, or because an invalid dead-letter
queue name was specified. Other possible reasons
include resource shortages or insufficient authority. The
completion code(&1) and the reason(&2) can be used to
identify the error. The dead-letter queue handler ends.
User Response: Take appropriate action based upon
the completion code and reason.

AMQ8746   **Unable to close dead-letter queue:**
                CompCode = &1 Reason = &2.
Explanation: The dead-letter queue handler
(runmqdlq) could not close the dead-letter queue. This
message is typically issued because of a resource
shortage or because the queue manager is ending. The
completion code and the reason can be used to identify
the error. The dead-letter queue handler ends.
User Response: Take appropriate action based upon
the completion code and reason.

AMQ8747   **Integer parameter outside permissible**
                range.
Explanation: The integer parameter (&2) supplied to
the dead-letter handler was outside of the valid range
for &3 on line &1.
User Response: Correct the input data and restart the
dead-letter queue handler.

AMQ8748   **Unable to get message from dead-letter**
                queue: CompCode = &1 Reason = &2.
Explanation: The dead-letter queue handler
(runmqdlq) could not get the next message from the
dead-letter queue. This message is typically issued
because of the queue manager ending, a resource
problem, or another process having deleted the
dead-letter queue. The completion code and the reason
can be used to identify the error. The dead-letter queue
handler ends.
User Response: Take appropriate action based upon
the completion code and reason.

AMQ8749   **Unable to commit/backout action on**
                dead-letter queue: CompCode = &1
                Reason = &2.
Explanation: The dead-letter queue handler
(runmqdlq) was unable to commit or backout an
update to the dead-letter queue. This message is
typically issued because of the queue manager ending,
or because of a resource shortage. If the queue manager
has ended, the update to the dead-letter queue (and
any associated updates) will be backed out when the
queue manager restarts. If the problem was due to a
resource problem then the updates will be backed out
when the dead-letter queue handler terminates. The
completion code and the reason can be used to identify
the error. The dead-letter queue handler ends.
User Response: Take appropriate action based upon
the completion code and reason.

AMQ8750   **No valid input provided to runmqdlq.**
Explanation: Either no input was provided to
runmqdlq, or the input to runmqdlq contained no valid
message templates. If input was provided to runmqdlq
but was found to be invalid, earlier messages will have
been produced explaining the cause of the error. The
dead-letter queue handler will ends.
User Response: Correct the input data and restart the
dead-letter queue handler.

AMQ8751   **Unable to obtain private storage.**
Explanation: The dead-letter queue handler
(runmqdlq) was unable to obtain private storage. This
problem would typically arise as a result of some more
global problem. For example if there is a persistent
problem that is causing messages to be written to the
DLQ and the same problem (for example queue full) is
preventing the dead-letter queue handler from taking
the requested action with the message, it is necessary
for the dead-letter queue handler to maintain a large
amount of state data to remember the retry counts
associated with each message, or if the dead-letter
queue contains a large number of messages and the
rules table has directed the dead-letter queue handler
to ignore the messages.
User Response: Investigate if some more global
problem exists, and if the dead-letter queue contains a
large number of messages. If the problem persists
contact your support center.

AMQ8752   **Parameter(&3) exceeds maximum length**
                on line &1.
Explanation:
User Response: Correct the input data and restart the
dead-letter queue handler.

AMQ8753   **Duplicate parameter(&3) found on line &1.**

Explanation: Two or more parameters of the same type were supplied on a single input line to the dead-letter queue handler.

User Response: Correct the input and restart the dead-letter queue handler.

AMQ8756   **Error detected releasing private storage.**

Explanation: The dead-letter queue handler (runmqdlq) was informed of an error while attempting to release an area of private storage. The dead-letter queue handler ends.

User Response: This message should be preceded by a message or FFST information from the internal routine that detected the error. Take the action associated with the earlier error information.

AMQ8757   **Integer parameter(&3) outside permissible range on line &1.**

Explanation: An integer supplied as input to the dead-letter handler was outside of the valid range of integers supported by the dead-letter queue handler.

User Response: Correct the input data and restart the dead-letter queue handler.

AMQ8758   **&1 errors detected in input to runmqdlq.**

Explanation: One or more errors have been detected in the input to the dead-letter queue handler(runmqdlq). Error messages will have been generated for each of these errors. The dead-letter queue handler ends.

User Response: Correct the input data and restart the dead-letter queue handler.

AMQ8759   **Invalid combination of parameters to dead-letter queue handler on line &1.**

Explanation: An invalid combination of input parameters has been supplied to the dead-letter queue handler. Possible causes are: no ACTION specified, ACTION(FWD) but no FWDQ specified, HEADER(YES|NO) specified without ACTION(FWD).

User Response: Correct the input data and restart the dead-letter queue handler.

AMQ8760   **Unexpected failure while initializing process: Reason = &1.**

Explanation: The dead-letter queue handler (runmqdlq) could not perform basic initialization required to use MQ services because of an unforeseen error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8761   **Unexpected failure while connecting to queue manager: CompCode = &1 Reason = &2.**

Explanation: The dead-letter queue handler (runmqdlq) could not connect to the requested queue manager because of an unforeseen error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8762   **Unexpected error while attempting to open queue manager: CompCode = &1 Reason = &2.**

Explanation: The dead-letter queue handler (runmqdlq) could not open the queue manager because of an unforeseen error. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8763   **Unexpected error while inquiring on queue manager: CompCode = &1 Reason = &2.**

Explanation: The dead letter queue handler (runmqdlq) could not inquire on the queue manager because of an unforeseen error. The completion code and the reason can be used to identify the error. The dead letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8764   **Unexpected error while attempting to close queue manager: CompCode = &1 Reason = &2.**

Explanation: The dead-letter queue handler (runmqdlq) could not close the queue manager because of an unforeseen error. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8765   Unexpected failure while opening dead-letter queue for browse: CompCode = &1 Reason = &2.

Explanation: The dead-letter queue handler (runmqdlq) could not open the dead-letter queue for browsing because of an unforeseen error. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8766   Unexpected error while closing dead-letter queue: CompCode = &1 Reason = &2.

Explanation: The dead-letter queue handler (runmqdlq) could not close the dead-letter queue because of an unforeseen error. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8767   Unexpected error while getting message from dead-letter queue: CompCode = &1 Reason = &2.

Explanation: The dead-letter queue handler (runmqdlq) could not get the next message from the dead-letter queue because of an unforeseen error. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8768   Unexpected error committing/backing out action on dead-letter queue: CompCode = &1 Reason = &2.

Explanation: The dead-letter queue handler (runmqdlq) was unable to either commit or backout an update to the dead-letter queue because of an unforeseen error. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8769   Unable to disconnect from queue manager: CompCode = &1 Reason = &2.

Explanation: The dead-letter queue handler (runmqdlq) was unable to disconnect from the queue manager because of an unexpected error. The completion code and the reason can be used to identify the error. The dead-letter queue handler ends.

User Response: Use the standard facilities supplied with your system to record the problem identifier and to save the generated output files. Contact your support center. Do not discard these files until the problem has been resolved.

AMQ8880   Storage class error

Explanation: An error occurred when attempting to access a storage class.

User Response:

AMQ8899   Before using NetBIOS, you need to define a Local Name in this system.

Explanation: This can be done either 1. - By setting the MQNAME environment variable or 2. - By setting the LOCALNAME parameter in the NetBIOS: stanza of configuration data NetBIOS: LocalName=xxxxxx (see the books MQSeries Intercommunication, and MQSeries Clients for details) To continue with this verification you MUST update the configuration data for this qmgr This local name must be different from the listener name.

User Response: None.

Chapter 8. AMQ9000-AMQ9999 Remote messages

> **Reading variables in the message**
> Some messages display text or numbers that vary according to the
> circumstances giving rise to the message; these are known as *message variables*.
> The message variables are indicated in this book by the use of the '&' symbol
> and a number: &1, &2, and so on.
>
> In some cases a message may have variables in the Explanation or User
> action. Find the values of the message variables by looking in the error log.
> The complete message, including the Explanation and the User action, is
> recorded there.

AMQ9001   Channel program ended normally.
Explanation: Channel program '&3' ended normally.
User Response: None.

AMQ9002   Channel program started.
Explanation: Channel program '&3' started.
User Response: None.

AMQ9181   The response set by the exit is not valid.
Explanation: The user exit '&3' returned a response
code '&1' that is not valid in the ExitResponse field of
the channel exit parameters (MQCXP). Message
AMQ9190 is issued giving more details, and the
channel stops.
User Response: Investigate why the user exit program
set a response code that is not valid.

**AMQ9182   The secondary response set by the exit
            is not valid.**
Explanation: The user exit '&3' returned a secondary
response code '&1' in the ExitResponse2 field of the
channel exit parameters (MQCXP) that is not valid.
Message AMQ9190 is issued giving more details, and
the channel stops.
User Response: Investigate why the user exit program
set a secondary response code that is not valid.

**AMQ9184   The exit buffer address set by the exit is
            not valid.**
Explanation: The user exit '&3' returned an address
'&1' for the exit buffer that is not valid, when the
secondary response code in the ExitResponse2 field of
the channel exit parameters (MQCXP) is set to
MQXR2_USE_EXIT_BUFFER. Message AMQ9190 is
issued giving more details, and the channel stops.
User Response: Investigate why the user exit program
set an exit buffer address that is not valid. The most
likely cause is the failure to set a value, so that the
value is 0.

**AMQ9185   The exit space set by the exit is not
            valid.**
Explanation: The user exit '&3' returned an exit space
value '&1' that is not valid in the ExitSpace field of the
channel exit parameters (MQCXP). Message AMQ9190
is issued giving more details, and the channel stops.
User Response: Investigate why the user exit program
set an exit space value that is not valid. Correct the
error.

**AMQ9186   Too much exit space reserved by send
            exits.**
Explanation: At exit initialization the send exits in the
send exit chain for channel &3 returned values in the
ExitSpace field of the channel exit parameters
(MQCXP). The total of these ExitSpace values is &1.
The maximum number of bytes that can be sent in a
single transmission is &2. Room must be left for at least
1024 bytes of message data in each transmission. So too
much exit space has been reserved by the send exits.
The channel stops.
User Response: Investigate why the send exit
programs set exit space values that are too large.
Correct the error.

**AMQ9189   The data length set by the exit is not
            valid.**
Explanation: The user exit '&3' returned a data length
value '&1' that was not greater than zero. Message
AMQ9190 is issued giving more details, and the
channel stops.
User Response: Investigate why the user exit program
set a data length that is not valid.

**AMQ9190   Channel stopping because of an error in
            the exit.**
Explanation: The user exit '&3', invoked for channel
'&4' with id '&1' and reason '&2', returned values that
are not valid, as reported in the preceding messages.
The channel stops.

85

User Response: Investigate why the user exit program set values that are not valid.

| **AMQ9195   Data length larger than maximum segment length.**
| **Explanation:** The data length '&1' set by send exit '&3' is larger than the maximum segment length (&2). The maximum segment length is the maximum number of bytes that can be sent in a single transmission minus the user exit space required by all the send exits subsequent to the current one in the send exit chain. Message AMQ9190 is issued giving more details, and the channel stops.
| **User Response:** Investigate why the user exit program set a data length that is not valid. Correct the error.

| **AMQ9196   Data length is larger than the agent buffer length.**
| **Explanation:** The data length '&1' set by exit '&3' is larger than the agent buffer length. The user exit returned data in the supplied agent buffer, but the length specified is greater than the length of the buffer. Message AMQ9190 is issued giving more details, and the channel stops.
| **User Response:** Investigate why the user exit program set a data length that is not valid. Correct the error.

AMQ9197   Data length is larger than the exit buffer length.
Explanation: The data length '&1' set by exit '&3' is larger than the exit buffer length. The user exit returned data in the supplied exit buffer, but the length specified is greater than the length of the buffer. Message AMQ9190 is issued giving more details, and the channel stops.
User Response: Investigate why the user exit program set a data length that is not valid.

| **AMQ9201   Allocate failed to host '&3'.**
| **Explanation:** The attempt to allocate a conversation using &4 to host &3 was not successful.
| **User Response:** The error may be due to an incorrect entry in the &4 parameters contained in the channel definition to host &3. Correct the error and try again. If the error persists, record the error values and contact your systems administrator. The return code from the &4&5 call was &1 (X'&2'). It may be possible that the listening program at host &3 is not running. If this is the case, perform the relevant operations to start the listening program for protocol &4 and try again. Another possible explanation is that there is a message sequence number mismatch between the two ends of the channel. Use the RESET CHANNEL command to bring the sequence numbers back into alignment.

AMQ9202   Remote host '&3' not available, retry later.
Explanation: The attempt to allocate a conversation using &4 to host '&3' was not successful. However the error may be a transitory one and it may be possible to successfully allocate a &4 conversation later.
User Response: Try the connection again later. If the failure persists, record the error values and contact your systems administrator. The return code from &4 is &1 (X'&2'). The reason for the failure may be that this host cannot reach the destination host. It may also be possible that the listening program at host '&3' was not running. If this is the case, perform the relevant operations to start the &4 listening program, and try again.

| **AMQ9203   A configuration error for &4 occurred.**
| **Explanation:** Error in configuration for communications to host &3. Allocation of a &4 conversation to host &3 was not possible.
| **User Response:** The configuration error may be one of the following:
| • If the communications protocol is LU 6.2, it may be that one of the transmission parameters (Mode, or TP Name) is incorrect. Correct the error and try again. The mode name should be the same as the mode defined on host &3. The TP name on &3 should be defined.
| • If the communications protocol is LU 6.2, it may be that an LU 6.2 session has not been established. Contact your systems administrator.
| • If the communications protocol is TCP/IP, it may be that the host name specified is incorrect. Correct the error and try again.
| • If the communications protocol is TCP/IP, it may be that the host name specified cannot be resolved to a network address. The host name may not be in the nameserver.

| The return code from the &4&5 call was &1 (X'&2').

| Record the error values and tell the system administrator.

| **AMQ9204   Connection to host &3 rejected.**
| **Explanation:** Connection to host &3 over &4 was rejected.
| **User Response:** The remote system may not be configured to allow connections from this host. Check the &4 listener program has been started on host &3.

| If the conversation uses LU 6.2, it is possible that either the User ID or Password supplied to the remote host is incorrect.

| If the conversation uses TCP/IP, it is possible that the remote host does not recognize the local host as a valid host.

| The return code from the &4&5 call was &1 X('&2').

Record the error values and tell the systems administrator.

AMQ9205   The host name supplied is not valid.
Explanation: The supplied &4 host name '&3' could not be resolved into a network address. Either the name server does not contain the host, or the name server was not available.
User Response: Check the &4 configuration on your host.

AMQ9206   Error sending data to host &3.
Explanation: An error occurred sending data over &4 to &3. This may be due to a communications failure.
User Response: The return code from the &4&5 call was &1 X('&2'). Record these values and tell your systems administrator.

AMQ9207   The data received from host '&3' is not valid.
Explanation: Incorrect data format received from host '&3' over &4. It may be that an unknown host is attempting to send data. An FFST file has been generated containing the invalid data received.
User Response: Tell the systems administrator.

AMQ9208   Error on receive from host &3.
Explanation: An error occurred receiving data from &3 over &4. This may be due to a communications failure.
User Response: The return code from the &4&5 call was &1 (X'&2'). Record these values and tell the systems administrator.

AMQ9209   Connection to host '&3' closed.
Explanation: An error occurred receiving data from '&3' over &4. The connection to the remote host has unexpectedly terminated.
User Response: Tell the systems administrator.

AMQ9210   Remote attachment failed.
Explanation: There was an incoming attachment from a remote host, but the local host could not complete the bind.
User Response: The return code from the &4&5 call was &1 (X'&2'). Record these values and tell the systems administrator who should check the &4 configuration.

AMQ9211   Error allocating storage.
Explanation: The program was unable to obtain enough storage.
User Response: Stop some programs which are using storage and retry the operation. If the problem persists contact your systems administrator.

AMQ9212   A TCP/IP socket could not be allocated.
Explanation: A TCP/IP socket could not be created, possibly because of a storage problem.
User Response: The return code from the &4&5 call was &1 (X'&2'). Try the program again. If the failure persists, record the error values and tell the systems administrator.

AMQ9213   A communications error for &4 occurred.
Explanation: An unexpected error occurred in communications.
User Response: The return code from the &4&5 call was &1 (X'&2'). Record these values and tell the systems administrator.

AMQ9214   Attempt to use an unsupported communications protocol.
Explanation: An attempt was made to use an unsupported communications protocol type &2.
User Response: Check the channel definition file. It may be that the communications protocol entered is not a currently supported one.

AMQ9215   Communications subsystem unavailable.
Explanation: An attempt was made to use the communications subsystem, but it has not been started.
User Response: Start the communications subsystem, and rerun the program.

AMQ9216   Usage: &3 [-m QMgrName] [-n TPName]
Explanation: Values passed to the responder channel program are not valid. The parameters that are not valid are as follows :-

&4

The -n parameter is not valid on Unix systems.

The responder channel program exits.
User Response: Correct the parameters passed to the channel program and retry the operation.

AMQ9217   The TCP/IP listener program could not be started.
Explanation: An attempt was made to start a new instance of the listener program, but the program was rejected.
User Response: The failure could be because either the subsystem has not been started (in this case you should start the subsystem), or there are too many programs waiting (in this case you should try to start the listener program later).

AMQ9218   The TCP/IP listener program could not bind to port number &1.
Explanation: An attempt to bind the TCP/IP socket to the listener port was unsuccessful.
User Response: The failure could be due to another program using the same port number. Record the return code &2 from the bind and tell the systems administrator.

AMQ9219   The TCP/IP listener program could not create a new connection for the incoming conversation.
Explanation: An attempt was made to create a new socket because an attach request was received, but an error occurred.
User Response: The failure may be transitory, try again later. If the problem persists, record the return code &1 and tell the systems administrator. It may be necessary to free some jobs, or restart the communications system.

AMQ9220   The &4 communications program could not be loaded.
Explanation: The attempt to load the &4 library or procedure '&3' failed with error code &1.
User Response: Either the library must be installed on the system or the environment changed to allow the program to locate it.

AMQ9221   Unsupported protocol was specified.
Explanation: The specified value of '&3' was not recognized as one of the protocols supported.
User Response: Correct the parameter and retry the operation.

AMQ9222   Cannot find the configuration file.
Explanation: The configuration file '&3' cannot be found. This file contains default definitions for communication parameters. Default values will be used.
User Response: None.

AMQ9223   Enter a protocol type.
Explanation: The operation you are performing requires that you enter the type of protocol.
User Response: Add the protocol parameter and retry the operation.

AMQ9224   Unexpected token detected.
Explanation: Keyword '&3' was read when a keyword was expected.
User Response: Correct the configuration data and retry the operation.

AMQ9225   File syntax error.
Explanation: A syntax error was detected while processing the configuration data.
User Response: Correct the problem and retry the operation.

AMQ9226   Usage: &3 [-m QMgrName] -t (TCP | UDP | LU62 | NETBIOS | SPX) [ProtocolOptions]
Explanation: Values passed to the listener program were invalid.

The parameter string passed to this program is as follows:

```
[-m QMgrName] ( -t TCP [-p Port] |
                -t UDP [-p Port] |
                -t LU62 [-n TPName] |
                -t NETBIOS [-l LocalName]
                        [-e Names] [-s Sessions]
                        [-o Commands] [-a Adaptor] |
                -t SPX [-x Socket] )
```

Default values will be used for parameters not supplied.
User Response: Correct the parameters passed to the listener program and retry the operation.

AMQ9227   &3 local host name not provided.
Explanation:
User Response: Add a local name to the configuration file and retry the operation.

AMQ9228   The &4 responder program could not be started.
Explanation: An attempt was made to start an instance of the responder program, but the program was rejected.
User Response: The failure could be because either the subsystem has not been started (in this case you should start the subsystem), or there are too many programs waiting (in this case you should try to start the responder program later). The &5 reason code was &1.

AMQ9229   The application has been ended.
Explanation: You have issued a request to end the application.
User Response: None.

AMQ9230   An unexpected &4 event occurred.
Explanation: During the processing of network events, an unexpected event &1 occurred.
User Response: None.

AMQ9231   The supplied parameter is not valid.
Explanation: The value of the &4 &5 parameter has
the value '&3'. This value has either not been specified
or has been specified incorrectly.
User Response: Check value of the &5 parameter and
correct it if necessary. If the fault persists, record the
return code (&1,&2) and &4 and tell the systems
administrator.

AMQ9232   No &3 specified
Explanation: The operation requires the specification
of the &3 field.
User Response: Specify the &3 and retry the
operation.

AMQ9233   Error creating Listener thread for &3.
Explanation: The process attempted to create a new
thread for an incoming connection. The most likely
cause of this problem is a shortage of an operating
system resource, for example, memory. Use any
previous FFSTs to determine the reason for the failure.
The MQSeries internal return code describing the
reason for the failure is &1.
User Response: Contact the systems administrator. If
the problem persists contact your IBM support center.

AMQ9235   The supplied Local LU was invalid.
Explanation: The &4 Local LU name '&3' was invalid.
User Response: Either the Local LU name was entered
incorrectly or it was not in the &4 communications
configuration. Correct the error and try again.

AMQ9236   The supplied Partner LU was invalid.
Explanation: The &4 Partner LU name '&3' was
invalid.
User Response: Either the Partner LU name was
entered incorrectly or it was not in the &4
communications configuration. Correct the error and
try again.

AMQ9237   A configuration error for &4 occurred.
Explanation: Allocation of a &4 conversation to host
'&3' was not possible. The configuration error may be
one of the following:

1. It may be that one of the transmission parameters
   (Mode, or TP Name) was incorrect. Correct the error
   and try again. The mode name should be the same
   as the mode defined on host &3. The TP name on
   &3 should be defined.

2. It may be that an LU 6.2 session has not been
   established. Contact your systems administrator.

The return code from &4 is &1 with associated &5 &2.
User Response: Record the error values and tell the
system administrator.

**AMQ9238   A communications error for &4
            occurred.**
Explanation: An unexpected error occurred in
communications.
User Response: The return code from the &4&3 call
was &1 with associated &5 &2.

**AMQ9239   Usage: &3 [-m QMgrName] -n TpName
            -g Gateway-name**
Explanation: Values passed to the listener program
were invalid. The parameter string passed to this
program is as follows, default values being used for
parameters not supplied: [-m QMgrName] -n TpName
-g Gateway-name
User Response: Correct the parameters passed to the
listener program and retry the operation.

AMQ9240   An SPX socket was already in use.
Explanation: The listener received return code &1
when attempting to open socket &2.
User Response: The specified socket is already in use
by another process. To use another socket, specify a
different socket on the command line to the runmqlsr
command, or update the default in the configuration
data.

AMQ9241   SPX is not available.
Explanation: MQSeries received return code &1 when
attempting to start SPX communications.
User Response: Ensure that IPX/SPX support is
installed on the machine and that it is started before
trying to start an MQSeries SPX channel.

AMQ9242   SPX resource problem.
Explanation: MQSeries received return code &1 when
attempting to start SPX communications, indicating a
resource problem.
User Response: Ensure that sufficient IPX/SPX
resources are available before commencing MQSeries
communications over IPX/SPX.

AMQ9243   The queue manager &3 does not exist.
Explanation: You tried to perform an action against a
queue manager that does not exist. You may have
specified the wrong queue manager name.
User Response: If you specified the wrong name,
correct the name and submit the command again. If the
queue manager does not exist, create the queue
manager and submit the command again.

**AMQ9245   Unable to obtain account details for
            channel MCA user ID.**
Explanation: MQSeries was unable to obtain the
account details for MCA user ID '&3'. This user ID was
the MCA user ID for channel '&4' on queue manager
'&5' and may have been defined in the channel

definition, or supplied either by a channel exit or by a client.
User Response: Ensure that the user ID is correct and that it is defined on the Windows local system, the local domain or on a trusted domain. For a domain user ID, ensure that all necessary domain controllers are available.

AMQ9401   Channel '&3' autodefined.
Explanation: Channel '&3' which did not previously exist has been autodefined.
User Response: None.

AMQ9402   Autodefinition exit for Channel '&3' failed to load.
Explanation: Autodefinition of Channel '&3' failed because '&4' would not load.
User Response: Ensure that the user exit is specified correctly in the queue manager definition, and that the user exit program is correct and available.

AMQ9403   Autodefinition of Channel '&3' suppressed by user exit.
Explanation: Autodefinition exit '&4' for Channel '&3' returned a failure code.
User Response: None.

AMQ9407   Cluster queue &3 is defined inconsistently.
Explanation: The definition of cluster queue &3 on the queue manager with UUID &4 has different DEFPRTY, DEFPSIST and/or DEFBIND values than the definition of the same cluster queue on the queue manager with UUID &5. Both definitions now exist in the local repository. All definitions of the same cluster queue should be identical. In particular, problems arise if your applications rely on a queue default value which is defined inconsistently to determine messaging behavior. This would apply, for instance, if the applications open a cluster queue with option MQOO_BIND_AS_Q_DEF. If different instances of the queue have different DEFBIND values the behavior of the message transfer differs depending on which instance of the queue is selected when it is opened. In general the instance selected varies across opens.
User Response: For each inconsistency decide which of the inconsistent values is the correct one. Alter the definitions of cluster queue &3 so that all definitions have correct DEFPRTY, DEFPSIST and DEFBIND values.

AMQ9408   BIND_ON_OPEN messages for channel &3 to dead-letter queue.
Explanation: The remote CLUSRCVR for channel &3 was deleted while undelivered BIND_ON_OPEN messages associated with that channel existed on the local SYSTEM.CLUSTER.TRANSMIT.QUEUE. These messages could not be allocated to another channel

because they were put BIND_ON_OPEN, but were very unlikely to ever flow along the channel with which they were associated as this has now been deleted. An attempt has therefore been made to move them from the transmission queue to the local dead-letter queue. The MQDLH reason is MQFB_BIND_OPEN_CLUSRCVR_DEL. Note that any internal MQSeries Clustering messages for the deleted channel will also have been removed from the SYSTEM.CLUSTER.TRANSMIT.QUEUE (these are discarded) so the current depth of the queue may have decreased by more than the number of user messages moved to the dead-letter queue.
User Response: Examine the contents of the dead-letter queue. Each message is contained in an MQDLH structure that includes the reason why it was written and where it was originally addressed. Also look at previous error messages to see if the attempt to put messages to the dead-letter queue failed.

AMQ9409   Repository manager ended abnormally.
Explanation: The repository manager ended abnormally.
User Response: Look at previous error messages for the repository manager in the error files to determine the cause of the failure.

AMQ9410   MQSeries repository manager started.
Explanation: The MQSeries repository manager has been started.
User Response: None.

AMQ9411   Repository manager ended normally.
Explanation: The repository manager ended normally.
User Response: None.

AMQ9412   Repository command received for '&3'.
Explanation: The repository manager received a command intended for some other queue manager, whose identifier is '&3'. The command was sent by the queue manager with identifier '&4'.
User Response: Check the channel and cluster definitions of the sending queue manager.

AMQ9413   Repository command format error, command code &1
Explanation: An internal error has occurred.
User Response: Collect the items listed in the 'Problem determination' section of the System Administration manual and contact your IBM support center.

AMQ9415   Repository command unexpected, command code &1, cluster object &3, sender &4
Explanation: An internal error has occurred.
User Response: Collect the items listed in the 'Problem determination' section of the System

Administration manual and contact your IBM support center.

AMQ9416   **Repository command processing error, RC=&2, command code &1, cluster object &3, sender &4.**

Explanation: An internal error has occurred.

User Response: Collect the items listed in the 'Problem determination' section of the System Administration manual and contact your IBM support center.

AMQ9417   **Manually defined CLUSSDR channels have been forcibly removed.**

Explanation: The administrator has asked for the queue manager '&3' to be deleted, or forcibly removed, but has not yet deleted the manually defined CLUSSDR channels to '&3'. The auto-defined channels to '&3' have been deleted, but '&3' continues to receive updates until the manually defined CLUSSDR channels have been deleted.

User Response: Delete the manually defined CLUSSDR channels to '&3'.

AMQ9418   **Only one repository for cluster &3.**

Explanation: The queue manager has received information about a cluster for which it is the only repository.

User Response: Alter the REPOS or REPOSNL attribute of the queue manager, that is to have the second full repository for the cluster, to specify the cluster name.

AMQ9419   **No cluster-receiver channels for cluster &3**

Explanation: The repository manager has received information about a cluster for which no cluster-receiver channels are known.

User Response: Define cluster-receiver channels for the cluster on the local queue manager.

AMQ9420   **No repositories for cluster &3.**

Explanation: The queue manager has received information about a cluster for which no repositories are known.

User Response: Alter the REPOS or REPOSNL attribute of the queue manager, that is to have a full repository for the cluster, to specify the cluster name.

AMQ9422   **Repository manager error, RC=&1**

Explanation: An internal error has occurred.

User Response: Collect the items listed in the 'Problem determination' section of the System Administration manual and contact your IBM support center.

AMQ9425   **An internal error has occurred.**

Explanation: Repository command merge error, command code &1, cluster object &3, sender &4.

User Response: Collect the items listed in the 'Problem determination' section of the System Administration manual and contact your IBM support center.

AMQ9426   **Repository command recipient unknown.**

Explanation: The repository manager tried to send a command to another queue manager using channel &4. The recipient queue manager, whose identifier is &3, could not be found. Command code &1.

User Response: Check the channel and cluster definitions of the sending and receiving queue managers.

AMQ9427   **CLUSSDR channel does not point to a repository queue manager.**

Explanation: A CLUSSDR channel must point to a queue manager that hosts repositories for all clusters of which the channel is a member. In addition, the CLUSRCVR for the channel must be a member of all the same clusters as the CLUSSDR channel. The queue manager pointed to by CLUSSDR channel &3 does not meet these criteria for cluster &4. The remote queue manager has a QMID of &5.

User Response: Check the definitions on the local and remote queue managers to ensure that the CLUSSDR channel points to a queue manager that hosts a repository for the cluster, and that the CLUSRCVR for the channel is a member of the cluster.

AMQ9428   **Unexpected publication of a cluster queue object received.**

Explanation: The local queue manager has received a publication of a cluster queue object from a remote queue manager on cluster &3. The local queue manager discards the request because it does not host a repository for cluster &3 and has not subscribed to the published object. The remote CLUSSDR channel used to access the local queue manager has a channel name of &4 and the remote queue manager has a QMID of &5.

User Response: Check the definitions on the local and remote queue managers to ensure that the CLUSSDR channel points to a repository queue manager for the cluster.

AMQ9429   **Unexpected publication of a cluster queue deletion received.**

Explanation: The local queue manager has received a publication of a cluster queue deletion from a remote queue manager on cluster &3. The local queue manager discards the request because it does not host a repository for cluster &3 and has not subscribed to the published object. The remote CLUSSDR channel used

to access the local queue manager has a channel name of &4 and the remote queue manager has a QMID of &5.

User Response: Check the definitions on the local and remote queue managers to ensure that the CLUSSDR channel points to a repository queue manager for the cluster.

AMQ9430   Unexpected cluster queue manager publication received.

Explanation: The local queue manager has received a cluster queue manager publication on cluster &3. The local queue manager should not have received the publication because it does not host a repository for cluster &3, it has not subscribed to information concerning the published object, and the published object does not match any of its CLUSSDRs. The queue manager that sent the publication to the local queue manager has QMID &4 (note that this is not necessarily the queue manager which originated the publication). CLUSSDR channel &5 was used to send the publication.

User Response: Check the CLUSSDR definition on the sending queue manager to ensure that it points to a repository queue manager for the cluster.

AMQ9431   Remote queue manager no longer hosts a repository for cluster

Explanation: The local queue manager has received a message from remote queue manager QMID &3 indicating that it no longer hosts a repository for cluster &4. CLUSSDR channel &5 is altered so that it can no longer be used to access queue manager &3 within cluster &4. If the local queue manager does not host a repository for cluster &4 the relevant subscriptions and publications are remade if possible.

User Response: None.

AMQ9432   Query received by a non-repository queue manager

Explanation: The local queue manager has received a query from a remote queue manager on cluster &3. The local queue manager discards the query because it does not host a repository for cluster &3. The remote CLUSSDR channel used to access the local queue manager has a channel name of &4 and the remote queue manager has a QMID of &5.

User Response: Check the definitions on the local and remote queue managers to ensure that the CLUSSDR channel points to a repository queue manager for the cluster.

AMQ9433   CLUSRCVR must be in the same cluster as its matching CLUSSDR.

Explanation: CLUSRCVR channel &3 is not defined as a member of cluster &4. The local queue manager has received a command that indicates that CLUSSDR channel &3 on the remote queue manager with QMID &5 is defined as a member of cluster &4.

User Response: Alter the CLUSRCVR or CLUSSDR definitions for channel &3, so that they are both members of the same cluster.

AMQ9434   Unrecognized message on &3.

Explanation: The repository manager found a message on one of its queues having, either a format that could not be recognized, or that did not come from a queue manager or repository manager. The message was put on the dead-letter queue.

User Response: Examine the message on the dead-letter queue to determine the originator of the message.

AMQ9435   Unable to put repository manager message.

Explanation: The repository manager tried to send a message to the SYSTEM.CLUSTER.COMMAND.QUEUE on another queue manager whose identifier is &3, but the MQPUT call was unsuccessful. MQCC=&1, MQRC=&2. Processing continues, but the repository information may be out of date.

User Response: Refer to the Application Programming Reference manual for information about MQCC &1 and MQRC &2. Check the channel and cluster definitions on the local and target queue managers, and ensure that the channels between them are running. When the problem is corrected, the repository information will normally be updated automatically. The REFRESH CLUSTER command can be used to ensure that the repository information is up to date.

AMQ9436   Unable to send repository manager message.

Explanation: The repository manager tried to send a message to the SYSTEM.CLUSTER.COMMAND.QUEUE on a queue manager that has the full repository for the specified cluster(&3), but the MQPUT call was unsuccessful. MQCC=&1, MQRC=&2. Processing continues, but repository information may be out of date.

User Response: Refer to the Application Programming Reference manual for information about MQCC &1 and MQRC &2. Check the channel and cluster definitions on the local and target queue managers, and ensure that the channels between them are running. When the problem is corrected, the repository information will normally be updated automatically. The REFRESH CLUSTER command can be used to ensure that the repository information is up to date.

AMQ9437   Unable to commit repository changes.

Explanation: The repository manager tried to commit some updates to the repository but was unsuccessful. Processing continues, but repository information may be out of date.

User Response: If this occurs when the repository manager is stopping, this message can be ignored,

because the repository information will normally be updated automatically when the repository manager is restarted. If there is an isolated occurrence at other times, use the REFRESH CLUSTER command to bring the repository information up to date. If the problem persists, contact your IBM support center for assistance.

AMQ9438   CONNAME could not be discovered for CLUSRCVR &3.
Explanation: TCP/IP CLUSRCVR &3 was validly specified with a blank or absent CONNAME parameter. However when the repository process, amqrrmfa, attempted to obtain the CONNAME (IP address) for itself it was unable to. If there is an existing matching CLUSRCVR object in the cache its CONNAME is used. The CONNAME used was &4.
User Response: Check the error log for a message arising from an associated TCP/IP call (gethostname, gethostbyname or inet_ntoa). Pass all the error information to your systems administrator.

AMQ9439   Repository corruption: bad CLQMGR object for channel &3.
Explanation: An internal error has occurred.
User Response: Collect the items listed in the 'Problem determination' chapter in the System Administration guide and contact your IBM support center.

AMQ9440   Reset command failed.
Explanation: Reset Cluster(&3) Qmname(&4) command failed. To issue this command, queue manager &5 must be a repository for cluster &3. Alter the queue manager attributes Repos, or Reposnl, to include cluster &3 and retry the command.
User Response: None.

AMQ9441   Reset command processed.
Explanation: The reset Cluster(&3) Qmname(&4) command has processed on this repository and &1 other queue managers have been sent notification.
User Response: None.

AMQ9442   Refresh Cluster command processed.
Explanation: The Refresh Cluster(&4) command caused &1 objects to be refreshed and &2 objects to be republished.
User Response: None.

AMQ9443   Suspend Qmgr Cluster command processed.
Explanation: The Suspend Qmgr Cluster command completed. &1 objects suspended.In the case of a name list the cluster name is the first name in the list.
User Response: None.

AMQ9444   Resume Qmgr Cluster command processed.
Explanation: The Resume Qmgr Cluster(&4) command completed. &1 objects resumed. In the case of a name list the cluster name is the first name in the list.
User Response: None.

AMQ9445   Error creating channel %3.
Explanation: Channel %4 tried to replace itself by creating channel %3. The attempt to create the channel was unsuccessful for the following reason: "%5". A previous message may give further information.
User Response: Rectify the problem which prevented successful creation of channel %3. Restart channel %4.

AMQ9446   Error deleting channel %3.
Explanation: Channel %3 tried to delete itself after creating channel %4 to replace it. The attempt to delete the channel was unsuccessful for the following reason: "%5".
User Response: If channel %3 still exists rectify the problem which prevented its deletion and then manually delete the channel.

AMQ9447   Unable to backout repository changes.
Explanation: Following an error, the repository manager tried to backout some updates to the repository, but was unsuccessful. The repository manager terminates.
User Response: If the repository manager subsequently restarts successfully, this message can be ignored. If the repository manager does not restart, contact your IBM support center for assistance.

AMQ9448   Repository manager stopping because of errors. Restart in &1 seconds.
Explanation: A severe error, as reported in the preceding messages, occurred during repository manager processing. The repository manager was unable to continue and terminates. The repository manager will try to restart after the specified interval.
User Response: Correct the problem reported in the preceding messages.

AMQ9449   Repository manager restarted.
Explanation: The repository manager restarted successfully following an error.
User Response: None.

AMQ9450   Usage: &3 [-m QMgrName] -f OutputFile [-v OutputFileVersion]
Explanation: Values passed to the channel table writer program were invalid.

The parameter string passed to this program is as follows:

[-m QMgrName] -f OutputFile [-v OutputFileVersion], where OutputFileVersion can be either 2 or 5 (5 is the default).

Default values will be used for parameters not supplied.
User Response: Correct the parameters passed to the channel table writer program and retry the operation.

AMQ9494   A protocol error was detected for channel '&3'.
Explanation: During communications with the remote queue manager, a TCP/IP read and receive call returned EINTR, indicating that it had been interrupted. Immediately after this the channel program detected a protocol error. The failure type was &1 with associated data of &2.
User Response: If you are running an AIX client you will avoid problems arising from EINTRs on TCP/IP reads, by writing your application so that system calls interrupted by signals are restarted. You must establish the signal handler with sigaction(2) and set the SA_RESTART flag in the sa_flags field of the new action structure. If you are running on a platform other than AIX, an AIX server, or an AIX client with an application that adheres to the restart guidelines provided above, contact the systems administrator who should examine the error logs to determine the cause of the failure.

AMQ9495   The CLWL exit '&3' is inconsistent with a dynamic cache.
Explanation: When the CLWL exit '&3' was called for the ExitReason MQXR_INIT, the value '&1' was returned in the ExitResponse2 field. This indicates the CLWL exit is incompatible with the Queue Manager cache type which is dynamic. Either change the Queue Manager cache type to static (using the Tuning Parameter, ClusterCacheType=STATIC) or rewrite the CLWL exit to be compatible with a dynamic cache. The CLWL exit has been suppressed.
User Response: None.

AMQ9498   The MQCD structure supplied was not valid.
Explanation: The value of the &3 parameter has the value '&4'. This value is invalid for the operation requested.
User Response: Change the parameter and retry the operation.

AMQ9499   An MQSeries listener will end shortly.
Explanation: One MQSeries listener detected in the system is scheduled for shutdown.
User Response: None.

AMQ9500   No Repository storage
Explanation: An operation failed because there was no storage available in the repository. An attempt was made to allocate &1 bytes from &3.
User Response: Reconfigure the Queue Manager to allocate a larger repository.

AMQ9501   Usage: &3 [-m QMgrName] -c ChlName.
Explanation: Values passed to the channel program are not valid. The parameter string passed to this program is as follows :- [-m QMgrName] -c ChlName Default values will be used for parameters not supplied.
User Response: Correct the parameters passed to the Channel program and retry the operation.

AMQ9502   Type of channel not suitable for action requested.
Explanation: The operation requested cannot be performed on channel '&3'. Some operations are only valid for certain channel types. For example, you can only ping a channel from the end sending the message.
User Response: Check whether the channel name is specified correctly. If it is check that the channel has been defined correctly.

AMQ9503   Channel negotiation failed.
Explanation: Channel '&3' between this machine and the remote machine could not be established due to a negotiation failure.
User Response: Tell the systems administrator, who should attempt to identify the cause of the channel failure using problem determination techniques. For example, look for FFST files, and examine the error logs on the local and remote systems where there may be messages explaining the cause of failure. More information may be obtained by repeating the operation with tracing enabled.

AMQ9504   A protocol error was detected for channel '&3'.
Explanation: During communications with the remote queue manager, the channel program detected a protocol error. The failure type was &1 with associated data of &2.
User Response: Contact the systems administrator who should examine the error logs to determine the cause of the failure.

AMQ9505   Channel sequence number wrap values are different.
Explanation: The sequence number wrap value for channel '&3' is &1, but the value specified at the remote location is &2. The two values must be the same before the channel can be started.
User Response: Change either the local or remote channel definitions so that the values specified for the message sequence number wrap values are the same.

AMQ9506   Message receipt confirmation failed.
Explanation: Channel '&3' has ended because the remote queue manager did not accept the last batch of messages.
User Response: The error log for the channel at the remote site will contain an explanation of the failure. Contact the remote Systems Administrator to resolve the problem.

AMQ9507   Channel '&3' is currently in-doubt.
Explanation: The requested operation cannot complete because the channel is in-doubt with host '&4'.
User Response: Examine the status of the channel, and either restart a channel to resolve the in-doubt state, or use the RESOLVE CHANNEL command to correct the problem manually.

AMQ9508   Program cannot connect to the queue manager.
Explanation: The connection attempt to queue manager '&4' failed with reason code &1.
User Response: Ensure that the queue manager is available and operational.

AMQ9509   Program cannot open queue manager object.
Explanation: The attempt to open either the queue or queue manager object '&4' on queue manager '&5' failed with reason code &1.
User Response: Ensure that the queue is available and retry the operation.

AMQ9510   Messages cannot be retrieved from a queue.
Explanation: The attempt to get messages from queue '&4' on queue manager '&5' failed with reason code &1.
User Response: If the reason code indicates a conversion problem, for example MQRC_SOURCE_CCSID_ERROR, remove the message(s) from the queue. Otherwise, ensure that the required queue is available and operational.

AMQ9511   Messages cannot be put to a queue.
Explanation: The attempt to put messages to queue '&4' on queue manager '&5' failed with reason code &1.
User Response: Ensure that the required queue is available and operational.

AMQ9512   Ping operation is not valid for channel '&3'.
Explanation: Ping may only be issued for SENDER or SERVER channel types.
User Response: If the local channel is a receiver channel, you must issue the ping from the remote queue manager.

AMQ9513   Maximum number of channels reached.
Explanation: The maximum number of channels that can be in use simultaneously has been reached. The number of permitted channels is a configurable parameter in the queue manager configuration file.
User Response: Wait for some of the operating channels to close. Retry the operation when some channels are available.

AMQ9514   Channel '&3' is in use.
Explanation: The requested operation failed because channel '&3' is currently active.
User Response: Either end the channel manually, or wait for it to close, and retry the operation.

AMQ9515   Channel '&3' changed.
Explanation: The statistics shown are for the channel requested, but it is a new instance of the channel. The previous channel instance has ended.
User Response: None.

AMQ9516   File error occurred.
Explanation: The filesystem returned error code &1 for file '&3'.
User Response: Record the name of the file '&3' and tell the systems administrator, who should ensure that file '&3' is correct and available.

AMQ9517   File damaged.
Explanation: The program has detected damage to the contents of file '&3'.
User Response: Record the values and tell the systems administrator who must restore a saved version of file '&3'. The return code was '&1' and the record length returned was '&2'.

AMQ9518   File '&3' not found.
Explanation: The program requires that the file '&3' is present and available.
User Response: This may be caused by invalid values for the optional environment variables MQCHLLIB, MQCHLTAB or MQDATA. If these variables are valid or not set then record the name of the file and tell the systems administrator who must ensure that file '&3' is available to the program.

AMQ9519   Channel '&3' not found.
Explanation: The requested operation failed because the program could not find a definition of channel '&3'.
User Response: Check that the name is specified correctly and the channel definition is available.

AMQ9520   Channel not defined remotely.
Explanation: There is no definition of channel '&3' at the remote location.
User Response: Add an appropriate definition to the remote hosts list of defined channels and retry the operation.

AMQ9521   Host is not supported by this channel.
Explanation: The connection across channel '&5' was refused because the remote host '&4' did not match the host '&3' specified in the channel definition.
User Response: Update the channel definition, or remove the explicit mention of the remote machine connection name.

AMQ9522   Error accessing the status table.
Explanation: The program could not access the channel status table.
User Response:

AMQ9523   Remote host detected a protocol error.
Explanation: During communications through channel '&3', the remote queue manager channel program detected a protocol error. The failure type was &1 with associated data of &2.
User Response: Tell the systems administrator, who should examine the error files to determine the cause of the failure.

AMQ9524   Remote queue manager unavailable.
Explanation: Channel '&3' cannot start because the remote queue manager is not currently available.
User Response: Either start the remote queue manager, or retry the operation later.

AMQ9525   Remote queue manager is ending.
Explanation: Channel '&3' is closing because the remote queue manager is ending.
User Response: None.

AMQ9526   Message sequence number error for channel '&3'.
Explanation: The local and remote queue managers do not agree on the next message sequence number. A message with sequence number &1 has been sent when sequence number &2 was expected.
User Response: Determine the cause of the inconsistency. It could be that the synchronization information has become damaged, or has been backed out to a previous version. If the situation cannot be resolved, the sequence number can be manually reset at the sending end of the channel using the RESET CHANNEL command.

AMQ9527   Cannot send message through channel '&3'.
Explanation: The channel has closed because the remote queue manager cannot receive a message.
User Response: Contact the systems administrator who should examine the error files of the remote queue manager, to determine why the message cannot be received, and then restart the channel.

AMQ9528   User requested closure of channel '&3'.
Explanation: The channel is closing because of a request by the user.
User Response: None.

AMQ9529   Target queue unknown on remote host.
Explanation: Communication using channel '&3' has ended because the target queue for a message is unknown at the remote host.
User Response: Ensure that the remote host contains a correctly defined target queue, and restart the channel.

AMQ9530   Program could not inquire queue attributes.
Explanation: The attempt to inquire the attributes of queue '&4' on queue manager '&5' failed with reason code &1.
User Response: Ensure that the queue is available and retry the operation.

AMQ9531   Transmission queue specification error.
Explanation: Queue '&4' identified as a transmission queue in the channel definition '&3' is not a transmission queue.
User Response: Ensure that the queue name is specified correctly. If so, alter the queue usage parameter of the queue to that of a transmission queue.

AMQ9532   Program cannot set queue attributes.
Explanation: The attempt to set the attributes of queue '&4' on queue manager '&5' failed with reason code &1.
User Response: Ensure that the queue is available and retry the operation.

AMQ9533   Channel '&3' is not currently active.
Explanation: The channel was not stopped because it was not currently active.
User Response: None.

AMQ9534   Channel '&3' is currently not enabled.
Explanation: The channel program ended because the channel is currently not enabled.
User Response: Issue the START CHANNEL command to re-enable the channel.

AMQ9535   User exit not valid.
Explanation: Channel program '&3' ended because user exit '&4' is not valid.
User Response: Ensure that the user exit is specified correctly in the channel definition, and that the user exit program is correct and available.

AMQ9536   Channel ended by an exit.
Explanation: Channel program '&3' was ended by exit '&4'.
User Response: None.

AMQ9537   Usage: &3 [-m QMgrName] [-q InitQ]
Explanation: Values passed to the Channel Initiator program are not valid. The parameters should be passed as follows: [-m QMgrName] [-q InitQ] Default values are used for parameters that are not supplied.
User Response: Correct the parameters passed to the program and retry the operation.

AMQ9538   Commit control error.
Explanation: An error occurred when attempting to start commitment control. Either exception '&3' was received when querying commitment status, or commitment control could not be started.
User Response: Refer to the error log for other messages pertaining to this problem.

AMQ9539   No channels available.
Explanation: The channel initiator program received a trigger message to start an MCA program to process queue '&3'. The program could not find a defined, available channel to start.
User Response: Ensure that there is a defined channel, which is enabled, to process the transmission queue.

AMQ9540   Commit failed.
Explanation: The program ended because return code &1 was received when an attempt was made to commit change to the resource managers. The commit ID was '&3'.
User Response: Tell the systems administrator.

AMQ9541   CCSID supplied for data conversion not supported.
Explanation: The program ended because, either the source CCSID '&1' or the target CCSID '&2' is not valid, or is not currently supported.
User Response: Correct the CCSID that is not valid, or ensure that the requested CCSID can be supported.

AMQ9542   Queue manager is ending.
Explanation: The program will end because the queue manager is quiescing.
User Response: None.

AMQ9543   Status table damaged.
Explanation: The channel status table has been damaged.
User Response: End all running channels and issue a DISPLAY CHSTATUS command to see the status of the channels. Use the standard facilities supplied with your system to record the problem identifier, and to save the generated output files. Contact your IBM support center. Do not discard these files until the problem has been resolved.

AMQ9544   Messages not put to destination queue.
Explanation: During the processing of channel '&3' one or more messages could not be put to the destination queue and attempts were made to put them to a dead-letter queue. The location of the queue is &1, where 1 is the local dead-letter queue and 2 is the remote dead-letter queue.
User Response: Examine the contents of the dead-letter queue. Each message is contained in a structure that describes why the message was put to the queue, and to where it was originally addressed. Also look at previous error messages to see if the attempt to put messages to a dead-letter queue failed. The program identifier (PID) of the processing program was '&4'.

AMQ9545   Disconnect interval expired.
Explanation: Channel '&3' closed because no messages arrived on the transmission queue within the disconnect interval period.
User Response: None.

AMQ9546   Error return code received.
Explanation: The program has ended because return code &1 was returned from an internal function.
User Response: Correct the cause of the failure and retry the operation.

AMQ9547   Type of remote channel not suitable for action requested.
Explanation: The operation requested cannot be performed because channel '&3' on the remote machine is not of a suitable type. For example, if the local channel is defined as a sender the remote machine must define its channel as either a receiver or requester.
User Response: Check that the channel name is specified correctly. If it is, check that the remote channel has been defined correctly.

AMQ9548   Message put to the 'dead-letter queue'.
Explanation: During processing a message has been put to the dead-letter queue.
User Response: Examine the contents of the dead-letter queue. Each message is contained in a structure that describes why the message was put to the queue, and to where it was originally addressed.

AMQ9549 Transmission Queue '&3' inhibited for MQGET.
Explanation: An MQGET failed because the transmission queue had been previously inhibited for MQGET.
User Response: None.

AMQ9550 Channel program &3 cannot be stopped at this time.
Explanation: The channel program can not be terminated immediately but should end shortly.
User Response: If the channel does not end in a short time issue the STOP CHANNEL command again.

AMQ9551 Protocol not supported by remote host
Explanation: The operation you are performing over Channel '&3' to the host at '&4' is not supported by the target host.
User Response: Check that the connection name parameter is specified correctly and that the levels of the products in use are compatible.

AMQ9552 Security flow not received.
Explanation: During communications through channel '&3' the local security exit requested security data from the remote machine. The security data has not been received so the channel has been closed.
User Response: Tell the systems administrator who should ensure that the security exit on the remote machine is defined correctly.

AMQ9553 The function is not supported.
Explanation: The &3 function &4 attempted is not currently supported on this platform.
User Response: None.

AMQ9554 User not authorized.
Explanation: You are not authorized to perform the Channel operation.
User Response: Tell the systems administrator who should ensure that the correct access permissions are available to you, and then retry the operation.

AMQ9555 File format error.
Explanation: The file '&3' does not have the expected format.
User Response: Ensure that the file name is specified correctly.

AMQ9556 Channel synchronization file missing or damaged.
Explanation: The channel synchronization file '&3' is missing or does not correspond to the stored channel information for queue manager '&4'.
User Response: Rebuild the synchronization file using the rcrmqmobj command rcrmqmobj -t syncfile (-m q-mgr-name)

AMQ9557 Queue Manager User ID initialization failed.
Explanation: The call to initialize the User ID failed with CompCode &1 and Reason &2.
User Response: Correct the error and try again.

AMQ9558 Remote Channel is not currently available.
Explanation: The channel program ended because the channel '&3' is not currently available on the remote system. This could be because the channel is disabled or that the remote system does not have sufficient resources to run a further channel.
User Response: Check the remote system to ensure that the channel is available to run, and retry the operation.

AMQ9560 Rebuild Synchronization File - program started
Explanation: Rebuilding the Synchronization file for Queue Manager '&3' .
User Response: None.

AMQ9561 Rebuild Synchronization File - program completed normally
Explanation: Rebuild Synchronization File program completed normally.
User Response: None.

AMQ9562 Synchronization file in use.
Explanation: The Synchronization file '&3' is in use and cannot be recreated.
User Response: Stop any channel activity and retry the rcrmqmobj command.

AMQ9563 Synchronization file cannot be deleted
Explanation: The filesystem returned error code &1 for file '&3'.
User Response: Tell the systems administrator who should ensure that file '&3' is available and not in use.

AMQ9564 Synchronization File cannot be created
Explanation: The filesystem returned error code &1 for file '&3'.
User Response: Tell the systems administrator.

AMQ9565 No dead-letter queue defined.
Explanation: The queue manager '&4' does not have a defined dead-letter queue.
User Response: Either correct the problem that caused the program to try and write a message to the dead-letter queue or create a dead-letter queue for the queue manager.

AMQ9566   Invalid MQSERVER value
Explanation: The value of the MQSERVER
environment variable was '&3'. The variable should be
in the format
'ChannelName/Protocol/ConnectionName'.
User Response: Correct the MQSERVER value and
retry the operation.

AMQ9572   Message header is not valid.
Explanation: Channel '&3' is stopping because a
message header is not valid. During the processing of
the channel, a message was found that has a header
that is not valid. The dead-letter queue has been
defined as a transmission queue, so a loop would be
created if the message had been put there.
User Response: Correct the problem that caused the
message to have a header that is not valid.

**AMQ9573   Maximum number of active channels
reached.**
Explanation: There are too many channels active to
start another. The current defined maximum number of
active channels is &1.
User Response: Either wait for some of the operating
channels to close or use the stop channel command to
close some channels. Retry the operation when some
channels are available. The maximum number of active
channels is a configurable parameter in the queue
manager configuration file.

AMQ9574   Channel &3 can now be started.
Explanation: Channel &3 has been waiting to start,
but there were no channels available because the
maximum number of active channels was running.
One, or more, of the active channels has now closed so
this channel can start.
User Response:

**AMQ9575   DCE Security: failed to get the user's
login name.**
Explanation: System call &4 to get the login name of
the user running MQSeries client application process
&1 failed with error value &2. This occurred in security
exit function create_cred. The exit will now attempt to
open channel &3 using the DCE default login context.
User Response: If you wish to run using the DCE
default login context take no action. If you wish to run
using the user's login name as the DCE security exit
principal examine the documentation for the operating
system on which you are running MQSeries clients and
reconfigure the operating system as necessary to allow
the &4 call to succeed.

**AMQ9576   DCE Security: an exit could not allocate
memory.**
Explanation: A DCE exit was unsuccessful in
obtaining the memory it needed. The failure occurred
in exit function &4. Channel &3 is closed.
User Response: Make more memory available to the
MQSeries system and restart the relevant channel.

AMQ9577   DCE security exit: no partner name.
Explanation: Channel &3 has not been opened
because the DCE security exit which initiates the
security context was not passed a valid partner name.
When the DCE security exit is called to initiate the
security context it is essential that the PartnerName
field in the MQCXP structure contains a valid partner
name. On this call it did not. This can arise as a result
of a usage error, for instance only specifying the
security exit on one end of the channel. The error was
reported from security exit function savePartnerName.
User Response: Check your usage of the DCE security
exit for errors, such as only specifying the exit in one of
the matching channel definitions. Correct any errors
found and retry.

AMQ9578   DCE Security: bad return from DCE call.
Explanation: Channel &3 has been closed because one
of the DCE channel exits received a bad return code
from DCE.
User Response: Consult the appropriate DCE manuals
to find out the meaning of major_status &1 and
minor_status &2 on call &5. Then rectify the error. The
exit function name is &4.

**AMQ9579   DCE Security: partner name does not
match target.**
Explanation: The DCE Security exit was requested to
perform a trusted channel check: target partner name
&4 was specified in the SCYDATA field of channel &3.
The actual partner name associated with channel &3
was &5, so the security exit suppressed the channel.
User Response: Examine the channel definition of
channel &3 and alter it so that the relevant name on
the partner system matches that specified in the
SCYDATA field.

AMQ9580   DCE Security: invalid message received.
Explanation: An IBM-supplied DCE exit on channel
&3 received a message that was not generated by a
matching exit, or was not the expected type of message.
The header.mechanism field had value &1. The
header.msgtype field had value &2. The name of the
exit function in which the error was discovered is &4.
User Response: Make sure that the exits at both ends
of the channel generate compatible flows.

AMQ9581   DCE Security: wrong exit called.
Explanation: Exit &4 on channel &3 was called for use as an MQSeries exit of the wrong type. DCE_SEC_SCY_CHANNELEXIT functions as a security exit; DCE_SEC_SRM_CHANNELEXIT functions as a send, receive or message exit. The ExitId parameter passed to the exit was &1.
User Response: Alter the exit definitions to ensure that exit &4 is called correctly.

AMQ9582   DCE Security: invalid exit function requested.
Explanation: Exit &4 on channel &3 was called with an invalid ExitReason (value &1).
User Response: Check that the exit is being run with a compatible release of MQSeries base code. If not then correct it. If it is, contact your IBM support center for help.

AMQ9583   The DCE security exit was not run.
Explanation: The DCE_SEC_SRM_CHANNELEXIT exit was called on channel &3; the value of pContext->mechanism (&1) passed was not valid.
User Response: This is probably because the DCE_SEC_SRM_CHANNELEXIT exit has been called without first calling the DCE_SEC_SCY_CHANNELEXIT security exit. Alter the system so that either both or neither are run.

AMQ9584   DCE Security: message too short.
Explanation: The DCE_SEC_SRM_CHANNELEXIT receive or message exit was called on channel &3 to process an incoming message. The pDataLength parameter supplied to the exit indicated that the message received was too short to be a valid message for the relevant exit. The *pDataLength value was &1.
User Response: Configure the system so that compatible send/receive/message exits are run at both ends of the channel.

AMQ9585   Maximum number of channel initiators reached.
Explanation: The maximum number of channels initiators that can be in use simultaneously has been reached. The number of permitted channel initiators is a configurable parameter in the queue manager configuration file.
User Response: Wait for one or more channel initiators to close and retry the operation or modify the configuration file to allow more initiators and restart the Queue Manager.

AMQ9586   Program cannot create queue manager object.
Explanation: The attempt to create object '&4' on queue manager '&5' failed with reason code &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier.

Contact your IBM support center.

AMQ9587   Program cannot open queue manager object.
Explanation: The attempt to open object '&4' on queue manager '&5' failed with reason code &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier. Contact your IBM support center.

AMQ9588   Program cannot update queue manager object.
Explanation: The attempt to update object '&4' on queue manager '&5' failed with reason code &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier. Contact your IBM support center.

AMQ9589   Program cannot query queue manager object.
Explanation: The attempt to query object '&4' on queue manager '&5' failed with reason code &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier. Contact your IBM support center.

AMQ9590   Program cannot close queue manager object.
Explanation: The attempt to close object '&4' on queue manager '&5' failed with reason code &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier. Contact your IBM support center.

AMQ9591   Program cannot prepare queue manager object.
Explanation: The attempt to prepare object '&4' on queue manager '&5' failed with reason code &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier. Contact your IBM support center.

AMQ9592   Program cannot resolve queue manager object.
Explanation: The attempt to resolve object '&4' on queue manager '&5' failed with reason code &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier. Contact your IBM support center.

AMQ9593   Program cannot delete queue manager object.
Explanation: The attempt to delete object '&4' on queue manager '&5' failed with reason code &1.
User Response: Use the standard facilities supplied with your system to record the problem identifier. Contact your IBM support center.

AMQ9594   Usage: runmqfmt [filename].
Explanation: Syntax for the usage of runmqfmt.
User Response: None.

AMQ9595   Usage: endmqlsr [-m QMgrName]
Explanation: The correct usage is shown.
User Response: Correct the parameters passed to the
endmqlsr program and retry the operation.

AMQ9596   Queue Manager '&3' still running
Explanation: The requested operation can not
complete because queue manager '&3' is still running.
User Response: End the queue manager and retry the
operation.

**AMQ9597   No MQSeries listeners found for the
              given Queue Manager**
Explanation: No MQSeries listener processes were
found in the system for Queue Manager '&3'.
User Response: None.

AMQ9598   &1 MQSeries listeners will end shortly.
Explanation: &1 MQSeries listeners detected in the
system are scheduled for shutdown.
User Response: None.

**AMQ9599   Program could not open queue manager
              object.**
Explanation: The attempt to open either the queue or
queue manager object '&4' on queue manager '&5' by
user '&3' failed with reason code &1.
User Response: Ensure that the queue is available and
retry the operation. If the message is from a remote
Queue Manager, check the Message Channel Agent
User Identifier has the correct authority.

AMQ9999   Channel program ended abnormally.
Explanation: Channel program '&3' ended abnormally.
User Response: Look at previous error messages for
channel program '&3' in the error files to determine the
cause of the failure.

Chapter 9. API completion and reason codes

For each call, a completion code and a reason code are returned by the queue manager or by an exit routine, to indicate the success or failure of the call.

For more information about the MQSeries API, see the *MQSeries Application Programming Guide* and the *MQSeries Application Programming Reference* manual.

Completion codes

The following is a list of the completion codes (MQCC) returned by MQSeries:

0   Successful completion (MQCC_OK)

The call completed fully; all output parameters have been set.

The *Reason* parameter always has the value MQRC_NONE in this case.

1   Warning (partial completion) (MQCC_WARNING)

The call completed partially. Some output parameters might have been set in addition to the *CompCode* and *Reason* output parameters.

The *Reason* parameter gives additional information.

2   Call failed (MQCC_FAILED)

The processing of the call did not complete, and the state of the queue manager is normally unchanged; exceptions are specifically noted. Only the *CompCode* and *Reason* output parameters have been set; all other parameters are unchanged.

The reason might be a fault in the application program, or it might be a result of some situation external to the program, for example the application's authority might have been revoked. The *Reason* parameter gives additional information.

Reason codes

The reason code parameter (MQRC) is a qualification to the completion code.

If there is no special reason to report, MQRC_NONE is returned. Typically, a successful call returns MQCC_OK and MQRC_NONE.

If the completion code is either MQCC_WARNING or MQCC_FAILED, the queue manager always reports a qualifying reason.

0	X'0000'	MQRC_NONE
		The call completed normally. The completion code (*CompCode*) is MQCC_OK.
		Corrective action: None.

Completion and reason codes

900	X'0384'	**MQRC_APPL_FIRST** This is the lowest value for an application-defined reason code returned by a data-conversion exit. Data-conversion exits can return reason codes in the range MQRC_APPL_FIRST through MQRC_APPL_LAST to indicate particular conditions that the exit has detected. Corrective action: As defined by the writer of the data-conversion exit.
999	X'03E7'	**MQRC_APPL_LAST** This is the highest value for an application-defined reason code returned by a data-conversion exit. Data-conversion exits can return reason codes in the range MQRC_APPL_FIRST through MQRC_APPL_LAST to indicate particular conditions that the exit has detected. Corrective action: As defined by the writer of the data-conversion exit.
2001	X'07D1'	**MQRC_ALIAS_BASE_Q_TYPE_ERROR** An MQOPEN or MQPUT1 call was issued specifying an alias queue as the destination, but the *BaseQName* in the alias queue definition resolves to a queue that is not a local queue, a local definition of a remote queue, or a cluster queue. Corrective action: Correct the queue definitions.
2002	X'07D2'	**MQRC_ALREADY_CONNECTED** An MQCONN or MQCONNX call was issued, but the application is already connected to the queue manager. On OS/390, this reason code occurs for batch and IMS applications only; it does not occur for CICS applications. On Windows NT, MTS objects do not receive this reason code, as additional connections to the queue manager are allowed. Corrective action: None. The *Hconn* parameter returned has the same value as was returned for the previous MQCONN or MQCONNX call. An MQCONN or MQCONNX call that returns this reason code does *not* mean that an additional MQDISC call must be issued in order to disconnect from the queue manager. If this reason code is returned because the application has been called in a situation where the connect has already been done, a corresponding MQDISC should *not* be issued, because this will cause the application that issued the original MQCONN or MQCONNX call to be disconnected as well.

2003	X'07D3'	**MQRC_BACKED_OUT**
		
		The current unit of work encountered a fatal error or was backed out. This occurs in the following cases:
		• On an MQCMIT or MQDISC call, when the commit operation has failed and the unit of work has been backed out. All resources that participated in the unit of work have been returned to their state at the start of the unit of work. The MQCMIT call returns completion code MQCC_FAILED; the MQDISC call returns completion code MQCC_WARNING.
		   On OS/390, this reason code occurs only for batch applications.
		• On an MQGET, MQPUT, or MQPUT1 call that is operating within a unit of work, when the unit of work has already encountered an error that prevents the unit of work being committed (for example, when the log space is exhausted). The application must issue the appropriate call to back out the unit of work. For a unit of work coordinated by the queue manager, this call is the MQBACK call, although the MQCMIT call has the same effect in these circumstances.
		   On OS/390, this case does not occur.
		
		Corrective action: Check the returns from previous calls to the queue manager. For example, a previous MQPUT call may have failed.

2004	X'07D4'	MQRC_BUFFER_ERROR The *Buffer* parameter is not valid for one of the following reasons: • The parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The parameter pointer points to storage that cannot be accessed for the entire length specified by *BufferLength*. • For calls where *Buffer* is an output parameter: the parameter pointer points to read-only storage. Corrective action: Correct the parameter.
2005	X'07D5'	MQRC_BUFFER_LENGTH_ERROR The *BufferLength* parameter is not valid, or the parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) This reason can also be returned to an MQSeries client program on the MQCONN or MQCONNX call if the negotiated maximum message size for the channel is smaller than the fixed part of any call structure. Corrective action: Specify a value that is zero or greater. For the mqAddString and mqSetString calls, the special value MQBL_NULL_TERMINATED is also valid.
2006	X'07D6'	MQRC_CHAR_ATTR_LENGTH_ERROR *CharAttrLength* is negative (for MQINQ or MQSET calls), or is not large enough to hold all selected attributes (MQSET calls only). This reason also occurs if the parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Specify a value large enough to hold the concatenated strings for all selected attributes.
2007	X'07D7'	MQRC_CHAR_ATTRS_ERROR *CharAttrs* is not valid. The parameter pointer is not valid, or points to read-only storage for MQINQ calls or to storage that is not as long as implied by *CharAttrLength*. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Correct the parameter.
2008	X'07D8'	MQRC_CHAR_ATTRS_TOO_SHORT For MQINQ calls, *CharAttrLength* is not large enough to contain all of the character attributes for which MQCA_* selectors are specified in the *Selectors* parameter. The call still completes, with the *CharAttrs* parameter string filled in with as many character attributes as there is room for. Only complete attribute strings are returned: if there is insufficient space remaining to accommodate an attribute in its entirety, that attribute and subsequent character attributes are omitted. Any space at the end of the string not used to hold an attribute is unchanged. An attribute that represents a set of values (for example, the namelist *Names* attribute) is treated as a single entity—either all of its values are returned, or none. Corrective action: Specify a large enough value, unless only a subset of the values is needed.

Completion and reason codes

2009	X'07D9'	MQRC_CONNECTION_BROKEN
		Connection to the queue manager has been lost. This can occur because the queue manager has ended. If the call is an MQGET call with the MQGMO_WAIT option, the wait has been canceled. All connection and object handles are now invalid.
		For MQSeries client applications, it is possible that the call did complete successfully, even though this reason code is returned with a *CompCode* of MQCC_FAILED.
		Corrective action: Applications can attempt to reconnect to the queue manager by issuing the MQCONN or MQCONNX call. It may be necessary to poll until a successful response is received.
		On OS/390 for CICS applications, it is not necessary to issue the MQCONN or MQCONNX call, because CICS applications are connected automatically.
		Any uncommitted changes in a unit of work should be backed out. A unit of work that is coordinated by the queue manager is backed out automatically.
2010	X'07DA'	MQRC_DATA_LENGTH_ERROR
		The *DataLength* parameter is not valid. Either the parameter pointer is not valid, or it points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		This reason can also be returned to an MQSeries client program that is putting and getting messages, if the application message data is longer than the negotiated maximum message size for the channel.
		Corrective action: Correct the parameter.
		If the error occurs for an MQSeries client program, also check that the maximum message size for the channel is big enough to accommodate the message being sent; if it is not big enough, increase the maximum message size for the channel.
2011	X'07DB'	MQRC_DYNAMIC_Q_NAME_ERROR
		On the MQOPEN call, a model queue is specified in the *ObjectName* field of the *ObjDesc* parameter, but the *DynamicQName* field is not valid, for one of the following reasons: • *DynamicQName* is completely blank (or blank up to the first null character in the field). • Characters are present that are not valid for a queue name. • An asterisk is present beyond the 33rd position (and before any null character). • An asterisk is present followed by characters that are not null and not blank.
		This reason code can also sometimes occur when a server application opens the reply queue specified by the *ReplyToQ* and *ReplyToQMgr* fields in the MQMD of a message that the server has just received. In this case the reason code indicates that the application that sent the original message placed incorrect values into the *ReplyToQ* and *ReplyToQMgr* fields in the MQMD of the original message.
		Corrective action: Specify a valid name.

2012	X'07DC'	MQRC_ENVIRONMENT_ERROR
		
		The call is not valid for the current environment.
		
		• On OS/390, one of the following applies:
		  – An MQCONN or MQCONNX call was issued, but the application has been linked with an adapter that is not supported in the environment in which the application is running. For example, this can arise when the application is linked with the MQSeries RRS adapter, but the application is running in a DB2 Stored Procedure address space. RRS is not supported in this environment. Stored Procedures wishing to use the MQSeries RRS adapter must run in a DB2 WLM-managed Stored Procedure address space.
		  – An MQCMIT or MQBACK call was issued in the CICS or IMS environment.
		  – The RRS subsystem is not up and running on the OS/390 system that ran the application.
		• On Compaq (DIGITAL) OpenVMS, OS/2, AS/400, Tandem NonStop Kernel, UNIX systems, and Windows NT, one of the following applies:
		  – The application is linked to the wrong libraries (threaded or nonthreaded).
		  – An MQBEGIN, MQCMIT, or MQBACK call was issued, but an external unit-of-work manager is in use. For example, this reason code occurs on Windows NT when an MTS object is running as a DTC transaction. This reason code also occurs if the queue manager does not support units of work.
		  – The MQBEGIN call was issued in an MQSeries client environment.
		  – An MQXCLWLN call was issued, but the call did not originate from a cluster workload exit.
		
		Corrective action: Do one of the following (as appropriate):
		• On OS/390:
		  – Link the application with the correct adapter.
		  – For a CICS or IMS application, issue the appropriate CICS or IMS call to commit or backout the unit of work.
		  – Start the RRS subsystem on the OS/390 system that is running the application.
		• In the other environments:
		  – Link the application with the correct libraries (threaded or nonthreaded).
		  – Remove from the application the call that is not supported.

2013	X'07DD'	MQRC_EXPIRY_ERROR On an MQPUT or MQPUT1 call, the value specified for the *Expiry* field in the message descriptor MQMD is not valid. Corrective action: Specify a value that is greater than zero, or the special value MQEI_UNLIMITED.
2014	X'07DE'	MQRC_FEEDBACK_ERROR On an MQPUT or MQPUT1 call, the value specified for the *Feedback* field in the message descriptor MQMD is not valid. The value is not MQFB_NONE, and is outside both the range defined for system feedback codes and the range defined for application feedback codes. Corrective action: Specify MQFB_NONE, or a value in the range MQFB_SYSTEM_FIRST through MQFB_SYSTEM_LAST, or MQFB_APPL_FIRST through MQFB_APPL_LAST.
2016	X'07E0'	MQRC_GET_INHIBITED MQGET calls are currently inhibited for the queue, or for the queue to which this queue resolves. Corrective action: If the system design allows get requests to be inhibited for short periods, retry the operation later.

Completion and reason codes

2017	X'07E1'	MQRC_HANDLE_NOT_AVAILABLE
		An MQOPEN or MQPUT1 call was issued, but the maximum number of open handles allowed for the current task has already been reached. Be aware that when a distribution list is specified on the MQOPEN or MQPUT1 call, each queue in the distribution list uses one handle.
		On OS/390, "task" means a CICS task, an MVS™ task, or an IMS-dependent region.
		Corrective action: Check whether the application is issuing MQOPEN calls without corresponding MQCLOSE calls. If it is, modify the application to issue the MQCLOSE call for each open object as soon as that object is no longer needed.
		Also check whether the application is specifying a distribution list containing a large number of queues that are consuming all of the available handles. If it is, increase the maximum number of handles that the task can use, or reduce the size of the distribution list. The maximum number of open handles that a task can use is given by the *MaxHandles* queue manager attribute.
2018	X'07E2'	MQRC_HCONN_ERROR
		The connection handle *Hconn* is not valid. This reason also occurs if the parameter pointer is not valid, or (for the MQCONN or MQCONNX call) points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		This reason code can also occur in an MTS environment when trying to use a connection handle in a situation where it is not valid, such as passing it between processes or packages, neither of which is supported. (Passing the connection handle between library packages *is* supported.)
		Corrective action: Ensure that a successful MQCONN or MQCONNX call is performed for the queue manager, and that an MQDISC call has not already been performed for it. Ensure that the handle is being used within its valid scope.
		On OS/390, also check that the application has been linked with the correct stub; this is CSQCSTUB for CICS® applications, CSQBSTUB for batch applications, and CSQQSTUB for IMS applications. Also, the stub used must not belong to a release of MQSeries that is more recent than the release on which the application will run.
2019	X'07E3'	MQRC_HOBJ_ERROR
		The object handle *Hobj* is not valid. This reason also occurs if the parameter pointer is not valid, or (for the MQOPEN call) points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		Corrective action: Ensure that a successful MQOPEN call is performed for this object, and that an MQCLOSE call has not already been performed for it. For MQGET and MQPUT calls, also ensure that the handle represents a queue object. Ensure that the handle is being used within its valid scope.
2020	X'07E4'	MQRC_INHIBIT_VALUE_ERROR
		On an MQSET call, the value specified for either the MQIA_INHIBIT_GET attribute or the MQIA_INHIBIT_PUT attribute is not valid.
		Corrective action: Specify a valid value.

2021	X'07E5'	MQRC_INT_ATTR_COUNT_ERROR On an MQINQ or MQSET call, the *IntAttrCount* parameter is negative (MQINQ or MQSET), or smaller than the number of integer attribute selectors (MQIA_*) specified in the *Selectors* parameter (MQSET only). This reason also occurs if the parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Specify a value large enough for all selected integer attributes.
2022	X'07E6'	MQRC_INT_ATTR_COUNT_TOO_SMALL On an MQINQ call, the *IntAttrCount* parameter is smaller than the number of integer attribute selectors (MQIA_*) specified in the *Selectors* parameter. The call completes with MQCC_WARNING, with the *IntAttrs* array filled in with as many integer attributes as there is room for. Corrective action: Specify a large enough value, unless only a subset of the values is needed.
2023	X'07E7'	MQRC_INT_ATTRS_ARRAY_ERROR On an MQINQ or MQSET call, the *IntAttrs* parameter is not valid. The parameter pointer is not valid (MQINQ and MQSET), or points to read-only storage or to storage that is not as long as indicated by the *IntAttrCount* parameter (MQINQ only). (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Correct the parameter.
2024	X'07E8'	MQRC_SYNCPOINT_LIMIT_REACHED An MQGET, MQPUT, or MQPUT1 call failed because it would have caused the number of uncommitted messages in the current unit of work to exceed the limit defined for the queue manager (see the *MaxUncommittedMsgs* queue-manager attribute). The number of uncommitted messages is the sum of the following since the start of the current unit of work: • Messages put by the application with the MQPMO_SYNCPOINT option • Messages retrieved by the application with the MQGMO_SYNCPOINT option • Trigger messages and COA report messages generated by the queue manager for messages put with the MQPMO_SYNCPOINT option • COD report messages generated by the queue manager for messages retrieved with the MQGMO_SYNCPOINT option • On Tandem NonStop Kernel, this reason code occurs when the maximum number of I/O operations in a single TM/MP transaction has been exceeded. Corrective action: Check whether the application is looping. If it is not, consider reducing the complexity of the application. Alternatively, increase the queue-manager limit for the maximum number of uncommitted messages within a unit of work. • On OS/390, the limit for the maximum number of uncommitted messages can be changed by using the DEFINE MAXSMSGS command. • On AS/400, the limit for the maximum number of uncommitted messages can be changed by using the CHGMQM command. • On Tandem NonStop Kernel, the application should cancel the transaction and retry with a smaller number of operations in the unit of work. See the *MQSeries for Tandem NonStop Kernel System Management Guide* for more details.

Completion and reason codes

2025	X'07E9'	MQRC_MAX_CONNS_LIMIT_REACHED The MQCONN or MQCONNX call was rejected because the maximum number of concurrent connections has been exceeded. • On OS/390, connection limits are applicable only to TSO and batch requests. The limits are determined by the customer using the following parameters of the CSQ6SYSP macro: – For TSO: IDFORE – For batch: IDBACK For more information, see the *MQSeries for OS/390 System Setup Guide*. • On Compaq (DIGITAL) OpenVMS, OS/2, AS/400, Tandem NonStop Kernel, UNIX systems, and Windows NT, this reason code can also occur on the MQOPEN call. Corrective action: Either increase the size of the appropriate install parameter value, or reduce the number of concurrent connections.
2026	X'07EA'	MQRC_MD_ERROR The MQMD structure is not valid, for one of the following reasons: • The *StrucId* field is not MQMD_STRUC_ID. • The *Version* field specifies a value that is not valid or not supported. • The parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The queue manager cannot copy the changed structure to application storage, even though the call is successful. This can occur, for example, if the pointer points to read-only storage. Corrective action: Correct the definition of the message descriptor. Ensure that required input fields are correctly set.
2027	X'07EB'	MQRC_MISSING_REPLY_TO_Q On an MQPUT or MQPUT1 call, the *ReplyToQ* field in the message descriptor MQMD is blank, but one or both of the following is true: • A reply was requested (that is, MQMT_REQUEST was specified in the *MsgType* field of the message descriptor). • A report message was requested in the *Report* field of the message descriptor. Corrective action: Specify the name of the queue to which the reply message or report message is to be sent.
2029	X'07ED'	MQRC_MSG_TYPE_ERROR On an MQPUT or MQPUT1 call, the value specified for the *MsgType* field in the message descriptor (MQMD) is not valid. Corrective action: Specify a valid value.

2030	X'07EE'	MQRC_MSG_TOO_BIG_FOR_Q
		An MQPUT or MQPUT1 call was issued to put a message on a queue, but the message was too long for the queue and MQMF_SEGMENTATION_ALLOWED was not specified in the *MsgFlags* field in MQMD. If segmentation is not allowed, the length of the message cannot exceed the lesser of the queue and queue-manager *MaxMsgLength* attributes.

This reason code can also occur when MQMF_SEGMENTATION_ALLOWED *is* specified, but the nature of the data present in the message prevents the queue manager splitting it into segments that are small enough to place on the queue:
• For a user-defined format, the smallest segment that the queue manager can create is 16 bytes.
• For a built-in format, the smallest segment that the queue manager can create depends on the particular format, but is greater than 16 bytes in all cases other than MQFMT_STRING (for MQFMT_STRING the minimum segment size is 16 bytes).

MQRC_MSG_TOO_BIG_FOR_Q can also occur in the *Feedback* field in the message descriptor of a report message; in this case it indicates that the error was encountered by a message channel agent when it attempted to put the message on a remote queue.

Corrective action: Check whether the *BufferLength* parameter is specified correctly; if it is, do one of the following:
• Increase the value of the queue's *MaxMsgLength* attribute; the queue-manager's *MaxMsgLength* attribute may also need increasing.
• Break the message into several smaller messages.
• Specify MQMF_SEGMENTATION_ALLOWED in the *MsgFlags* field in MQMD; this will allow the queue manager to break the message into segments. |
| 2031 | X'07EF' | MQRC_MSG_TOO_BIG_FOR_Q_MGR |
| | | An MQPUT or MQPUT1 call was issued to put a message on a queue, but the message was too long for the queue manager and MQMF_SEGMENTATION_ALLOWED was not specified in the *MsgFlags* field in MQMD. If segmentation is not allowed, the length of the message cannot exceed the lesser of the queue and queue-manager *MaxMsgLength* attributes.

This reason code can also occur when MQMF_SEGMENTATION_ALLOWED *is* specified, but the nature of the data present in the message prevents the queue manager splitting it into segments that are small enough for the queue-manager limit:
• For a user-defined format, the smallest segment that the queue manager can create is 16 bytes.
• For a built-in format, the smallest segment that the queue manager can create depends on the particular format, but is greater than 16 bytes in all cases other than MQFMT_STRING (for MQFMT_STRING the minimum segment size is 16 bytes).

MQRC_MSG_TOO_BIG_FOR_Q_MGR can also occur in the *Feedback* field in the message descriptor of a report message; in this case it indicates that the error was encountered by a message channel agent when it attempted to put the message on a remote queue.

This reason also occurs if a channel, through which the message is to pass, has restricted the maximum message length to a value that is actually less than that supported by the queue manager, and the message length is greater than this value.

On OS/390, this return code is issued only if you are using CICS for distributed queuing. Otherwise, MQRC_MSG_TOO_BIG_FOR_CHANNEL is issued.

Corrective action: Check whether the *BufferLength* parameter is specified correctly; if it is, do one of the following:
• Increase the value of the queue-manager's *MaxMsgLength* attribute; the queue's *MaxMsgLength* attribute may also need increasing.
• Break the message into several smaller messages.
• Specify MQMF_SEGMENTATION_ALLOWED in the *MsgFlags* field in MQMD; this will allow the queue manager to break the message into segments.
• Check the channel definitions. |

Completion and reason codes

2033	X'07F1'	**MQRC_NO_MSG_AVAILABLE**
		
		An MQGET call was issued, but there is no message on the queue satisfying the selection criteria specified in MQMD (the *MsgId* and *CorrelId* fields), and in MQGMO (the *Options* and *MatchOptions* fields). Either the MQGMO_WAIT option was not specified, or the time interval specified by the *WaitInterval* field in MQGMO has expired. This reason is also returned for an MQGET call for browse, when the end of the queue has been reached.
		
		This reason code can also be returned by the mqGetBag and mqExecute calls. mqGetBag is similar to MQGET. For the mqExecute call, the completion code can be either MQCC_WARNING or MQCC_FAILED:
		• If the completion code is MQCC_WARNING, some response messages were received during the specified wait interval, but not all. The response bag contains system-generated nested bags for the messages that were received.
		• If the completion code is MQCC_FAILED, no response messages were received during the specified wait interval.
		
		Corrective action: If this is an expected condition, no corrective action is required.
		
		If this is an unexpected condition, check that:
		• The message was put on the queue successfully.
		• The unit of work (if any) used for the MQPUT or MQPUT1 call was committed successfully.
		• The options controlling the selection criteria are specified correctly. All of the following can affect the eligibility of a message for return on the MQGET call:
		    MQGMO_LOGICAL_ORDER
		    MQGMO_ALL_MSGS_AVAILABLE
		    MQGMO_ALL_SEGMENTS_AVAILABLE
		    MQGMO_COMPLETE_MSG
		    MQMO_MATCH_MSG_ID
		    MQMO_MATCH_CORREL_ID
		    MQMO_MATCH_GROUP_ID
		    MQMO_MATCH_MSG_SEQ_NUMBER
		    MQMO_MATCH_OFFSET
		    Value of *MsgId* field in MQMD
		    Value of *CorrelId* field in MQMD
		
		Consider waiting longer for the message.

2034	X'07F2'	**MQRC_NO_MSG_UNDER_CURSOR** An MQGET call was issued with either the MQGMO_MSG_UNDER_CURSOR or the MQGMO_BROWSE_MSG_UNDER_CURSOR option. However, the browse cursor is not positioned at a retrievable message. This is caused by one of the following: • The cursor is positioned logically before the first message (as it is before the first MQGET call with a browse option has been successfully performed). • The message the browse cursor was positioned on has been locked or removed from the queue (probably by some other application) since the browse operation was performed. • The message the browse cursor was positioned on has expired. Corrective action: Check the application logic. This may be an expected reason if the application design allows multiple servers to compete for messages after browsing. Consider also using the MQGMO_LOCK option with the preceding browse MQGET call.

2035	X'07F3'	MQRC_NOT_AUTHORIZED
		The user is not authorized to perform the operation attempted: • On an MQCONN or MQCONNX call, the user is not authorized to connect to the queue manager. On OS/390, for CICS applications, MQRC_CONNECTION_NOT_AUTHORIZED is issued instead. • On an MQOPEN or MQPUT1 call, the user is not authorized to open the object for the option(s) specified. On OS/390, if the object being opened is a model queue, this reason also arises if the user is not authorized to create a dynamic queue with the required name. • On an MQCLOSE call, the user is not authorized to delete the object, which is a permanent dynamic queue, and the *Hobj* parameter specified on the MQCLOSE call is not the handle returned by the MQOPEN call that created the queue. This reason code can also occur in the *Feedback* field in the message descriptor of a report message; in this case it indicates that the error was encountered by a message channel agent when it attempted to put the message on a remote queue. Corrective action: Ensure that the correct queue manager or object was specified, and that appropriate authority exists. On OS/390, to determine for which object you are not authorized, you can use the violation messages issued by the External Security Manager.
2036	X'07F4'	MQRC_NOT_OPEN_FOR_BROWSE An MQGET call was issued with one of the following options: MQGMO_BROWSE_FIRST MQGMO_BROWSE_NEXT MQGMO_BROWSE_MSG_UNDER_CURSOR MQGMO_MSG_UNDER_CURSOR but the queue had not been opened for browse. Corrective action: Specify MQOO_BROWSE when the queue is opened.
2037	X'07F5'	MQRC_NOT_OPEN_FOR_INPUT An MQGET call was issued to retrieve a message from a queue, but the queue had not been opened for input. Corrective action: Specify one of the following when the queue is opened: MQOO_INPUT_SHARED MQOO_INPUT_EXCLUSIVE MQOO_INPUT_AS_Q_DEF
2038	X'07F6'	MQRC_NOT_OPEN_FOR_INQUIRE An MQINQ call was issued to inquire object attributes, but the object had not been opened for inquire. Corrective action: Specify MQOO_INQUIRE when the object is opened.
2039	X'07F7'	MQRC_NOT_OPEN_FOR_OUTPUT An MQPUT call was issued to put a message on a queue, but the queue had not been opened for output. Corrective action: Specify MQOO_OUTPUT when the queue is opened.

Completion and reason codes

2040	X'07F8'	MQRC_NOT_OPEN_FOR_SET An MQSET call was issued to set queue attributes, but the queue had not been opened for set. Corrective action: Specify MQOO_SET when the object is opened.
2041	X'07F9'	MQRC_OBJECT_CHANGED Object definitions that affect this object have been changed since the *Hobj* handle used on this call was returned by the MQOPEN call. This reason does not occur if the object handle is specified in the *Context* field of the *PutMsgOpts* parameter on the MQPUT or MQPUT1 call. Corrective action: Issue an MQCLOSE call to return the handle to the system. It is then usually sufficient to reopen the object and retry the operation. However, if the object definitions are critical to the application logic, an MQINQ call can be used after reopening the object, to obtain the new values of the object attributes.
2042	X'07FA'	MQRC_OBJECT_IN_USE An MQOPEN call was issued, but the object in question has already been opened by this or another application with options that conflict with those specified in the *Options* parameter. This arises if the request is for shared input, but the object is already open for exclusive input; it also arises if the request is for exclusive input, but the object is already open for input (of any sort). MCAs for receiver channels, or the intra-group queuing (IGQ) agent, may keep the destination queues open even when messages are not being transmitted; this results in the queues appearing to be "in use". Use the MQSeries command DISPLAY QSTATUS to find out what is keeping the queues open. On OS/390, this reason can also occur for an MQOPEN or MQPUT1 call, if the object to be opened (which can be a queue, or for MQOPEN a namelist or process object) is in the process of being deleted. Corrective action: System design should specify whether an application is to wait and retry, or take other action.
2043	X'07FB'	MQRC_OBJECT_TYPE_ERROR On the MQOPEN or MQPUT1 call, the *ObjectType* field in the object descriptor MQOD specifies a value that is not valid. For the MQPUT1 call, the object type must be MQOT_Q. Corrective action: Specify a valid object type.
2044	X'07FC'	MQRC_OD_ERROR On the MQOPEN or MQPUT1 call, the object descriptor MQOD is not valid, for one of the following reasons: • The *StrucId* field is not MQOD_STRUC_ID. • The *Version* field specifies a value that is not valid or not supported. • The parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The queue manager cannot copy the changed structure to application storage, even though the call is successful. This can occur, for example, if the pointer points to read-only storage. Corrective action: Correct the definition of the object descriptor. Ensure that required input fields are set correctly.

2045	X'07FD'	MQRC_OPTION_NOT_VALID_FOR_TYPE
		On an MQOPEN or MQCLOSE call, an option is specified that is not valid for the type of object or queue being opened or closed.
		For the MQOPEN call, this includes the following cases:
		• An option that is inappropriate for the object type (for example, MQOO_OUTPUT for an MQOT_PROCESS object).
		• An option that is unsupported for the queue type (for example, MQOO_INQUIRE for a remote queue that has no local definition).
		• One or more of the following options: MQOO_INPUT_AS_Q_DEF MQOO_INPUT_SHARED MQOO_INPUT_EXCLUSIVE MQOO_BROWSE MQOO_INQUIRE MQOO_SET
		when either:
		– the queue name is resolved through a cell directory, or
		– *ObjectQMgrName* in the object descriptor specifies the name of a local definition of a remote queue (in order to specify a queue-manager alias), and the queue named in the *RemoteQMgrName* attribute of the definition is the name of the local queue manager.
		For the MQCLOSE call, this includes the following case: The MQCO_DELETE or MQCO_DELETE_PURGE option when the queue is not a dynamic queue.
		This reason code can also occur on the MQOPEN call when the object being opened is of type MQOT_NAMELIST, MQOT_PROCESS, or MQOT_Q_MGR, but the *ObjectQMgrName* field in MQOD is neither blank nor the name of the local queue manager.
		Corrective action: Specify the correct option; see the *MQSeries Application Programming Reference* for more information. For the MQOPEN call, ensure that the *ObjectQMgrName* field is set correctly. For the MQCLOSE call, either correct the option or change the definition type of the model queue that is used to create the new queue.
2046	X'07FE'	MQRC_OPTIONS_ERROR
		The *Options* parameter or field contains options that are not valid, or a combination of options that is not valid.
		• For the MQOPEN, MQCLOSE, MQXCNVC, mqBagToBuffer, mqBufferToBag, mqCreateBag, and mqExecute calls, *Options* is a separate parameter on the call.
		This reason also occurs if the parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		• For the MQBEGIN, MQCONNX, MQGET, MQPUT, and MQPUT1 calls, *Options* is a field in the relevant options structure (MQBO, MQCNO, MQGMO, or MQPMO).
		Corrective action: Specify valid options. Check the description of the *Options* parameter or field to determine which options and combinations of options are valid. If multiple options are being set by adding the individual options together, ensure that the same option is not added twice.
2047	X'07FF'	MQRC_PERSISTENCE_ERROR
		On an MQPUT or MQPUT1 call, the value specified for the *Persistence* field in the message descriptor MQMD is not valid.
		Corrective action: Specify one of the following values: MQPER_PERSISTENT MQPER_NOT_PERSISTENT MQPER_PERSISTENCE_AS_Q_DEF

Completion and reason codes

2048	X'0800'	MQRC_PERSISTENT_NOT_ALLOWED On an MQPUT or MQPUT1 call, the value specified for the *Persistence* field in MQMD (or obtained from the *DefPersistence* queue attribute) specifies MQPER_PERSISTENT, but the queue on which the message is being placed does not support persistent messages. Persistent messages cannot be placed on: • Temporary dynamic queues • Shared queues This reason code can also occur in the *Feedback* field in the message descriptor of a report message; in this case it indicates that the error was encountered by a message channel agent when it attempted to put the message on a remote queue. Corrective action: Specify MQPER_NOT_PERSISTENT if the message is to be placed on a temporary dynamic queue or shared queue. If persistence is required, use a permanent dynamic queue or predefined queue in place of a temporary dynamic queue. Be aware that server applications are recommended to send reply messages (message type MQMT_REPLY) with the same persistence as the original request message (message type MQMT_REQUEST). If the request message is persistent, the reply queue specified in the *ReplyToQ* field in the message descriptor MQMD cannot be a temporary dynamic queue; a permanent dynamic or predefined queue must be used as the reply queue in this situation.
2049	X'0801'	MQRC_PRIORITY_EXCEEDS_MAXIMUM An MQPUT or MQPUT1 call was issued, but the value of the *Priority* field in the message descriptor MQMD exceeds the maximum priority supported by the local queue manager. The message is accepted by the queue manager, but is placed on the queue at the queue manager's maximum priority. The *Priority* field in the message descriptor retains the value specified by the application that put the message. Corrective action: None required, unless this reason code was not expected by the application that put the message.
2050	X'0802'	MQRC_PRIORITY_ERROR An MQPUT or MQPUT1 call was issued, but the value of the *Priority* field in the message descriptor MQMD is not valid. The maximum priority supported by the queue manager is given by the *MaxPriority* queue-manager attribute. Corrective action: Specify a value in the range zero through *MaxPriority*, or the special value MQPRI_PRIORITY_AS_Q_DEF.
2051	X'0803'	MQRC_PUT_INHIBITED MQPUT and MQPUT1 calls are currently inhibited for the queue, or for the queue to which this queue resolves. This reason code can also occur in the *Feedback* field in the message descriptor of a report message; in this case it indicates that the error was encountered by a message channel agent when it attempted to put the message on a remote queue. Corrective action: If the system design allows put requests to be inhibited for short periods, retry the operation later.

2052	X'0804'	MQRC_Q_DELETED An *Hobj* queue handle specified on a call refers to a dynamic queue that has been deleted since the queue was opened. On OS/390, this can also occur with the MQOPEN and MQPUT1 calls if a dynamic queue is being opened, but the queue is in a logically-deleted state. See MQCLOSE for more information about this. Corrective action: Issue an MQCLOSE call to return the handle and associated resources to the system (the MQCLOSE call will succeed in this case). Check the design of the application that caused the error.
2053	X'0805'	MQRC_Q_FULL On an MQPUT or MQPUT1 call, the call failed because the queue is full, that is, it already contains the maximum number of messages possible. This reason code can also occur in the *Feedback* field in the message descriptor of a report message; in this case it indicates that the error was encountered by a message channel agent when it attempted to put the message on a remote queue. Corrective action: Retry the operation later. Consider increasing the maximum depth for this queue, or arranging for more instances of the application to service the queue.
2055	X'0807'	MQRC_Q_NOT_EMPTY An MQCLOSE call was issued for a permanent dynamic queue, but the call failed because the queue is not empty or still in use. One of the following applies: • The MQCO_DELETE option was specified, but there are messages on the queue. • The MQCO_DELETE or MQCO_DELETE_PURGE option was specified, but there are uncommitted get or put calls outstanding against the queue. See the usage notes pertaining to dynamic queues for the MQCLOSE call for more information. This reason code is also returned from a Programmable Command Format (PCF) command to clear or delete a queue, if the queue contains uncommitted messages (or committed messages in the case of delete queue without the purge option). Corrective action: Check why there might be messages on the queue. Be aware that the *CurrentQDepth* queue attribute might be zero even though there are one or more messages on the queue; this can happen if the messages have been retrieved as part of a unit of work that has not yet been committed. If the messages can be discarded, try using the MQCLOSE call with the MQCO_DELETE_PURGE option. Consider retrying the call later.
2056	X'0808'	MQRC_Q_SPACE_NOT_AVAILABLE An MQPUT or MQPUT1 call was issued, but there is no space available for the queue on disk or other storage device. This reason code can also occur in the *Feedback* field in the message descriptor of a report message; in this case it indicates that the error was encountered by a message channel agent when it attempted to put the message on a remote queue. On OS/390, this reason code does not occur. Corrective action: Check whether an application is putting messages in an infinite loop. If not, make more disk space available for the queue.

Completion and reason codes

2057	X'0809'	MQRC_Q_TYPE_ERROR One of the following occurred: • On an MQOPEN call, the *ObjectQMgrName* field in the object descriptor MQOD or object record MQOR specifies the name of a local definition of a remote queue (in order to specify a queue-manager alias), and in that local definition the *RemoteQMgrName* attribute is the name of the local queue manager. However, the *ObjectName* field in MQOD or MQOR specifies the name of a model queue on the local queue manager; this is not allowed. See the *MQSeries Application Programming Guide* for more information. • On an MQPUT1 call, the object descriptor MQOD or object record MQOR specifies the name of a model queue. • On a previous MQPUT or MQPUT1 call, the *ReplyToQ* field in the message descriptor specified the name of a model queue, but a model queue cannot be specified as the destination for reply or report messages. Only the name of a predefined queue, or the name of the *dynamic* queue created from the model queue, can be specified as the destination. In this situation the reason code MQRC_Q_TYPE_ERROR is returned in the *Reason* field of the MQDLH structure when the reply message or report message is placed on the dead-letter queue. Corrective action: Specify a valid queue.
2058	X'080A'	MQRC_Q_MGR_NAME_ERROR On an MQCONN or MQCONNX call, the value specified for the *QMgrName* parameter is not valid or not known. This reason also occurs if the parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) On OS/390 for CICS applications, this reason can occur on *any* call if the original connect specified an incorrect or unrecognized name. This reason code can also occur if an MQSeries client application attempts to connect to a queue manager within an MQSeries-client queue-manager group (see the *QMgrName* parameter of MQCONN), and either: • Queue-manager groups are not supported. • There is no queue-manager group with the specified name. Corrective action: Use an all-blank name if possible, or verify that the name used is valid.

2059	X'080B'	MQRC_Q_MGR_NOT_AVAILABLE
		On an MQCONN or MQCONNX call, the queue manager identified by the *QMgrName* parameter is not available for connection.
		• On OS/390: – For batch applications, this reason can be returned to applications running in non-MQSeries LPARs. – For CICS applications, this reason can occur on *any* call if the original connect specified a queue manager whose name was recognized, but which is not available.
		• On AS/400, this reason can also be returned by the MQOPEN and MQPUT1 calls, when MQHC_DEF_HCONN is specified for the *Hconn* parameter by an application running in compatibility mode.
		This reason code can also occur if an MQSeries client application attempts to connect to a queue manager within an MQSeries-client queue-manager group when none of the queue managers in the group is available for connection (see the *QMgrName* parameter of the MQCONN call).
		This reason code can also occur if the call is issued by an MQSeries client application and there is an error with the client-connection or the corresponding server-connection channel definitions.
		On OS/390, this reason code can also occur if the optional OS/390® client attachment feature has not been installed.
		Corrective action: Ensure that the queue manager has been started. If the connection is from a client application, check the channel definitions.
2061	X'080D'	MQRC_REPORT_OPTIONS_ERROR
		An MQPUT or MQPUT1 call was issued, but the *Report* field in the message descriptor MQMD contains one or more options that are not recognized by the local queue manager. The options that cause this reason code to be returned depend on the destination of the message.
		This reason code can also occur in the *Feedback* field in the MQMD of a report message, or in the *Reason* field in the MQDLH structure of a message on the dead-letter queue; in both cases it indicates that the destination queue manager does not support one or more of the report options specified by the sender of the message.
		Corrective action: Do the following: • Ensure that the *Report* field in the message descriptor is initialized with a value when the message descriptor is declared, or is assigned a value prior to the MQPUT or MQPUT1 call. Specify MQRO_NONE if no report options are required. • Ensure that the report options specified are ones that are documented in the *MQSeries Application Programming Reference*. Remove any report options that are not documented in this book. • If multiple report options are being set by adding the individual report options together, ensure that the same report option is not added twice. • Check that conflicting report options are not specified. For example, do not add both MQRO_EXCEPTION and MQRO_EXCEPTION_WITH_DATA to the *Report* field; only one of these can be specified.
2062	X'080E'	MQRC_SECOND_MARK_NOT_ALLOWED
		An MQGET call was issued specifying the MQGMO_MARK_SKIP_BACKOUT option in the *Options* field of MQGMO, but a message has already been marked within the current unit of work. Only one marked message is allowed within each unit of work.
		This reason code occurs only on OS/390.
		Corrective action: Modify the application so that no more than one message is marked within each unit of work.

Completion and reason codes

2063	X'080F'	MQRC_SECURITY_ERROR An MQCONN, MQCONNX, MQOPEN, MQPUT1, or MQCLOSE call was issued, but it failed because a security error occurred. On OS/390, the security error was returned by the External Security Manager. Corrective action: Note the error from the security manager, and contact your system programmer or security administrator. On AS/400, the FFST log will contain the error information.
2065	X'0811'	MQRC_SELECTOR_COUNT_ERROR On an MQINQ or MQSET call, the *SelectorCount* parameter specifies a value that is not valid. This reason also occurs if the parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Specify a value in the range 0 through 256.
2066	X'0812'	MQRC_SELECTOR_LIMIT_EXCEEDED On an MQINQ or MQSET call, the *SelectorCount* parameter specifies a value that is larger than the maximum supported (256). Corrective action: Reduce the number of selectors specified on the call; the valid range is 0 through 256.
2067	X'0813'	MQRC_SELECTOR_ERROR An MQINQ or MQSET call was issued, but the *Selectors* array contains a selector that is not valid for one of the following reasons: • The selector is not supported or out of range. • The selector is not applicable to the type of object whose attributes are being inquired or set. • The selector is for an attribute that cannot be set. This reason also occurs if the parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Ensure that the value specified for the selector is valid for the object type represented by *Hobj*. For the MQSET call, also ensure that the selector represents an integer attribute that can be set.
2068	X'0814'	MQRC_SELECTOR_NOT_FOR_TYPE On the MQINQ call, one or more selectors in the *Selectors* array is not applicable to the type of the queue whose attributes are being inquired. This reason also occurs when the queue is a cluster queue that resolved to a remote instance of the queue. In this case only a subset of the attributes that are valid for local queues can be inquired. The call completes with MQCC_WARNING, with the attribute values for the inapplicable selectors set as follows: • For integer attributes, the corresponding elements of *IntAttrs* are set to MQIAV_NOT_APPLICABLE. • For character attributes, the appropriate parts of the *CharAttrs* string are set to a character string consisting entirely of asterisks (*). Corrective action: Verify that the selector specified is the one that was intended. If the queue is a cluster queue, specifying one of the MQOO_BROWSE, MQOO_INPUT_*, or MQOO_SET options in addition to MQOO_INQUIRE forces the queue to resolve to the local instance of the queue. However, if there is no local instance of the queue the MQOPEN call fails.

2069	X'0815'	MQRC_SIGNAL_OUTSTANDING
		An MQGET call was issued with either the MQGMO_SET_SIGNAL or MQGMO_WAIT option, but there is already a signal outstanding for the queue handle *Hobj*.
		This reason code occurs only in the following environments: OS/390, Windows 95, Windows 98.
		Corrective action: Check the application logic. If it is necessary to set a signal or wait when there is a signal outstanding for the same queue, a different object handle must be used.
2070	X'0816'	MQRC_SIGNAL_REQUEST_ACCEPTED
		An MQGET call was issued specifying MQGMO_SET_SIGNAL in the *GetMsgOpts* parameter, but no suitable message was available; the call returns immediately. The application can now wait for the signal to be delivered. • On OS/390, the application should wait on the Event Control Block pointed to by the *Signal1* field. • On Windows 95, Windows 98, the application should wait for the signal Windows message to be delivered.
		This reason code occurs only in the following environments: OS/390, Windows 95, Windows 98.
		Corrective action: Wait for the signal; when it is delivered, check the signal to ensure that a message is now available. If it is, reissue the MQGET call. • On OS/390, wait on the ECB pointed to by the *Signal1* field and, when it is posted, check it to ensure that a message is now available. • On Windows 95, Windows 98, the application (thread) should continue executing its message loop.
2071	X'0817'	MQRC_STORAGE_NOT_AVAILABLE
		The call failed because there is insufficient main storage available.
		Corrective action: Ensure that active applications are behaving correctly, for example, that they are not looping unexpectedly. If no problems are found, make more main storage available.
		On OS/390, if no application problems are found, ask your systems programmer to increase the size of the region in which the queue manager runs.
2072	X'0818'	MQRC_SYNCPOINT_NOT_AVAILABLE
		Either MQGMO_SYNCPOINT was specified on an MQGET call or MQPMO_SYNCPOINT was specified on an MQPUT or MQPUT1 call, but the local queue manager was unable to honor the request. If the queue manager does not support units of work, the *SyncPoint* queue-manager attribute will have the value MQSP_NOT_AVAILABLE.
		This reason code can also occur on the MQGET, MQPUT, and MQPUT1 calls when an external unit-of-work coordinator is being used. If that coordinator requires an explicit call to start the unit of work, but the application has not issued that call prior to the MQGET, MQPUT, or MQPUT1 call, reason code MQRC_SYNCPOINT_NOT_AVAILABLE is returned. • On AS/400, this reason codes means that AS/400 Commitment Control is not started, or is unavailable for use by the queue manager. • On OS/390, this reason code does not occur.
		Corrective action: Remove the specification of MQGMO_SYNCPOINT or MQPMO_SYNCPOINT, as appropriate.
		On AS/400, ensure that Commitment Control has been started. If this reason code occurs after Commitment Control has been started, contact your systems programmer.

Completion and reason codes

2075	X'081B'	MQRC_TRIGGER_CONTROL_ERROR On an MQSET call, the value specified for the MQIA_TRIGGER_CONTROL attribute selector is not valid. Corrective action: Specify a valid value.
2076	X'081C'	MQRC_TRIGGER_DEPTH_ERROR On an MQSET call, the value specified for the MQIA_TRIGGER_DEPTH attribute selector is not valid. Corrective action: Specify a value that is greater than zero.
2077	X'081D'	MQRC_TRIGGER_MSG_PRIORITY_ERR On an MQSET call, the value specified for the MQIA_TRIGGER_MSG_PRIORITY attribute selector is not valid. Corrective action: Specify a value in the range zero through the value of *MaxPriority* queue-manager attribute.
2078	X'081E'	MQRC_TRIGGER_TYPE_ERROR On an MQSET call, the value specified for the MQIA_TRIGGER_TYPE attribute selector is not valid. Corrective action: Specify a valid value.
2079	X'081F'	MQRC_TRUNCATED_MSG_ACCEPTED On an MQGET call, the message length was too large to fit into the supplied buffer. The MQGMO_ACCEPT_TRUNCATED_MSG option was specified, so the call completes. The message is removed from the queue (subject to unit-of-work considerations), or, if this was a browse operation, the browse cursor is advanced to this message. The *DataLength* parameter is set to the length of the message before truncation, the *Buffer* parameter contains as much of the message as fits, and the MQMD structure is filled in. Corrective action: None, because the application expected this situation.
2080	X'0820'	MQRC_TRUNCATED_MSG_FAILED On an MQGET call, the message length was too large to fit into the supplied buffer. The MQGMO_ACCEPT_TRUNCATED_MSG option was *not* specified, so the message has not been removed from the queue. If this was a browse operation, the browse cursor remains where it was before this call, but if MQGMO_BROWSE_FIRST was specified, the browse cursor is positioned logically before the highest-priority message on the queue. The *DataLength* field is set to the length of the message before truncation, the *Buffer* parameter contains as much of the message as fits, and the MQMD structure is filled in. Corrective action: Supply a buffer that is at least as large as *DataLength*, or specify MQGMO_ACCEPT_TRUNCATED_MSG if not all of the message data is required.
2082	X'0822'	MQRC_UNKNOWN_ALIAS_BASE_Q An MQOPEN or MQPUT1 call was issued specifying an alias queue as the target, but the *BaseQName* in the alias queue attributes is not recognized as a queue name. This reason code can also occur when *BaseQName* is the name of a cluster queue that cannot be resolved successfully. Corrective action: Correct the queue definitions.

2085	X'0825'	MQRC_UNKNOWN_OBJECT_NAME An MQOPEN or MQPUT1 call was issued, but the object identified by the *ObjectName* and *ObjectQMgrName* fields in the object descriptor MQOD cannot be found. One of the following applies: • The *ObjectQMgrName* field is one of the following: – Blank – The name of the local queue manager – The name of a local definition of a remote queue (a queue-manager alias) in which the *RemoteQMgrName* attribute is the name of the local queue manager but no object with the specified *ObjectName* and *ObjectType* exists on the local queue manager. • The object being opened is a cluster queue that is hosted on a remote queue manager, but the local queue manager does not have a defined route to the remote queue manager. • The object being opened is a queue definition that has QSGDISP(GROUP). Such definitions cannot be used with the MQOPEN and MQPUT1 calls. Corrective action: Specify a valid object name. Ensure that the name is padded to the right with blanks if necessary. If this is correct, check the queue definitions.
2086	X'0826'	MQRC_UNKNOWN_OBJECT_Q_MGR On an MQOPEN or MQPUT1 call, the *ObjectQMgrName* field in the object descriptor MQOD does not satisfy the naming rules for objects. For more information, see the *MQSeries Application Programming Guide*. This reason also occurs if the *ObjectType* field in the object descriptor has the value MQOT_Q_MGR, and the *ObjectQMgrName* field is not blank, but the name specified is not the name of the local queue manager. Corrective action: Specify a valid queue manager name. To refer to the local queue manager, a name consisting entirely of blanks or beginning with a null character can be used. Ensure that the name is padded to the right with blanks or terminated with a null character if necessary.

Completion and reason codes

2087	X'0827'	MQRC_UNKNOWN_REMOTE_Q_MGR
		On an MQOPEN or MQPUT1 call, an error occurred with the queue-name resolution, for one of the following reasons:
		• *ObjectQMgrName* is blank or the name of the local queue manager, *ObjectName* is the name of a local definition of a remote queue (or an alias to one), and one of the following is true: – *RemoteQMgrName* is blank or the name of the local queue manager. Note that this error occurs even if *XmitQName* is not blank. – *XmitQName* is blank, but there is no transmission queue defined with the name of *RemoteQMgrName*, and the *DefXmitQName* queue-manager attribute is blank. – *RemoteQMgrName* and *RemoteQName* specify a cluster queue that cannot be resolved successfully, and the *DefXmitQName* queue-manager attribute is blank.
		• *ObjectQMgrName* is the name of a local definition of a remote queue (containing a queue-manager alias definition), and one of the following is true: – *RemoteQName* is not blank. – *XmitQName* is blank, but there is no transmission queue defined with the name of *RemoteQMgrName*, and the *DefXmitQName* queue-manager attribute is blank.
		• *ObjectQMgrName* is not: – Blank – The name of the local queue manager – The name of a transmission queue – The name of a queue-manager alias definition (that is, a local definition of a remote queue with a blank *RemoteQName*) but the *DefXmitQName* queue-manager attribute is blank.
		• *ObjectQMgrName* is the name of a model queue.
		• The queue name is resolved through a cell directory. However, there is no queue defined with the same name as the remote queue manager name obtained from the cell directory, and the *DefXmitQName* queue-manager attribute is blank.
		Corrective action: Check the values specified for *ObjectQMgrName* and *ObjectName*. If these are correct, check the queue definitions.
2090	X'082A'	MQRC_WAIT_INTERVAL_ERROR
		On the MQGET call, the value specified for the *WaitInterval* field in the *GetMsgOpts* parameter is not valid.
		Corrective action: Specify a value greater than or equal to zero, or the special value MQWI_UNLIMITED if an indefinite wait is required.
2091	X'082B'	MQRC_XMIT_Q_TYPE_ERROR
		On an MQOPEN or MQPUT1 call, a message is to be sent to a remote queue manager. The *ObjectName* or *ObjectQMgrName* field in the object descriptor specifies the name of a local definition of a remote queue but one of the following applies to the *XmitQName* attribute of the definition: • *XmitQName* is not blank, but specifies a queue that is not a local queue • *XmitQName* is blank, but *RemoteQMgrName* specifies a queue that is not a local queue
		This reason also occurs if the queue name is resolved through a cell directory, and the remote queue manager name obtained from the cell directory is the name of a queue, but this is not a local queue.
		Corrective action: Check the values specified for *ObjectName* and *ObjectQMgrName*. If these are correct, check the queue definitions. For more information on transmission queues, see the *MQSeries Application Programming Guide*.

2092	X'082C'	MQRC_XMIT_Q_USAGE_ERROR
		On an MQOPEN or MQPUT1 call, a message is to be sent to a remote queue manager, but one of the following occurred:
		• *ObjectQMgrName* specifies the name of a local queue, but it does not have a *Usage* attribute of MQUS_TRANSMISSION.
		• The *ObjectName* or *ObjectQMgrName* field in the object descriptor specifies the name of a local definition of a remote queue but one of the following applies to the *XmitQName* attribute of the definition: – *XmitQName* is not blank, but specifies a queue that does not have a *Usage* attribute of MQUS_TRANSMISSION – *XmitQName* is blank, but *RemoteQMgrName* specifies a queue that does not have a *Usage* attribute of MQUS_TRANSMISSION
		• The queue name is resolved through a cell directory, and the remote queue manager name obtained from the cell directory is the name of a local queue, but it does not have a *Usage* attribute of MQUS_TRANSMISSION.
		Corrective action: Check the values specified for *ObjectName* and *ObjectQMgrName*. If these are correct, check the queue definitions. For more information on transmission queues, see the *MQSeries Application Programming Guide*.
2093	X'082D'	MQRC_NOT_OPEN_FOR_PASS_ALL
		An MQPUT call was issued with the MQPMO_PASS_ALL_CONTEXT option specified in the *PutMsgOpts* parameter, but the queue had not been opened with the MQOO_PASS_ALL_CONTEXT option.
		Corrective action: Specify MQOO_PASS_ALL_CONTEXT (or another option that implies it) when the queue is opened.
2094	X'082E'	MQRC_NOT_OPEN_FOR_PASS_IDENT
		An MQPUT call was issued with the MQPMO_PASS_IDENTITY_CONTEXT option specified in the *PutMsgOpts* parameter, but the queue had not been opened with the MQOO_PASS_IDENTITY_CONTEXT option.
		Corrective action: Specify MQOO_PASS_IDENTITY_CONTEXT (or another option that implies it) when the queue is opened.
2095	X'082F'	MQRC_NOT_OPEN_FOR_SET_ALL
		An MQPUT call was issued with the MQPMO_SET_ALL_CONTEXT option specified in the *PutMsgOpts* parameter, but the queue had not been opened with the MQOO_SET_ALL_CONTEXT option.
		Corrective action: Specify MQOO_SET_ALL_CONTEXT when the queue is opened.
2096	X'0830'	MQRC_NOT_OPEN_FOR_SET_IDENT
		An MQPUT call was issued with the MQPMO_SET_IDENTITY_CONTEXT option specified in the *PutMsgOpts* parameter, but the queue had not been opened with the MQOO_SET_IDENTITY_CONTEXT option.
		Corrective action: Specify MQOO_SET_IDENTITY_CONTEXT (or another option that implies it) when the queue is opened.

Completion and reason codes

2097	X'0831'	MQRC_CONTEXT_HANDLE_ERROR On an MQPUT or MQPUT1 call, MQPMO_PASS_IDENTITY_CONTEXT or MQPMO_PASS_ALL_CONTEXT was specified, but the handle specified in the *Context* field of the *PutMsgOpts* parameter is either not a valid queue handle, or it is a valid queue handle but the queue was not opened with MQOO_SAVE_ALL_CONTEXT. Corrective action: Specify MQOO_SAVE_ALL_CONTEXT when the queue referred to is opened.
2098	X'0832'	MQRC_CONTEXT_NOT_AVAILABLE On an MQPUT or MQPUT1 call, MQPMO_PASS_IDENTITY_CONTEXT or MQPMO_PASS_ALL_CONTEXT was specified, but the queue handle specified in the *Context* field of the *PutMsgOpts* parameter has no context associated with it. This arises if no message has yet been successfully retrieved with the queue handle referred to, or if the last successful MQGET call was a browse. This condition does not arise if the message that was last retrieved had no context associated with it. On OS/390, if a message is received by a message channel agent that is putting messages with the authority of the user identifier in the message, this code is returned in the *Feedback* field of an exception report if the message has no context associated with it. Corrective action: Ensure that a successful nonbrowse get call has been issued with the queue handle referred to.
2099	X'0833'	MQRC_SIGNAL1_ERROR An MQGET call was issued, specifying MQGMO_SET_SIGNAL in the *GetMsgOpts* parameter, but the *Signal1* field is not valid. • On OS/390, the address contained in the *Signal1* field is not valid, or points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • On Windows 95, Windows 98, the window handle in the *Signal1* field is not valid. This reason code occurs only in the following environments: OS/390, Windows 95, Windows 98. Corrective action: Correct the setting of the *Signal1* field.
2100	X'0834'	MQRC_OBJECT_ALREADY_EXISTS An MQOPEN call was issued to create a dynamic queue, but a queue with the same name as the dynamic queue already exists. On OS/390, a rare "race condition" can also give rise to this reason code; see the description of reason code MQRC_NAME_IN_USE for more details. Corrective action: If supplying a dynamic queue name in full, ensure that it obeys the naming conventions for dynamic queues; if it does, either supply a different name, or delete the existing queue if it is no longer required. Alternatively, allow the queue manager to generate the name. If the queue manager is generating the name (either in part or in full), reissue the MQOPEN call.

2101	X'0835'	MQRC_OBJECT_DAMAGED
		The object accessed by the call is damaged and cannot be used. For example, this may be because the definition of the object in main storage is not consistent, or because it differs from the definition of the object on disk, or because the definition on disk cannot be read.
		The object cannot be used until the problem is corrected. The object can be deleted, although it may not be possible to delete the associated user space.
		On OS/390, this reason code does not occur.
		Corrective action: It may be necessary to stop and restart the queue manager, or to restore the queue-manager data from back-up storage.
		On Compaq (DIGITAL) OpenVMS, OS/2, AS/400, Tandem NonStop Kernel, and UNIX systems, consult the FFST record to obtain more detail about the problem.
2102	X'0836'	MQRC_RESOURCE_PROBLEM
		There are insufficient system resources to complete the call successfully.
		Corrective action: Run the application when the machine is less heavily loaded. • On OS/390, check the operator console for messages that may provide additional information. • On Compaq (DIGITAL) OpenVMS, OS/2, AS/400, Tandem NonStop Kernel, and UNIX systems, consult the FFST record to obtain more detail about the problem.
2103	X'0837'	MQRC_ANOTHER_Q_MGR_CONNECTED
		An MQCONN or MQCONNX call was issued, but the thread or process is already connected to a different queue manager. The thread or process can connect to only one queue manager at a time.
		On OS/390, this reason code does not occur.
		On Windows NT, MTS objects do not receive this reason code, as connections to other queue managers are allowed.
		Corrective action: Use the MQDISC call to disconnect from the queue manager that is already connected, and then issue the MQCONN or MQCONNX call to connect to the new queue manager.
		Disconnecting from the existing queue manager will close any queues that are currently open; it is recommended that any uncommitted units of work should be committed or backed out before the MQDISC call is issued.

Completion and reason codes

2104	X'0838'	MQRC_UNKNOWN_REPORT_OPTION An MQPUT or MQPUT1 call was issued, but the *Report* field in the message descriptor MQMD contains one or more options that are not recognized by the local queue manager. The options are accepted. The options that cause this reason code to be returned depend on the destination of the message. Corrective action: If this reason code is expected, no corrective action is required. If this reason code is not expected, do the following: • Ensure that the *Report* field in the message descriptor is initialized with a value when the message descriptor is declared, or is assigned a value prior to the MQPUT or MQPUT1 call. • Ensure that the report options specified are ones that are documented in the *MQSeries Application Programming Reference*. Remove any report options that are not documented in this book. • If multiple report options are being set by adding the individual report options together, ensure that the same report option is not added twice. • Check that conflicting report options are not specified. For example, do not add both MQRO_EXCEPTION and MQRO_EXCEPTION_WITH_DATA to the *Report* field; only one of these can be specified.
2105	X'0839'	MQRC_STORAGE_CLASS_ERROR The MQPUT or MQPUT1 call was issued, but the storage-class object defined for the queue does not exist. This reason code occurs only on OS/390. Corrective action: Create the storage-class object required by the queue, or modify the queue definition to use an existing storage class. The name of the storage-class object used by the queue is given by the `StorageClass` queue attribute.
2106	X'083A'	MQRC_COD_NOT_VALID_FOR_XCF_Q An MQPUT or MQPUT1 call was issued, but the *Report* field in the message descriptor MQMD specifies one of the MQRO_COD_* options and the target queue is an XCF queue. MQRO_COD_* options cannot be specified for XCF queues. This reason code occurs only on OS/390. Corrective action: Remove the relevant MQRO_COD_* option.
2107	X'083B'	MQRC_XWAIT_CANCELED An MQXWAIT call was issued, but the call has been canceled because a STOP CHINIT command has been issued (or the queue manager has been stopped, which causes the same effect). Refer to the *MQSeries Intercommunication* book for details of the MQXWAIT call. This reason code occurs only on OS/390. Corrective action: Tidy up and terminate.

2108	X'083C'	MQRC_XWAIT_ERROR
		An MQXWAIT call was issued, but the invocation was not valid for one of the following reasons:
		• The wait descriptor MQXWD contains data that is not valid.
		• The linkage stack level is not valid.
		• The addressing mode is not valid.
		• There are too many wait events outstanding.
		This reason code occurs only on OS/390.
		Corrective action: Obey the rules for using the MQXWAIT call. Refer to the *MQSeries Intercommunication* book for details of this call.
2109	X'083D'	MQRC_SUPPRESSED_BY_EXIT
		On any call other than MQCONN or MQDISC, the API crossing exit suppressed the call.
		This reason code occurs only on OS/390.
		Corrective action: Obey the rules for MQI calls that the exit enforces. To find out the rules, see the writer of the exit.
2110	X'083E'	MQRC_FORMAT_ERROR
		An MQGET call was issued with the MQGMO_CONVERT option specified in the *GetMsgOpts* parameter, but the message cannot be converted successfully due to an error associated with the message format. Possible errors include:
		• The format name in the message is MQFMT_NONE.
		• A user-written exit with the name specified by the *Format* field in the message cannot be found.
		• The message contains data that is not consistent with the format definition.
		The message is returned unconverted to the application issuing the MQGET call, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING.
		If the message consists of several parts, each of which is described by its own *CodedCharSetId* and *Encoding* fields (for example, a message with format name MQFMT_DEAD_LETTER_HEADER), some parts may be converted and other parts not converted. However, the values returned in the various *CodedCharSetId* and *Encoding* fields always correctly describe the relevant message data.
		Corrective action: Check the format name that was specified when the message was put. If this is not one of the built-in formats, check that a suitable exit with the same name as the format is available for the queue manager to load. Verify that the data in the message corresponds to the format expected by the exit.

Completion and reason codes

2111	X'083F'	MQRC_SOURCE_CCSID_ERROR
		The coded character-set identifier from which character data is to be converted is not valid or not supported.
		This can occur on the MQGET call when the MQGMO_CONVERT option is included in the *GetMsgOpts* parameter; the coded character-set identifier in error is the *CodedCharSetId* field in the message being retrieved. In this case, the message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING.
		This reason can also occur on the MQGET call when the message contains one or more MQSeries header structures (MQCIH, MQDLH, MQIIH, MQRMH), and the *CodedCharSetId* field in the message specifies a character set that does not have SBCS characters for the characters that are valid in queue names. MQSeries header structures containing such characters are not valid, and so the message is returned unconverted. The Unicode character set UCS-2 is an example of such a character set.
		If the message consists of several parts, each of which is described by its own *CodedCharSetId* and *Encoding* fields (for example, a message with format name MQFMT_DEAD_LETTER_HEADER), some parts may be converted and other parts not converted. However, the values returned in the various *CodedCharSetId* and *Encoding* fields always correctly describe the relevant message data.
		This reason can also occur on the MQXCNVC call; the coded character-set identifier in error is the *SourceCCSID* parameter. Either the *SourceCCSID* parameter specifies a value that is not valid or not supported, or the *SourceCCSID* parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		Corrective action: Check the character-set identifier that was specified when the message was put, or that was specified for the *SourceCCSID* parameter on the MQXCNVC call. If this is correct, check that it is one for which queue-manager conversion is supported. If queue-manager conversion is not supported for the specified character set, conversion must be carried out by the application.
2112	X'0840'	MQRC_SOURCE_INTEGER_ENC_ERROR
		On an MQGET call, with the MQGMO_CONVERT option included in the *GetMsgOpts* parameter, the *Encoding* value in the message being retrieved specifies an integer encoding that is not recognized. The message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING.
		If the message consists of several parts, each of which is described by its own *CodedCharSetId* and *Encoding* fields (for example, a message with format name MQFMT_DEAD_LETTER_HEADER), some parts may be converted and other parts not converted. However, the values returned in the various *CodedCharSetId* and *Encoding* fields always correctly describe the relevant message data.
		This reason code can also occur on the MQXCNVC call, when the *Options* parameter contains an unsupported MQDCC_SOURCE_* value, or when MQDCC_SOURCE_ENC_UNDEFINED is specified for a UCS2 code page.
		Corrective action: Check the integer encoding that was specified when the message was put. If this is correct, check that it is one for which queue-manager conversion is supported. If queue-manager conversion is not supported for the required integer encoding, conversion must be carried out by the application.

2113	X'0841'	MQRC_SOURCE_DECIMAL_ENC_ERROR
		On an MQGET call with the MQGMO_CONVERT option included in the *GetMsgOpts* parameter, the *Encoding* value in the message being retrieved specifies a decimal encoding that is not recognized. The message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING. If the message consists of several parts, each of which is described by its own *CodedCharSetId* and *Encoding* fields (for example, a message with format name MQFMT_DEAD_LETTER_HEADER), some parts may be converted and other parts not converted. However, the values returned in the various *CodedCharSetId* and *Encoding* fields always correctly describe the relevant message data. Corrective action: Check the decimal encoding that was specified when the message was put. If this is correct, check that it is one for which queue-manager conversion is supported. If queue-manager conversion is not supported for the required decimal encoding, conversion must be carried out by the application.
2114	X'0842'	MQRC_SOURCE_FLOAT_ENC_ERROR
		On an MQGET call, with the MQGMO_CONVERT option included in the *GetMsgOpts* parameter, the *Encoding* value in the message being retrieved specifies a floating-point encoding that is not recognized. The message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING. If the message consists of several parts, each of which is described by its own *CodedCharSetId* and *Encoding* fields (for example, a message with format name MQFMT_DEAD_LETTER_HEADER), some parts may be converted and other parts not converted. However, the values returned in the various *CodedCharSetId* and *Encoding* fields always correctly describe the relevant message data. Corrective action: Check the floating-point encoding that was specified when the message was put. If this is correct, check that it is one for which queue-manager conversion is supported. If queue-manager conversion is not supported for the required floating-point encoding, conversion must be carried out by the application.

Completion and reason codes

2115	X'0843'	**MQRC_TARGET_CCSID_ERROR**
		The coded character-set identifier to which character data is to be converted is not valid or not supported.
		This can occur on the MQGET call when the MQGMO_CONVERT option is included in the *GetMsgOpts* parameter; the coded character-set identifier in error is the *CodedCharSetId* field in the *MsgDesc* parameter. In this case, the message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING.
		This reason can also occur on the MQGET call when the message contains one or more MQSeries header structures (MQCIH, MQDLH, MQIIH, MQRMH), and the *CodedCharSetId* field in the *MsgDesc* parameter specifies a character set that does not have SBCS characters for the characters that are valid in queue names. The Unicode character set UCS-2 is an example of such a character set.
		This reason can also occur on the MQXCNVC call; the coded character-set identifier in error is the *TargetCCSID* parameter. Either the *TargetCCSID* parameter specifies a value that is not valid or not supported, or the *TargetCCSID* parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		Corrective action: Check the character-set identifier that was specified for the *CodedCharSetId* field in the *MsgDesc* parameter on the MQGET call, or that was specified for the *SourceCCSID* parameter on the MQXCNVC call. If this is correct, check that it is one for which queue-manager conversion is supported. If queue-manager conversion is not supported for the specified character set, conversion must be carried out by the application.
2116	X'0844'	**MQRC_TARGET_INTEGER_ENC_ERROR**
		On an MQGET call with the MQGMO_CONVERT option included in the *GetMsgOpts* parameter, the *Encoding* value in the *MsgDesc* parameter specifies an integer encoding that is not recognized. The message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message being retrieved, and the call completes with MQCC_WARNING.
		This reason code can also occur on the MQXCNVC call, when the *Options* parameter contains an unsupported MQDCC_TARGET_* value, or when MQDCC_TARGET_ENC_UNDEFINED is specified for a UCS2 code page.
		Corrective action: Check the integer encoding that was specified. If this is correct, check that it is one for which queue-manager conversion is supported. If queue-manager conversion is not supported for the required integer encoding, conversion must be carried out by the application.
2117	X'0845'	**MQRC_TARGET_DECIMAL_ENC_ERROR**
		On an MQGET call with the MQGMO_CONVERT option included in the *GetMsgOpts* parameter, the *Encoding* value in the *MsgDesc* parameter specifies a decimal encoding that is not recognized. The message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING.
		Corrective action: Check the decimal encoding that was specified. If this is correct, check that it is one for which queue-manager conversion is supported. If queue-manager conversion is not supported for the required decimal encoding, conversion must be carried out by the application.

2118	X'0846'	MQRC_TARGET_FLOAT_ENC_ERROR
		On an MQGET call with the MQGMO_CONVERT option included in the *GetMsgOpts* parameter, the *Encoding* value in the *MsgDesc* parameter specifies a floating-point encoding that is not recognized. The message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING. Corrective action: Check the floating-point encoding that was specified. If this is correct, check that it is one for which queue-manager conversion is supported. If queue-manager conversion is not supported for the required floating-point encoding, conversion must be carried out by the application.
2119	X'0847'	MQRC_NOT_CONVERTED An MQGET call was issued with the MQGMO_CONVERT option specified in the *GetMsgOpts* parameter, but an error occurred during conversion of the data in the message. The message data is returned unconverted, the values of the *CodedCharSetId* and *Encoding* fields in the *MsgDesc* parameter are set to those of the message returned, and the call completes with MQCC_WARNING. If the message consists of several parts, each of which is described by its own *CodedCharSetId* and *Encoding* fields (for example, a message with format name MQFMT_DEAD_LETTER_HEADER), some parts may be converted and other parts not converted. However, the values returned in the various *CodedCharSetId* and *Encoding* fields always correctly describe the relevant message data. This error may also indicate that a parameter to the data-conversion service is not supported. Corrective action: Check that the message data is correctly described by the *Format*, *CodedCharSetId* and *Encoding* parameters that were specified when the message was put. Also check that these values, and the *CodedCharSetId* and *Encoding* specified in the *MsgDesc* parameter on the MQGET call, are supported for queue-manager conversion. If the required conversion is not supported, conversion must be carried out by the application.
2120	X'0848'	MQRC_CONVERTED_MSG_TOO_BIG On an MQGET call with the MQGMO_CONVERT option included in the *GetMsgOpts* parameter, the message data expanded during data conversion and exceeded the size of the buffer provided by the application. However, the message had already been removed from the queue because prior to conversion the message data could be accommodated in the application buffer without truncation. The message is returned unconverted, with the *CompCode* parameter of the MQGET call set to MQCC_WARNING. If the message consists of several parts, each of which is described by its own character-set and encoding fields (for example, a message with format name MQFMT_DEAD_LETTER_HEADER), some parts may be converted and other parts not converted. However, the values returned in the various character-set and encoding fields always correctly describe the relevant message data. This reason can also occur on the MQXCNVC call, when the *TargetBuffer* parameter is too small too accommodate the converted string, and the string has been truncated to fit in the buffer. The length of valid data returned is given by the *DataLength* parameter; in the case of a DBCS string or mixed SBCS/DBCS string, this length may be *less than* the length of *TargetBuffer*. Corrective action: For the MQGET call, check that the exit is converting the message data correctly and setting the output length *DataLength* to the appropriate value. If it is, the application issuing the MQGET call must provide a larger buffer for the *Buffer* parameter. For the MQXCNVC call, if the string must be converted without truncation, provide a larger output buffer.

Completion and reason codes

| 2121 | X'0849' | MQRC_NO_EXTERNAL_PARTICIPANTS

An MQBEGIN call was issued to start a unit of work coordinated by the queue manager, but no participating resource managers have been registered with the queue manager. As a result, only changes to MQSeries resources can be coordinated by the queue manager in the unit of work.

This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT.

Corrective action: If the application does not require non-MQSeries resources to participate in the unit of work, this reason code can be ignored or the MQBEGIN call removed. Otherwise consult your system support programmer to determine why the required resource managers have not been registered with the queue manager; the queue manager's configuration file may be in error. |
|------|---------|---|
| 2122 | X'084A' | MQRC_PARTICIPANT_NOT_AVAILABLE

An MQBEGIN call was issued to start a unit of work coordinated by the queue manager, but one or more of the participating resource managers that had been registered with the queue manager is not available. As a result, changes to those resources cannot be coordinated by the queue manager in the unit of work.

This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT.

Corrective action: If the application does not require non-MQSeries resources to participate in the unit of work, this reason code can be ignored. Otherwise consult your system support programmer to determine why the required resource managers are not available. The resource manager may have been halted temporarily, or there may be an error in the queue manager's configuration file. |
| 2123 | X'084B' | MQRC_OUTCOME_MIXED

The queue manager is acting as the unit-of-work coordinator for a unit of work that involves other resource managers, but one of the following occurred:
• An MQCMIT or MQDISC call was issued to commit the unit of work, but one or more of the participating resource managers backed-out the unit of work instead of committing it. As a result, the outcome of the unit of work is mixed.
• An MQBACK call was issued to back out a unit of work, but one or more of the participating resource managers had already committed the unit of work.

This reason code occurs in the following environments: AIX, HP-UX, OS/2, Sun Solaris, Windows NT.

Corrective action: Examine the queue-manager error logs for messages relating to the mixed outcome; these messages identify the resource managers that are affected. Use procedures local to the affected resource managers to resynchronize the resources.

This reason code does not prevent the application initiating further units of work. |

2124	X'084C'	MQRC_OUTCOME_PENDING The queue manager is acting as the unit-of-work coordinator for a unit of work that involves other resource managers, and an MQCMIT or MQDISC call was issued to commit the unit of work, but one or more of the participating resource managers has not confirmed that the unit of work was committed successfully. The completion of the commit operation will happen at some point in the future, but there remains the possibility that the outcome will be mixed. This reason code occurs in the following environments: AIX, HP-UX, OS/2, Sun Solaris, Windows NT. Corrective action: Use the normal error-reporting mechanisms to determine whether the outcome was mixed. If it was, take appropriate action to resynchronize the resources. This reason code does not prevent the application initiating further units of work.
2125	X'084D'	MQRC_BRIDGE_STARTED The IMS bridge has been started. Corrective action: None. This reason code is only used to identify the corresponding event message.
2126	X'084E'	MQRC_BRIDGE_STOPPED The IMS bridge has been stopped. Corrective action: None. This reason code is only used to identify the corresponding event message.
2127	X'084F'	MQRC_ADAPTER_STORAGE_SHORTAGE On an MQCONN call, the adapter was unable to acquire storage. This reason code occurs only on OS/390. Corrective action: Notify the system programmer. The system programmer should determine why the system is short on storage, and take appropriate action, for example, increase the region size on the step or job card.
2128	X'0850'	MQRC_UOW_IN_PROGRESS An MQBEGIN call was issued to start a unit of work coordinated by the queue manager, but a unit of work is already in existence for the connection handle specified. This may be a global unit of work started by a previous MQBEGIN call, or a unit of work that is local to the queue manager or one of the cooperating resource managers. No more than one unit of work can exist concurrently for a connection handle. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT. Corrective action: Review the application logic to determine why there is a unit of work already in existence. Move the MQBEGIN call to the appropriate place in the application.
2129	X'0851'	MQRC_ADAPTER_CONN_LOAD_ERROR On an MQCONN call, the connection handling module (CSQBCON for batch and CSQQCONN for IMS) could not be loaded, so the adapter could not link to it. This reason code occurs only on OS/390. Corrective action: Ensure that the correct library concatenation has been specified in the batch application program execution JCL, and in the MQSeries startup JCL.

Completion and reason codes

2130	X'0852'	MQRC_ADAPTER_SERV_LOAD_ERROR
		On an MQI call, the batch adapter could not load the API service module CSQBSRV, and so could not link to it.
		This reason code occurs only on OS/390.
		Corrective action: Ensure that the correct library concatenation has been specified in the batch application program execution JCL, and in the MQSeries startup JCL.
2131	X'0853'	MQRC_ADAPTER_DEFS_ERROR
		On an MQCONN call, the subsystem definition module (CSQBDEFV for batch and CSQQDEFV for IMS) does not contain the required control block identifier.
		This reason code occurs only on OS/390.
		Corrective action: Check your library concatenation. If this is correct, check that the CSQBDEFV or CSQQDEFV module contains the required subsystem ID.
2132	X'0854'	MQRC_ADAPTER_DEFS_LOAD_ERROR
		On an MQCONN call, the subsystem definition module (CSQBDEFV for batch and CSQQDEFV for IMS) could not be loaded.
		This reason code occurs only on OS/390.
		Corrective action: Ensure that the correct library concatenation has been specified in the application program execution JCL, and in the MQSeries startup JCL.
2133	X'0855'	MQRC_ADAPTER_CONV_LOAD_ERROR
		On an MQGET call, the adapter (batch or IMS) could not load the data conversion services modules.
		This reason code occurs only on OS/390.
		Corrective action: Ensure that the correct library concatenation has been specified in the batch application program execution JCL, and in the MQSeries startup JCL.
2134	X'0856'	MQRC_BO_ERROR
		On an MQBEGIN call, the begin-options structure MQBO is not valid, for one of the following reasons: • The *StrucId* field is not MQBO_STRUC_ID. • The *Version* field is not MQBO_VERSION_1. • The parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The queue manager cannot copy the changed structure to application storage, even though the call is successful. This can occur, for example, if the pointer points to read-only storage.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Correct the definition of the MQBO structure. Ensure that required input fields are set correctly.

2135	X'0857'	MQRC_DH_ERROR
		On an MQPUT or MQPUT1 call, the distribution header structure MQDH in the message data is not valid.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Correct the definition of the MQDH structure. Ensure that the fields are set correctly.
2136	X'0858'	MQRC_MULTIPLE_REASONS
		An MQOPEN, MQPUT or MQPUT1 call was issued to open a distribution list or put a message to a distribution list, but the result of the call was not the same for all of the destinations in the list. One of the following applies: • The call succeeded for some of the destinations but not others. The completion code is MQCC_WARNING in this case. • The call failed for all of the destinations, but for differing reasons. The completion code is MQCC_FAILED in this case.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Examine the MQRR response records to identify the destinations for which the call failed, and the reason for the failure. Ensure that sufficient response records are provided by the application on the call to enable the error(s) to be determined. For the MQPUT1 call, the response records must be specified using the MQOD structure, and not the MQPMO structure.
2137	X'0859'	MQRC_OPEN_FAILED
		A queue or other MQSeries object could not be opened successfully, for one of the following reasons: • An MQCONN or MQCONNX call was issued, but the queue manager was unable to open an object that is used internally by the queue manager. As a result, processing cannot continue. The error log will contain the name of the object that could not be opened. • An MQPUT call was issued to put a message to a distribution list, but the message could not be sent to the destination to which this reason code applies because that destination was not opened successfully by the MQOPEN call. This reason occurs only in the *Reason* field of the MQRR response record.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Do one of the following: • If the error occurred on the MQCONN or MQCONNX call, ensure that the required objects exist by running the following command and then retrying the application: `STRMQM -c qmgr` where qmgr should be replaced by the name of the queue manager. • If the error occurred on the MQPUT call, examine the MQRR response records specified on the MQOPEN call to determine the reason that the queue failed to open. Ensure that sufficient response records are provided by the application on the call to enable the error(s) to be determined.

Completion and reason codes

2138	X'085A'	MQRC_ADAPTER_DISC_LOAD_ERROR
		On an MQDISC call, the disconnect handling module (CSQBDSC for batch and CSQQDISC for IMS) could not be loaded, so the adapter could not link to it.
		This reason code occurs only on OS/390.
		Corrective action: Ensure that the correct library concatenation has been specified in the application program execution JCL, and in the MQSeries startup JCL. Any uncommitted changes in a unit of work should be backed out. A unit of work that is coordinated by the queue manager is backed out automatically.
2139	X'085B'	MQRC_CNO_ERROR
		On an MQCONNX call, the connect-options structure MQCNO is not valid, for one of the following reasons: • The *StrucId* field is not MQCNO_STRUC_ID. • The *Version* field specifies a value that is not valid or not supported. • The parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The queue manager cannot copy the changed structure to application storage, even though the call is successful. This can occur, for example, if the parameter pointer points to read-only storage.
		This reason code occurs in the following environments: AIX, HP-UX, OS/390, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Correct the definition of the MQCNO structure. Ensure that required input fields are set correctly.
2140	X'085C'	MQRC_CICS_WAIT_FAILED
		On any MQI call, the CICS adapter issued an EXEC CICS WAIT request, but the request was rejected by CICS.
		This reason code occurs only on OS/390.
		Corrective action: Examine the CICS trace data for actual response codes. The most likely cause is that the task has been canceled by the operator or by the system.
2141	X'085D'	MQRC_DLH_ERROR
		On an MQPUT or MQPUT1 call, the dead letter header structure MQDLH in the message data is not valid.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Correct the definition of the MQDLH structure. Ensure that the fields are set correctly.
2142	X'085E'	MQRC_HEADER_ERROR
		The MQPUT or MQPUT1 call was used to put a message containing an MQSeries header structure, but the header structure is not valid.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Correct the definition of the MQSeries header structure. Ensure that the fields are set correctly.

2143	X'085F'	MQRC_SOURCE_LENGTH_ERROR
		On the MQXCNVC call, the *SourceLength* parameter specifies a length that is less than zero or not consistent with the string's character set or content (for example, the character set is a double-byte character set, but the length is not a multiple of two). This reason also occurs if the *SourceLength* parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		This reason code can also occur on the MQGET call when the MQGMO_CONVERT option is specified. In this case it indicates that the MQRC_SOURCE_LENGTH_ERROR reason was returned by an MQXCNVC call issued by the data conversion exit.
		Corrective action: Specify a length that is zero or greater. If the reason code occurs on the MQGET call, check that the logic in the data-conversion exit is correct.
2144	X'0860'	MQRC_TARGET_LENGTH_ERROR
		On the MQXCNVC call, the *TargetLength* parameter is not valid for one of the following reasons: • *TargetLength* is less than zero. • The *TargetLength* parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The MQDCC_FILL_TARGET_BUFFER option is specified, but the value of *TargetLength* is such that the target buffer cannot be filled completely with valid characters. This can occur when *TargetCCSID* is a pure DBCS character set (such as UCS-2), but *TargetLength* specifies a length that is an odd number of bytes.
		This reason code can also occur on the MQGET call when the MQGMO_CONVERT option is specified. In this case it indicates that the MQRC_TARGET_LENGTH_ERROR reason was returned by an MQXCNVC call issued by the data conversion exit.
		Corrective action: Specify a length that is zero or greater. If the MQDCC_FILL_TARGET_BUFFER option is specified, and *TargetCCSID* is a pure DBCS character set, ensure that *TargetLength* specifies a length that is a multiple of two.
		If the reason code occurs on the MQGET call, check that the logic in the data-conversion exit is correct.
2145	X'0861'	MQRC_SOURCE_BUFFER_ERROR
		On the MQXCNVC call, the *SourceBuffer* parameter pointer is not valid, or points to storage that cannot be accessed for the entire length specified by *SourceLength*. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		This reason code can also occur on the MQGET call when the MQGMO_CONVERT option is specified. In this case it indicates that the MQRC_SOURCE_BUFFER_ERROR reason was returned by an MQXCNVC call issued by the data conversion exit.
		Corrective action: Specify a valid buffer. If the reason code occurs on the MQGET call, check that the logic in the data-conversion exit is correct.

Completion and reason codes

2146	X'0862'	MQRC_TARGET_BUFFER_ERROR
		On the MQXCNVC call, the *TargetBuffer* parameter pointer is not valid, or points to read-only storage, or to storage that cannot be accessed for the entire length specified by *TargetLength*. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		This reason code can also occur on the MQGET call when the MQGMO_CONVERT option is specified. In this case it indicates that the MQRC_TARGET_BUFFER_ERROR reason was returned by an MQXCNVC call issued by the data conversion exit.
		Corrective action: Specify a valid buffer. If the reason code occurs on the MQGET call, check that the logic in the data-conversion exit is correct.
2148	X'0864'	MQRC_IIH_ERROR
		On an MQPUT or MQPUT1 call, the IMS information header structure MQIIH in the message data is not valid.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Correct the definition of the MQIIH structure. Ensure that the fields are set correctly.
2149	X'0865'	MQRC_PCF_ERROR
		An MQPUT or MQPUT1 call was issued to put a message containing PCF data, but the length of the message does not equal the sum of the lengths of the PCF structures present in the message. This can occur for messages with the following format names: MQFMT_ADMIN MQFMT_EVENT MQFMT_PCF
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Ensure that the length of the message specified on the MQPUT or MQPUT1 call equals the sum of the lengths of the PCF structures contained within the message data.
2150	X'0866'	MQRC_DBCS_ERROR
		On the MQXCNVC call, the *SourceCCSID* parameter specifies the coded character-set identifier of a double-byte character set (DBCS), but the *SourceBuffer* parameter does not contain a valid DBCS string. This may be because the string contains characters that are not valid DBCS characters, or because the string is a mixed SBCS/DBCS string and the shift-out/shift-in characters are not correctly paired.
		This reason code can also occur on the MQGET call when the MQGMO_CONVERT option is specified. In this case it indicates that the MQRC_DBCS_ERROR reason was returned by an MQXCNVC call issued by the data conversion exit.
		Corrective action: Specify a valid string.
		If the reason code occurs on the MQGET call, check that the data in the message is valid, and that the logic in the data-conversion exit is correct.

Completion and reason codes

2152	X'0868'	MQRC_OBJECT_NAME_ERROR An MQOPEN or MQPUT1 call was issued to open a distribution list (that is, the *RecsPresent* field in MQOD is greater than zero), but the *ObjectName* field is neither blank nor the null string. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: If it is intended to open a distribution list, set the *ObjectName* field to blanks or the null string. If it is not intended to open a distribution list, set the *RecsPresent* field to zero.
2153	X'0869'	MQRC_OBJECT_Q_MGR_NAME_ERROR An MQOPEN or MQPUT1 call was issued to open a distribution list (that is, the *RecsPresent* field in MQOD is greater than zero), but the *ObjectQMgrName* field is neither blank nor the null string. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: If it is intended to open a distribution list, set the *ObjectQMgrName* field to blanks or the null string. If it is not intended to open a distribution list, set the *RecsPresent* field to zero.
2154	X'086A'	MQRC_RECS_PRESENT_ERROR An MQOPEN or MQPUT1 call was issued, but the call failed for one of the following reasons: • *RecsPresent* in MQOD is less than zero. • *ObjectType* in MQOD is not MQOT_Q, and *RecsPresent* is not zero. *RecsPresent* must be zero if the object being opened is not a queue. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: If it is intended to open a distribution list, set the *ObjectType* field to MQOT_Q and *RecsPresent* to the number of destinations in the list. If it is not intended to open a distribution list, set the *RecsPresent* field to zero.
2155	X'086B'	MQRC_OBJECT_RECORDS_ERROR An MQOPEN or MQPUT1 call was issued to open a distribution list (that is, the *RecsPresent* field in MQOD is greater than zero), but the MQOR object records are not specified correctly. One of the following applies: • *ObjectRecOffset* is zero and *ObjectRecPtr* is zero or the null pointer. • *ObjectRecOffset* is not zero and *ObjectRecPtr* is not zero and not the null pointer. • *ObjectRecPtr* is not a valid pointer. • *ObjectRecPtr* or *ObjectRecOffset* points to storage that is not accessible. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Ensure that one of *ObjectRecOffset* and *ObjectRecPtr* is zero and the other nonzero. Ensure that the field used points to accessible storage.

Completion and reason codes

2156	X'086C'	MQRC_RESPONSE_RECORDS_ERROR An MQOPEN or MQPUT1 call was issued to open a distribution list (that is, the *RecsPresent* field in MQOD is greater than zero), but the MQRR response records are not specified correctly. One of the following applies: • *ResponseRecOffset* is not zero and *ResponseRecPtr* is not zero and not the null pointer. • *ResponseRecPtr* is not a valid pointer. • *ResponseRecPtr* or *ResponseRecOffset* points to storage that is not accessible. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Ensure that at least one of *ResponseRecOffset* and *ResponseRecPtr* is zero. Ensure that the field used points to accessible storage.
2157	X'086D'	MQRC_ASID_MISMATCH On any MQI call, the caller's primary ASID was found to be different from the home ASID. This reason code occurs only on OS/390. Corrective action: Correct the application (MQI calls cannot be issued in cross-memory mode). Any uncommitted changes in a unit of work should be backed out. A unit of work that is coordinated by the queue manager is backed out automatically.
2158	X'086E'	MQRC_PMO_RECORD_FLAGS_ERROR An MQPUT or MQPUT1 call was issued to put a message, but the *PutMsgRecFields* field in the MQPMO structure is not valid, for one of the following reasons: • The field contains flags that are not valid. • The message is being put to a distribution list, and put message records have been provided (that is, *RecsPresent* is greater than zero, and one of *PutMsgRecOffset* or *PutMsgRecPtr* is nonzero), but *PutMsgRecFields* has the value MQPMRF_NONE. • MQPMRF_ACCOUNTING_TOKEN is specified without either MQPMO_SET_IDENTITY_CONTEXT or MQPMO_SET_ALL_CONTEXT. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Ensure that *PutMsgRecFields* is set with the appropriate MQPMRF_* flags to indicate which fields are present in the put message records. If MQPMRF_ACCOUNTING_TOKEN is specified, ensure that either MQPMO_SET_IDENTITY_CONTEXT or MQPMO_SET_ALL_CONTEXT is also specified. Alternatively, set both *PutMsgRecOffset* and *PutMsgRecPtr* to zero.
2159	X'086F'	MQRC_PUT_MSG_RECORDS_ERROR An MQPUT or MQPUT1 call was issued to put a message to a distribution list, but the MQPMR put message records are not specified correctly. One of the following applies: • *PutMsgRecOffset* is not zero and *PutMsgRecPtr* is not zero and not the null pointer. • *PutMsgRecPtr* is not a valid pointer. • *PutMsgRecPtr* or *PutMsgRecOffset* points to storage that is not accessible. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Ensure that at least one of *PutMsgRecOffset* and *PutMsgRecPtr* is zero. Ensure that the field used points to accessible storage.

2160	X'0870'	MQRC_CONN_ID_IN_USE On an MQCONN call, the connection identifier assigned by MQSeries to the connection between a CICS or IMS allied address space and the queue manager conflicts with the connection identifier of another connected CICS or IMS system. The connection identifier assigned is as follows: • For CICS, the applid • For IMS, the IMSID parameter on the IMSCTRL (sysgen) macro, or the IMSID parameter on the execution parameter (EXEC card in IMS control region JCL) • For batch, the job name • For TSO, the user ID A conflict arises only if there are two CICS systems, two IMS systems, or one each of CICS and IMS, having the same connection identifiers. Batch and TSO connections need not have unique identifiers. This reason code occurs only on OS/390. Corrective action: Ensure that the naming conventions used in different systems that might connect to MQSeries do not conflict.

2161	X'0871'	MQRC_Q_MGR_QUIESCING
		
		An MQI call was issued, but the call failed because the queue manager is quiescing (preparing to shut down).
		
		When the queue manager is quiescing, the MQOPEN, MQPUT, MQPUT1, and MQGET calls can still complete successfully, but the application can request that they fail by specifying the appropriate option on the call:
		• MQOO_FAIL_IF_QUIESCING on MQOPEN
		• MQPMO_FAIL_IF_QUIESCING on MQPUT or MQPUT1
		• MQGMO_FAIL_IF_QUIESCING on MQGET
		
		Specifying these options enables the application to become aware that the queue manager is preparing to shut down.
		• On OS/390:
		  – For batch applications, this reason can be returned to applications running in non-MQSeries LPARs.
		  – For CICS applications, this reason can be returned when no connection was established.
		• On AS/400 for applications running in compatibility mode, this reason can be returned when no connection was established.
		
		Corrective action: The application should tidy up and end. If the application specified the MQOO_FAIL_IF_QUIESCING, MQPMO_FAIL_IF_QUIESCING, or MQGMO_FAIL_IF_QUIESCING option on the failing call, the relevant option can be removed and the call reissued. By omitting these options, the application can continue working in order to complete and commit the current unit of work, but the application should not start a new unit of work.

Completion and reason codes

2162	X'0872'	**MQRC_Q_MGR_STOPPING** An MQI call was issued, but the call failed because the queue manager is shutting down. If the call was an MQGET call with the MQGMO_WAIT option, the wait has been canceled. No more MQI calls can be issued. For MQSeries client applications, it is possible that the call did complete successfully, even though this reason code is returned with a *CompCode* of MQCC_FAILED. On OS/390, the MQRC_CONNECTION_BROKEN reason may be returned instead if, as a result of system scheduling factors, the queue manager shuts down before the call completes. Corrective action: The application should tidy up and end. If the application is in the middle of a unit of work coordinated by an external unit-of-work coordinator, the application should issue the appropriate call to back out the unit of work. Any unit of work that is coordinated by the queue manager is backed out automatically.
2163	X'0873'	**MQRC_DUPLICATE_RECOV_COORD** On an MQCONN or MQCONNX call, a recovery coordinator already exists for the connection name specified on the connection call issued by the adapter. A conflict arises only if there are two CICS systems, two IMS systems, or one each of CICS and IMS, having the same connection identifiers. Batch and TSO connections need not have unique identifiers. This reason code occurs only on OS/390. Corrective action: Ensure that the naming conventions used in different systems that might connect to MQSeries do not conflict.
2173	X'087D'	**MQRC_PMO_ERROR** On an MQPUT or MQPUT1 call, the MQPMO structure is not valid, for one of the following reasons: • The *StrucId* field is not MQPMO_STRUC_ID. • The *Version* field specifies a value that is not valid or not supported. • The parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The queue manager cannot copy the changed structure to application storage, even though the call is successful. This can occur, for example, if the pointer points to read-only storage. Corrective action: Correct the definition of the MQPMO structure. Ensure that required input fields are correctly set.
2183	X'0887'	**MQRC_API_EXIT_LOAD_ERROR** The API-crossing exit module could not be linked. If this reason is returned when the API-crossing exit is invoked *after* the call has been executed, the call itself may have executed correctly. This reason code occurs only on OS/390. Corrective action: Ensure that the correct library concatenation has been specified, and that the API-crossing exit module is executable and correctly named. Any uncommitted changes in a unit of work should be backed out. A unit of work that is coordinated by the queue manager is backed out automatically.

2184	X'0888'	MQRC_REMOTE_Q_NAME_ERROR On an MQOPEN or MQPUT1 call, one of the following occurred: • A local definition of a remote queue (or an alias to one) was specified, but the *RemoteQName* attribute in the remote queue definition is entirely blank. Note that this error occurs even if the *XmitQName* in the definition is not blank. • The *ObjectQMgrName* field in the object descriptor was not blank and not the name of the local queue manager, but the *ObjectName* field is blank. Corrective action: Alter the local definition of the remote queue and supply a valid remote queue name, or supply a nonblank *ObjectName* in the object descriptor, as appropriate.
2185	X'0889'	MQRC_INCONSISTENT_PERSISTENCE The MQPUT call was issued to put a message that has a value for the *Persistence* field in MQMD that is different from the previous message put using that queue handle. This is not permitted when the MQPMO_LOGICAL_ORDER option is specified and there is already a current message group or logical message. All messages in a group and all segments in a logical message must be persistent, or all must be nonpersistent. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Modify the application to ensure that all of the messages in the group or logical message are put with the same value for the *Persistence* field in MQMD.
2186	X'088A'	MQRC_GMO_ERROR On an MQGET call, the MQGMO structure is not valid, for one of the following reasons: • The *StrucId* field is not MQGMO_STRUC_ID. • The *Version* field specifies a value that is not valid or not supported. • The parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The queue manager cannot copy the changed structure to application storage, even though the call is successful. This can occur, for example, if the pointer points to read-only storage. Corrective action: Correct the definition of the MQGMO structure. Ensure that required input fields are correctly set.
2187	X'088B'	MQRC_CICS_BRIDGE_RESTRICTION It is not permitted to issue MQI calls from user transactions that are run in an MQSeries-CICS bridge environment where the bridge exit also issues MQI calls. The MQI call fails. If this occurs in the bridge exit, it will result in a transaction abend. If it occurs in the user transaction, this may result in a transaction abend. This reason code occurs only on OS/390. Corrective action: The transaction cannot be run using the MQSeries-CICS bridge. Refer to the appropriate CICS manual for information about restrictions in the MQSeries-CICS bridge environment.
2188	X'088C'	MQRC_STOPPED_BY_CLUSTER_EXIT An MQOPEN, MQPUT, or MQPUT1 call was issued to open or put a message on a cluster queue, but the cluster workload exit rejected the call. This reason code occurs in the following environments: AIX, HP-UX, OS/390, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Check the cluster workload exit to ensure that it has been written correctly. Determine why it rejected the call and correct the problem.

Completion and reason codes

2189	X'088D'	MQRC_CLUSTER_RESOLUTION_ERROR An MQOPEN, MQPUT, or MQPUT1 call was issued to open or put a message on a cluster queue, but the queue definition could not be resolved correctly because a response was required from the repository manager but none was available. This reason code occurs in the following environments: AIX, HP-UX, OS/390, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Check that the repository manager is operating and that the queue and channel definitions are correct.
2190	X'088E'	MQRC_CONVERTED_STRING_TOO_BIG On an MQGET call with the MQGMO_CONVERT option included in the *GetMsgOpts* parameter, a string in a fixed-length field in the message expanded during data conversion and exceeded the size of the field. When this happens, the queue manager tries discarding trailing blank characters and characters following the first null character in order to make the string fit, but in this case there were insufficient characters that could be discarded. This reason code can also occur for messages with a format name of MQFMT_IMS_VAR_STRING. When this happens, it indicates that the IMS variable string expanded such that its length exceeded the capacity of the 2-byte binary length field contained within the structure of the IMS variable string. (The queue manager never discards trailing blanks in an IMS variable string.) The message is returned unconverted, with the *CompCode* parameter of the MQGET call set to MQCC_WARNING. If the message consists of several parts, each of which is described by its own character-set and encoding fields (for example, a message with format name MQFMT_DEAD_LETTER_HEADER), some parts may be converted and other parts not converted. However, the values returned in the various character-set and encoding fields always correctly describe the relevant message data. This reason code does not occur if the string could be made to fit by discarding trailing blank characters. Corrective action: Check that the fields in the message contain the correct values, and that the character-set identifiers specified by the sender and receiver of the message are correct. If they are, the layout of the data in the message must be modified to increase the lengths of the field(s) so that there is sufficient space to allow the string(s) to expand when converted.
2191	X'088F'	MQRC_TMC_ERROR On an MQPUT or MQPUT1 call, the character trigger message structure MQTMC or MQTMC2 in the message data is not valid. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the MQTMC or MQTMC2 structure. Ensure that the fields are set correctly.

2192	X'0890'	MQRC_PAGESET_FULL An MQI call was issued to operate on a queue, but the call failed because the external storage medium is full. One of the following applies: • A page-set data set is full (nonshared queues only). • A coupling-facility structure is full (shared queues only). This reason code occurs only on OS/390. Corrective action: Check which queues contain messages and look for applications that might be filling the queues unintentionally. Be aware that the queue that has caused the page set or coupling-facility structure to become full is not necessarily the queue referenced by the MQI call that returned MQRC_PAGESET_FULL. Check that all of the usual server applications are operating correctly and processing the messages on the queues. If the applications and servers are operating correctly, increase the number of server applications to cope with the message load, or request the system programmer to increase the size of the page-set data sets.
2192	X'0890'	MQRC_STORAGE_MEDIUM_FULL An MQI call was issued to operate on a queue, but the call failed because the external storage medium is full. One of the following applies: • A page-set data set is full (nonshared queues only). • A coupling-facility structure is full (shared queues only). This reason code occurs only on OS/390. Corrective action: Check which queues contain messages and look for applications that might be filling the queues unintentionally. Be aware that the queue that has caused the page set or coupling-facility structure to become full is not necessarily the queue referenced by the MQI call that returned MQRC_STORAGE_MEDIUM_FULL. Check that all of the usual server applications are operating correctly and processing the messages on the queues. If the applications and servers are operating correctly, increase the number of server applications to cope with the message load, or request the system programmer to increase the size of the page-set data sets.

Completion and reason codes

2193	X'0891'	MQRC_PAGESET_ERROR An error was encountered with the page set while attempting to access it for a locally defined queue. This could be because the queue is on a page set that does not exist. A console message is issued that tells you the number of the page set in error. For example if the error occurred in the TEST job, and your user identifier is ABCDEFG, the message is: `CSQI041I CSQIALLC JOB TEST USER ABCDEFG HAD ERROR ACCESSING PAGE SET 27` If this reason code occurs while attempting to delete a dynamic queue with MQCLOSE, the dynamic queue has not been deleted. This reason code occurs only on OS/390. Corrective action: Check that the storage class for the queue maps to a valid page set using the DISPLAY Q(xx) STGCLASS, DISPLAY STGCLASS(xx), and DISPLAY USAGE PSID commands. If you are unable to resolve the problem, notify the system programmer who should: • Collect the following diagnostic information: – A description of the actions that led to the error – A listing of the application program being run at the time of the error – Details of the page sets defined for use by MQSeries • Attempt to re-create the problem, and take a system dump immediately after the error occurs • Contact your IBM Support Center
2194	X'0892'	MQRC_NAME_NOT_VALID_FOR_TYPE An MQOPEN call was issued to open the queue manager definition, but the *ObjectName* field in the *ObjDesc* parameter is not blank. Corrective action: Ensure that the *ObjectName* field is set to blanks.
2195	X'0893'	MQRC_UNEXPECTED_ERROR The call was rejected because an unexpected error occurred. Corrective action: Check the application's parameter list to ensure, for example, that the correct number of parameters was passed, and that data pointers and storage keys are valid. If the problem cannot be resolved, contact your system programmer. • On OS/390, check whether any information has been displayed on the console. If this error occurs on an MQCONN or MQCONNX call, check that the subsystem named is an active MQSeries subsystem. In particular, check that it is not a DB2 subsystem. If the problem cannot be resolved, rerun the application with a CSQSNAP DD card (if you have not already got a dump) and send the resulting dump to IBM. • On OS/2 and AS/400, consult the FFST record to obtain more detail about the problem. • On Compaq (DIGITAL) OpenVMS, Tandem NonStop Kernel, and UNIX systems, consult the FDC file to obtain more detail about the problem.
2196	X'0894'	MQRC_UNKNOWN_XMIT_Q On an MQOPEN or MQPUT1 call, a message is to be sent to a remote queue manager. The *ObjectName* or the *ObjectQMgrName* in the object descriptor specifies the name of a local definition of a remote queue (in the latter case queue-manager aliasing is being used), but the *XmitQName* attribute of the definition is not blank and not the name of a locally-defined queue. Corrective action: Check the values specified for *ObjectName* and *ObjectQMgrName*. If these are correct, check the queue definitions. For more information on transmission queues, see the *MQSeries Application Programming Guide*.

2197	X'0895'	MQRC_UNKNOWN_DEF_XMIT_Q
		An MQOPEN or MQPUT1 call was issued specifying a remote queue as the destination. If a local definition of the remote queue was specified, or if a queue-manager alias is being resolved, the *XmitQName* attribute in the local definition is blank.
		Because there is no queue defined with the same name as the destination queue manager, the queue manager has attempted to use the default transmission queue. However, the name defined by the *DefXmitQName* queue-manager attribute is not the name of a locally-defined queue.
		Corrective action: Correct the queue definitions, or the queue-manager attribute. See the *MQSeries Application Programming Guide* for more information.
2198	X'0896'	MQRC_DEF_XMIT_Q_TYPE_ERROR
		An MQOPEN or MQPUT1 call was issued specifying a remote queue as the destination. Either a local definition of the remote queue was specified, or a queue-manager alias was being resolved, but in either case the *XmitQName* attribute in the local definition is blank.
		Because there is no transmission queue defined with the same name as the destination queue manager, the local queue manager has attempted to use the default transmission queue. However, although there is a queue defined by the *DefXmitQName* queue-manager attribute, it is not a local queue.
		Corrective action: Do one of the following: • Specify a local transmission queue as the value of the *XmitQName* attribute in the local definition of the remote queue. • Define a local transmission queue with a name that is the same as that of the remote queue manager. • Specify a local transmission queue as the value of the *DefXmitQName* queue-manager attribute.
		See the *MQSeries Application Programming Guide* for more information.
2199	X'0897'	MQRC_DEF_XMIT_Q_USAGE_ERROR
		An MQOPEN or MQPUT1 call was issued specifying a remote queue as the destination. Either a local definition of the remote queue was specified, or a queue-manager alias was being resolved, but in either case the *XmitQName* attribute in the local definition is blank.
		Because there is no transmission queue defined with the same name as the destination queue manager, the local queue manager has attempted to use the default transmission queue. However, the queue defined by the *DefXmitQName* queue-manager attribute does not have a *Usage* attribute of MQUS_TRANSMISSION.
		Corrective action: Do one of the following: • Specify a local transmission queue as the value of the *XmitQName* attribute in the local definition of the remote queue. • Define a local transmission queue with a name that is the same as that of the remote queue manager. • Specify a different local transmission queue as the value of the *DefXmitQName* queue-manager attribute. • Change the *Usage* attribute of the *DefXmitQName* queue to MQUS_TRANSMISSION.
		See the *MQSeries Application Programming Guide* for more information.

Completion and reason codes

2201	X'0899'	MQRC_NAME_IN_USE An MQOPEN call was issued to create a dynamic queue, but a queue with the same name as the dynamic queue already exists. The existing queue is one that is logically deleted, but for which there are still one or more open handles. This reason code occurs only on OS/390. Corrective action: Either ensure that all handles for the previous dynamic queue are closed, or ensure that the name of the new queue is unique; see the description for reason code MQRC_OBJECT_ALREADY_EXISTS.
2202	X'089A'	MQRC_CONNECTION_QUIESCING This reason code is issued for CICS and IMS applications when the connection to the queue manager is in quiescing state, and an application issues one of the following calls: • MQCONN or MQCONNX on IMS (these calls do not return this reason code on CICS, as the calls cannot determine that the queue manager is shutting down) • MQOPEN, with no connection established, or with MQOO_FAIL_IF_QUIESCING included in the *Options* parameter • MQGET, with MQGMO_FAIL_IF_QUIESCING included in the *Options* field of the *GetMsgOpts* parameter • MQPUT or MQPUT1, with MQPMO_FAIL_IF_QUIESCING included in the *Options* field of the *PutMsgOpts* parameter MQRC_CONNECTION_QUIESCING is also issued by the message channel agent (MCA) when the queue manager is in quiescing state. This reason code occurs only on OS/390. Corrective action: The application should tidy up and terminate.
2203	X'089B'	MQRC_CONNECTION_STOPPING This reason code is issued for CICS and IMS applications when the connection to the queue manager is shutting down, and the application issues an MQI call. No more message-queuing calls can be issued. • For the MQCONN or MQCONNX call, MQRC_CONNECTION_STOPPING is returned only on IMS. (These calls do not return this reason code on CICS, as the calls cannot determine that the queue manager is shutting down.) • For the MQGET call, if the MQGMO_WAIT option was specified, the wait is canceled. Note that the MQRC_CONNECTION_BROKEN reason may be returned instead if, as a result of system scheduling factors, the queue manager shuts down before the call completes. MQRC_CONNECTION_STOPPING is also issued by the message channel agent (MCA) when the queue manager is shutting down. For MQSeries client applications, it is possible that the call did complete successfully, even though this reason code is returned with a *CompCode* of MQCC_FAILED. This reason code occurs on OS/390, plus MQSeries clients connected to this system. Corrective action: The application should tidy up and terminate. Any uncommitted changes in a unit of work should be backed out. A unit of work that is coordinated by the queue manager is backed out automatically.

2204	X'089C'	MQRC_ADAPTER_NOT_AVAILABLE This is issued only for CICS applications, if any call is issued and the CICS adapter (a Task Related User Exit) has been disabled, or has not been enabled. This reason code occurs only on OS/390. Corrective action: The application should tidy up and terminate. Any uncommitted changes in a unit of work should be backed out. A unit of work that is coordinated by the queue manager is backed out automatically.

2206	X'089E'	MQRC_MSG_ID_ERROR
		
		An MQGET call was issued to retrieve a message using the message identifier as a selection criterion, but the call failed because selection by message identifier is not supported on this queue.
		• On OS/390, the queue is a shared queue, but the *IndexType* queue attribute does not have an appropriate value:
		  – If selection is by message identifier alone, *IndexType* must have the value MQIT_MSG_ID.
		  – If selection is by message identifier and correlation identifier combined, *IndexType* must have the value MQIT_MSG_ID or MQIT_CORREL_ID.
		• On Tandem NonStop Kernel, a key file is required but has not been defined.
		
		Corrective action: Do one of the following:
		• Modify the application so that it does not use selection by message identifier: set the *MsgId* field to MQMI_NONE and do not specify MQMO_MATCH_MSG_ID in MQGMO.
		• On OS/390, change the *IndexType* queue attribute to MQIT_MSG_ID.
		• On Tandem NonStop Kernel, define a key file.

2207	X'089F'	MQRC_CORREL_ID_ERROR
		
		An MQGET call was issued to retrieve a message using the correlation identifier as a selection criterion, but the call failed because selection by correlation identifier is not supported on this queue.
		• On OS/390, the queue is a shared queue, but the *IndexType* queue attribute does not have an appropriate value:
		  – If selection is by correlation identifier alone, *IndexType* must have the value MQIT_CORREL_ID.
		  – If selection is by correlation identifier and message identifier combined, *IndexType* must have the value MQIT_CORREL_ID or MQIT_MSG_ID.
		• On Tandem NonStop Kernel, a key file is required but has not been defined.
		
		Corrective action: Do one of the following:
		• On OS/390, change the *IndexType* queue attribute to MQIT_CORREL_ID.
		• On Tandem NonStop Kernel, define a key file.
		• Modify the application so that it does not use selection by correlation identifier: set the *CorrelId* field to MQCI_NONE and do not specify MQMO_MATCH_CORREL_ID in MQGMO.

2208	X'08A0'	MQRC_FILE_SYSTEM_ERROR An unexpected return code was received from the file system, in attempting to perform an operation on a queue. This reason code occurs only on VSE/ESA. Corrective action: Check the file system definition for the queue that was being accessed. For a VSAM file, check that the control interval is large enough for the maximum message length allowed for the queue.

Completion and reason codes

2209	X'08A1'	MQRC_NO_MSG_LOCKED An MQGET call was issued with the MQGMO_UNLOCK option, but no message was currently locked. On OS/390, this reason code does not occur. Corrective action: Check that a message was locked by an earlier MQGET call with the MQGMO_LOCK option for the same handle, and that no intervening call has caused the message to become unlocked.
2217	X'08A9'	MQRC_CONNECTION_NOT_AUTHORIZED This reason code arises only for CICS applications. For these, connection to the queue manager is done by the adapter. If that connection fails because the CICS subsystem is not authorized to connect to the queue manager, this reason code is issued whenever an application running under that subsystem subsequently issues an MQI call. This reason code occurs only on OS/390. Corrective action: Ensure that the subsystem is authorized to connect to the queue manager.
2218	X'08AA'	MQRC_MSG_TOO_BIG_FOR_CHANNEL A message was put to a remote queue, but the message is larger than the maximum message length allowed by the channel. This reason code is returned in the *Feedback* field in the message descriptor of a report message. On OS/390, this return code is issued only if you are not using CICS for distributed queuing. Otherwise, MQRC_MSG_TOO_BIG_FOR_Q_MGR is issued. Corrective action: Check the channel definitions. Increase the maximum message length that the channel can accept, or break the message into several smaller messages.
2219	X'08AB'	MQRC_CALL_IN_PROGRESS The application issued an MQI call whilst another MQI call was already being processed for that connection. Only one call per application connection can be processed at a time. Concurrent calls can arise only in certain specialized situations, such as in an exit invoked as part of the processing of an MQI call. For example, the data-conversion exit may be invoked as part of the processing of the MQGET call. • On OS/390, concurrent calls can arise only with batch or IMS applications; an example is when a subtask ends while an MQI call is in progress (for example, an MQGET that is waiting), and there is an end-of-task exit routine that issues another MQI call. • On OS/2, Windows client, and Windows NT, concurrent calls can also arise if an MQI call is issued in response to a user message while another MQI call is in progress. Corrective action: Ensure that an MQI call cannot be issued while another one is active. Do not issue MQI calls from within a data-conversion exit. On OS/390, if you want to provide a subtask to allow an application that is waiting for a message to arrive to be canceled, wait for the message by using MQGET with MQGMO_SET_SIGNAL, rather than MQGMO_WAIT.
2220	X'08AC'	MQRC_RMH_ERROR On an MQPUT or MQPUT1 call, the reference message header structure MQRMH in the message data is not valid. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the MQRMH structure. Ensure that the fields are set correctly.

2222	X'08AE'	MQRC_Q_MGR_ACTIVE
		This condition is detected when a queue manager becomes active.
		On OS/390, this event is not generated for the first start of a queue manager, only on subsequent restarts.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2223	X'08AF'	MQRC_Q_MGR_NOT_ACTIVE
		This condition is detected when a queue manager is requested to stop or quiesce.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2224	X'08B0'	MQRC_Q_DEPTH_HIGH
		An MQPUT or MQPUT1 call has caused the queue depth to be incremented to or above the limit specified in the *QDepthHighLimit* attribute.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2225	X'08B1'	MQRC_Q_DEPTH_LOW
		An MQGET call has caused the queue depth to be decremented to or below the limit specified in the *QDepthLowLimit* attribute.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2226	X'08B2'	MQRC_Q_SERVICE_INTERVAL_HIGH
		No successful gets or puts have been detected within an interval that is greater than the limit specified in the *QServiceInterval* attribute.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2227	X'08B3'	MQRC_Q_SERVICE_INTERVAL_OK
		A successful get has been detected within an interval that is less than or equal to the limit specified in the *QServiceInterval* attribute.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2232	X'08B8'	MQRC_UNIT_OF_WORK_NOT_STARTED
		An MQGET, MQPUT or MQPUT1 call was issued to get or put a message within a unit of work, but no TM/MP transaction had been started. If MQGMO_NO_SYNCPOINT is not specified on MQGET, or MQPMO_NO_SYNCPOINT is not specified on MQPUT or MQPUT1 (the default), the call requires a unit of work.
		Corrective action: Ensure a TM/MP transaction is available, or issue the MQGET call with the MQGMO_NO_SYNCPOINT option, or the MQPUT or MQPUT1 call with the MQPMO_NO_SYNCPOINT option, which will cause a transaction to be started automatically.

Completion and reason codes

2233	X'08B9'	MQRC_CHANNEL_AUTO_DEF_OK This condition is detected when the automatic definition of a channel is successful. The channel is defined by the MCA. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: None. This reason code is only used to identify the corresponding event message.
2234	X'08BA'	MQRC_CHANNEL_AUTO_DEF_ERROR This condition is detected when the automatic definition of a channel fails; this may be because an error occurred during the definition process, or because the channel automatic-definition exit inhibited the definition. Additional information is returned in the event message indicating the reason for the failure. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Examine the additional information returned in the event message to determine the reason for the failure.
2235	X'08BB'	MQRC_CFH_ERROR On an MQPUT or MQPUT1 call, the PCF header structure MQCFH in the message data is not valid. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the MQCFH structure. Ensure that the fields are set correctly.
2236	X'08BC'	MQRC_CFIL_ERROR On an MQPUT or MQPUT1 call, a PCF integer list parameter structure MQCFIL in the message data is not valid. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the MQCFIL structure. Ensure that the fields are set correctly.
2237	X'08BD'	MQRC_CFIN_ERROR On an MQPUT or MQPUT1 call, a PCF integer parameter structure MQCFIN in the message data is not valid. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the MQCFIN structure. Ensure that the fields are set correctly.

2238	X'08BE'	MQRC_CFSL_ERROR On an MQPUT or MQPUT1 call, a PCF string list parameter structure MQCFSL in the message data is not valid. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the MQCFSL structure. Ensure that the fields are set correctly.
2239	X'08BF'	MQRC_CFST_ERROR On an MQPUT or MQPUT1 call, a PCF string parameter structure MQCFST in the message data is not valid. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the MQCFST structure. Ensure that the fields are set correctly.
2241	X'08C1'	MQRC_INCOMPLETE_GROUP An operation was attempted on a queue using a queue handle that had an incomplete message group. This reason code can arise in the following situations: • On the MQPUT call, when the application attempts to put a message that is not in a group and specifies MQPMO_LOGICAL_ORDER. The call fails in this case. • On the MQPUT call, when the application attempts to put a message that is not the next one in the group and does *not* specify MQPMO_LOGICAL_ORDER, but the previous MQPUT call for the queue handle did specify MQPMO_LOGICAL_ORDER. The call succeeds with completion code MQCC_WARNING in this case. • On the MQGET call, when the application attempts to get a message that is not the next one in the group and does *not* specify MQGMO_LOGICAL_ORDER, but the previous MQGET call for the queue handle did specify MQGMO_LOGICAL_ORDER. The call succeeds with completion code MQCC_WARNING in this case. • On the MQCLOSE call, when the application attempts to close the queue that has the incomplete message group. The call succeeds with completion code MQCC_WARNING. If there is an incomplete logical message as well as an incomplete message group, reason code MQRC_INCOMPLETE_MSG is returned in preference to MQRC_INCOMPLETE_GROUP. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: If this reason code is expected, no corrective action is required. Otherwise, ensure that the MQPUT call for the last message in the group specifies MQMF_LAST_MSG_IN_GROUP.

Completion and reason codes

2242	X'08C2'	MQRC_INCOMPLETE_MSG An operation was attempted on a queue using a queue handle that had an incomplete logical message. This reason code can arise in the following situations: • On the MQPUT call, when the application attempts to put a message that is not a segment and specifies MQPMO_LOGICAL_ORDER. The call fails in this case. • On the MQPUT call, when the application attempts to put a message that is not the next segment and does *not* specify MQPMO_LOGICAL_ORDER, but the previous MQPUT call for the queue handle did specify MQPMO_LOGICAL_ORDER. The call succeeds with completion code MQCC_WARNING in this case. • On the MQGET call, when the application attempts to get a message that is not the next segment and does *not* specify MQGMO_LOGICAL_ORDER, but the previous MQGET call for the queue handle did specify MQGMO_LOGICAL_ORDER. The call succeeds with completion code MQCC_WARNING in this case. • On the MQCLOSE call, when the application attempts to close the queue that has the incomplete logical message. The call succeeds with completion code MQCC_WARNING. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: If this reason code is expected, no corrective action is required. Otherwise, ensure that the MQPUT call for the last segment specifies MQMF_LAST_SEGMENT.
2243	X'08C3'	MQRC_INCONSISTENT_CCSIDS An MQGET call was issued specifying the MQGMO_COMPLETE_MSG option, but the message to be retrieved consists of two or more segments that have differing values for the *CodedCharSetId* field in MQMD. This can arise when the segments take different paths through the network, and some of those paths have MCA sender conversion enabled. The call succeeds with a completion code of MQCC_WARNING, but only the first few segments that have identical character-set identifiers are returned. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Remove the MQGMO_COMPLETE_MSG option from the MQGET call and retrieve the remaining message segments one by one.
2244	X'08C4'	MQRC_INCONSISTENT_ENCODINGS An MQGET call was issued specifying the MQGMO_COMPLETE_MSG option, but the message to be retrieved consists of two or more segments that have differing values for the *Encoding* field in MQMD. This can arise when the segments take different paths through the network, and some of those paths have MCA sender conversion enabled. The call succeeds with a completion code of MQCC_WARNING, but only the first few segments that have identical encodings are returned. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Remove the MQGMO_COMPLETE_MSG option from the MQGET call and retrieve the remaining message segments one by one.

2245	X'08C5'	MQRC_INCONSISTENT_UOW
		One of the following applies:
		• An MQPUT call was issued to put a message in a group or a segment of a logical message, but the value specified or defaulted for the MQPMO_SYNCPOINT option is not consistent with the current group and segment information retained by the queue manager for the queue handle.
		If the current call specifies MQPMO_LOGICAL_ORDER, the call fails. If the current call does not specify MQPMO_LOGICAL_ORDER, but the previous MQPUT call for the queue handle did, the call succeeds with completion code MQCC_WARNING.
		• An MQGET call was issued to remove from the queue a message in a group or a segment of a logical message, but the value specified or defaulted for the MQGMO_SYNCPOINT option is not consistent with the current group and segment information retained by the queue manager for the queue handle.
		If the current call specifies MQGMO_LOGICAL_ORDER, the call fails. If the current call does not specify MQGMO_LOGICAL_ORDER, but the previous MQGET call for the queue handle did, the call succeeds with completion code MQCC_WARNING.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Modify the application to ensure that the same unit-of-work specification is used for all messages in the group, or all segments of the logical message.
2246	X'08C6'	MQRC_INVALID_MSG_UNDER_CURSOR
		An MQGET call was issued specifying the MQGMO_COMPLETE_MSG option with either MQGMO_MSG_UNDER_CURSOR or MQGMO_BROWSE_MSG_UNDER_CURSOR, but the message that is under the cursor has an MQMD with an *Offset* field that is greater than zero. Because MQGMO_COMPLETE_MSG was specified, the message is not valid for retrieval.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Reposition the browse cursor so that it is located on a message whose *Offset* field in MQMD is zero. Alternatively, remove the MQGMO_COMPLETE_MSG option.
2247	X'08C7'	MQRC_MATCH_OPTIONS_ERROR
		An MQGET call was issued, but the value of the *MatchOptions* field in the *GetMsgOpts* parameter is not valid. Either an undefined option is specified, or a defined option that is not valid in the current circumstances is specified. In the latter case, it means that all of the following are true:
		• MQGMO_LOGICAL_ORDER is specified.
		• There is a current message group or logical message for the queue handle.
		• Neither of the following options is specified: MQGMO_BROWSE_MSG_UNDER_CURSOR MQGMO_MSG_UNDER_CURSOR
		• One or more of the MQMO_* options is specified.
		• The values of the fields in the *MsgDesc* parameter corresponding to the MQMO_* options specified, differ from the values of those fields in the MQMD for the message to be returned next.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Ensure that only valid options are specified for the field.

Completion and reason codes

2248	X'08C8'	MQRC_MDE_ERROR
		The MQMDE structure at the start of the application message data is not valid, for one of the following reasons: • The *StrucId* field is not MQMDE_STRUC_ID. • The *Version* field is less than MQMDE_VERSION_2. • The *StrucLength* field is less than MQMDE_LENGTH_2, or (for *Version* equal to MQMDE_VERSION_2 only) greater than MQMDE_LENGTH_2. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the message descriptor extension. Ensure that required input fields are correctly set.
2249	X'08C9'	MQRC_MSG_FLAGS_ERROR
		An MQPUT or MQPUT1 call was issued, but the *MsgFlags* field in the message descriptor MQMD contains one or more message flags that are not recognized by the local queue manager. The message flags that cause this reason code to be returned depend on the destination of the message. This reason code can also occur in the *Feedback* field in the MQMD of a report message, or in the *Reason* field in the MQDLH structure of a message on the dead-letter queue; in both cases it indicates that the destination queue manager does not support one or more of the message flags specified by the sender of the message. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Do the following: • Ensure that the *MsgFlags* field in the message descriptor is initialized with a value when the message descriptor is declared, or is assigned a value prior to the MQPUT or MQPUT1 call. Specify MQMF_NONE if no message flags are needed. • Ensure that the message flags specified are ones that are documented in the *MQSeries Application Programming Reference*. Remove any message flags that are not documented in this book. • If multiple message flags are being set by adding the individual message flags together, ensure that the same message flag is not added twice.
2250	X'08CA'	MQRC_MSG_SEQ_NUMBER_ERROR
		An MQGET, MQPUT, or MQPUT1 call was issued, but the value of the *MsgSeqNumber* field in the MQMD or MQMDE structure is less than one or greater than 999 999 999. This error can also occur on the MQPUT call if the *MsgSeqNumber* field would have become greater than 999 999 999 as a result of the call. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Specify a value in the range 1 through 999 999 999. Do not attempt to create a message group containing more than 999 999 999 messages.

2251	X'08CB'	MQRC_OFFSET_ERROR An MQPUT or MQPUT1 call was issued, but the value of the *Offset* field in the MQMD or MQMDE structure is less than zero or greater than 999 999 999. This error can also occur on the MQPUT call if the *Offset* field would have become greater than 999 999 999 as a result of the call. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Specify a value in the range 0 through 999 999 999. Do not attempt to create a message segment that would extend beyond an offset of 999 999 999.
2252	X'08CC'	MQRC_ORIGINAL_LENGTH_ERROR An MQPUT or MQPUT1 call was issued to put a report message that is a segment, but the *OriginalLength* field in the MQMD or MQMDE structure is either: • Less than one (for a segment that is not the last segment), or • Less than zero (for a segment that is the last segment) This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Specify a value that is greater than zero. Zero is valid only for the last segment.
2253	X'08CD'	MQRC_SEGMENT_LENGTH_ZERO An MQPUT or MQPUT1 call was issued to put the first or an intermediate segment of a logical message, but the length of the application message data in the segment (excluding any MQSeries headers that may be present) is zero. The length must be at least one for the first or intermediate segment. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Check the application logic to ensure that segments are put with a length of one or greater. Only the last segment of a logical message is permitted to have a length of zero.
2255	X'08CF'	MQRC_UOW_NOT_AVAILABLE An MQGET, MQPUT, or MQPUT1 call was issued to get or put a message outside a unit of work, but the options specified on the call required the queue manager to process the call within a unit of work. Because there is already a user-defined unit of work in existence, the queue manager was unable to create a temporary unit of work for the duration of the call. This reason occurs in the following circumstances: • On an MQGET call, when the MQGMO_COMPLETE_MSG option is specified in MQGMO and the logical message to be retrieved is persistent and consists of two or more segments. • On an MQPUT or MQPUT1 call, when the MQMF_SEGMENTATION_ALLOWED flag is specified in MQMD and the message requires segmentation. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Issue the MQGET, MQPUT, or MQPUT1 call inside the user-defined unit of work. Alternatively, for the MQPUT or MQPUT1 call, reduce the size of the message so that it does not require segmentation by the queue manager.

Completion and reason codes

2256	X'08D0'	MQRC_WRONG_GMO_VERSION An MQGET call was issued specifying options that required an MQGMO with a version number not less than MQGMO_VERSION_2, but the MQGMO supplied did not satisfy this condition. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Modify the application to pass a version-2 MQGMO. Check the application logic to ensure that the *Version* field in MQGMO has been set to MQGMO_VERSION_2. Alternatively, remove the option that requires the version-2 MQGMO.
2257	X'08D1'	MQRC_WRONG_MD_VERSION An MQGET, MQPUT, or MQPUT1 call was issued specifying options that required an MQMD with a version number not less than MQMD_VERSION_2, but the MQMD supplied did not satisfy this condition. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Modify the application to pass a version-2 MQMD. Check the application logic to ensure that the *Version* field in MQMD has been set to MQMD_VERSION_2. Alternatively, remove the option that requires the version-2 MQMD.

2258	X'08D2'	MQRC_GROUP_ID_ERROR
		
		An MQPUT or MQPUT1 call was issued to put a distribution-list message that is also a message in a group, a message segment, or has segmentation allowed, but an invalid combination of options and values was specified. All of the following are true:
		
		• MQPMO_LOGICAL_ORDER is not specified in the *Options* field in MQPMO.
		• Either there are too few MQPMR records provided by MQPMO, or the *GroupId* field is not present in the MQPMR records.
		• One or more of the following flags is specified in the *MsgFlags* field in MQMD or MQMDE:
		    MQMF_SEGMENTATION_ALLOWED
		    MQMF_*_MSG_IN_GROUP
		    MQMF_*_SEGMENT
		• The *GroupId* field in MQMD or MQMDE is not MQGI_NONE.
		
		This combination of options and values would result in the same group identifier being used for all of the destinations in the distribution list; this is not permitted by the queue manager.
		
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		
		Corrective action: Specify MQGI_NONE for the *GroupId* field in MQMD or MQMDE. Alternatively, if the call is MQPUT specify MQPMO_LOGICAL_ORDER in the *Options* field in MQPMO.

2259	X'08D3'	MQRC_INCONSISTENT_BROWSE
		An MQGET call was issued with the MQGMO_BROWSE_NEXT option specified, but the specification of the MQGMO_LOGICAL_ORDER option for the call is different from the specification of that option for the previous call for the queue handle. Either both calls must specify MQGMO_LOGICAL_ORDER, or neither call must specify MQGMO_LOGICAL_ORDER.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Add or remove the MQGMO_LOGICAL_ORDER option as appropriate. Alternatively, to switch between logical order and physical order, specify the MQGMO_BROWSE_FIRST option to restart the scan from the beginning of the queue, omitting or specifying MQGMO_LOGICAL_ORDER as required.
2260	X'08D4'	MQRC_XQH_ERROR
		On an MQPUT or MQPUT1 call, the transmission queue header structure MQXQH in the message data is not valid.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Correct the definition of the MQXQH structure. Ensure that the fields are set correctly.
2261	X'08D5'	MQRC_SRC_ENV_ERROR
		This reason occurs when a channel exit that processes reference messages detects an error in the source environment data of a reference message header (MQRMH). One of the following is true: • *SrcEnvLength* is less than zero. • *SrcEnvLength* is greater than zero, but there is no source environment data. • *SrcEnvLength* is greater than zero, but *SrcEnvOffset* is negative, zero, or less than the length of the fixed part of MQRMH. • *SrcEnvLength* is greater than zero, but *SrcEnvOffset* plus *SrcEnvLength* is greater than *StrucLength*.
		The exit returns this reason in the *Feedback* field of the MQCXP structure. If an exception report is requested, it is copied to the *Feedback* field of the MQMD associated with the report.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Specify the source environment data correctly.
2262	X'08D6'	MQRC_SRC_NAME_ERROR
		This reason occurs when a channel exit that processes reference messages detects an error in the source name data of a reference message header (MQRMH). One of the following is true: • *SrcNameLength* is less than zero. • *SrcNameLength* is greater than zero, but there is no source name data. • *SrcNameLength* is greater than zero, but *SrcNameOffset* is negative, zero, or less than the length of the fixed part of MQRMH. • *SrcNameLength* is greater than zero, but *SrcNameOffset* plus *SrcNameLength* is greater than *StrucLength*.
		The exit returns this reason in the *Feedback* field of the MQCXP structure. If an exception report is requested, it is copied to the *Feedback* field of the MQMD associated with the report.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Specify the source name data correctly.

Completion and reason codes

2263	X'08D7'	MQRC_DEST_ENV_ERROR This reason occurs when a channel exit that processes reference messages detects an error in the destination environment data of a reference message header (MQRMH). One of the following is true: • *DestEnvLength* is less than zero. • *DestEnvLength* is greater than zero, but there is no destination environment data. • *DestEnvLength* is greater than zero, but *DestEnvOffset* is negative, zero, or less than the length of the fixed part of MQRMH. • *DestEnvLength* is greater than zero, but *DestEnvOffset* plus *DestEnvLength* is greater than *StrucLength*. The exit returns this reason in the *Feedback* field of the MQCXP structure. If an exception report is requested, it is copied to the *Feedback* field of the MQMD associated with the report. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Specify the destination environment data correctly.
2264	X'08D8'	MQRC_DEST_NAME_ERROR This reason occurs when a channel exit that processes reference messages detects an error in the destination name data of a reference message header (MQRMH). One of the following is true: • *DestNameLength* is less than zero. • *DestNameLength* is greater than zero, but there is no destination name data. • *DestNameLength* is greater than zero, but *DestNameOffset* is negative, zero, or less than the length of the fixed part of MQRMH. • *DestNameLength* is greater than zero, but *DestNameOffset* plus *DestNameLength* is greater than *StrucLength*. The exit returns this reason in the *Feedback* field of the MQCXP structure. If an exception report is requested, it is copied to the *Feedback* field of the MQMD associated with the report. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Specify the destination name data correctly.
2265	X'08D9'	MQRC_TM_ERROR On an MQPUT or MQPUT1 call, the trigger message structure MQTM in the message data is not valid. This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Correct the definition of the MQTM structure. Ensure that the fields are set correctly.

2266	X'08DA'	MQRC_CLUSTER_EXIT_ERROR
		An MQOPEN, MQPUT, or MQPUT1 call was issued to open or put a message on a cluster queue, but the cluster workload exit defined by the queue-manager's *ClusterWorkloadExit* attribute failed unexpectedly or did not respond in time. Subsequent MQOPEN, MQPUT, and MQPUT1 calls for this queue handle are processed as though the *ClusterWorkloadExit* attribute were blank.
		On OS/390, a message giving more information about the error is written to the system log, for example message CSQV455E or CSQV456E.
		This reason code occurs in the following environments: AIX, HP-UX, OS/390, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Check the cluster workload exit to ensure that it has been written correctly.
2267	X'08DB'	MQRC_CLUSTER_EXIT_LOAD_ERROR
		An MQCONN or MQCONNX call was issued to connect to a queue manager, but the call failed because the cluster workload exit defined by the queue-manager's *ClusterWorkloadExit* attribute could not be loaded.
		On OS/390, if the cluster workload exit cannot be loaded, a message is written to the system log, for example message CSQV453I. Processing continues as though the *ClusterWorkloadExit* attribute had been blank.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Ensure that the cluster workload exit has been installed in the correct location.
2268	X'08DC'	MQRC_CLUSTER_PUT_INHIBITED
		An MQOPEN call with the MQOO_OUTPUT and MQOO_BIND_ON_OPEN options in effect was issued for a cluster queue, but the call failed because all of the following are true: • All instances of the cluster queue are currently put-inhibited (that is, all of the queue instances have the *InhibitPut* attribute set to MQQA_PUT_INHIBITED). • There is no local instance of the queue. (If there is a local instance, the MQOPEN call succeeds, even if the local instance is put-inhibited.) • There is no cluster workload exit for the queue, or there is a cluster workload exit but it did not choose a queue instance. (If the cluster workload exit does choose a queue instance, the MQOPEN call succeeds, even if that instance is put-inhibited.)
		If the MQOO_BIND_NOT_FIXED option is specified on the MQOPEN call, the call can succeed even if all of the queues in the cluster are put-inhibited. However, a subsequent MQPUT call may fail if all of the queues are still put-inhibited at the time of the MQPUT call.
		This reason code occurs in the following environments: AIX, HP-UX, OS/390, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: If the system design allows put requests to be inhibited for short periods, retry the operation later. If the problem persists, determine why all of the queues in the cluster are put-inhibited.

Completion and reason codes

2269	X'08DD'	MQRC_CLUSTER_RESOURCE_ERROR An MQOPEN, MQPUT, or MQPUT1 call was issued for a cluster queue, but an error occurred whilst trying to use a resource required for clustering. This reason code occurs in the following environments: AIX, HP-UX, OS/390, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Do the following: • Check that the SYSTEM.CLUSTER.* queues are not put inhibited or full. • Check the event queues for any events relating to the SYSTEM.CLUSTER.* queues, as these may give guidance as to the nature of the failure. • Check that the repository queue manager is available. • On OS/390, check the console for signs of the failure, such as full page sets.
2270	X'08DE'	MQRC_NO_DESTINATIONS_AVAILABLE An MQPUT or MQPUT1 call was issued to put a message on a cluster queue, but at the time of the call there were no longer any instances of the queue in the cluster. The message therefore could not be sent. This situation can occur when MQOO_BIND_NOT_FIXED is specified on the MQOPEN call that opens the queue, or MQPUT1 is used to put the message. This reason code occurs in the following environments: AIX, HP-UX, OS/390, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems. Corrective action: Check the queue definition and queue status to determine why all instances of the queue were removed from the cluster. Correct the problem and rerun the application.
2271	X'08DF'	MQRC_CONN_TAG_IN_USE An MQCONNX call was issued specifying one of the MQCNO_*_CONN_TAG_* options, but the call failed because the connection tag specified by *ConnTag* in MQCNO is in use by an active process or thread, or there is an unresolved unit of work that references this connection tag. This reason code occurs only on OS/390. Corrective action: The problem is likely to be transitory. The application should wait a short while and then retry the operation.
2273	X'08E1'	MQRC_CONNECTION_ERROR An MQCONN or MQCONNX call failed for one of the following reasons: • The system parameter module is not at the same release level as the queue manager. • An internal error was detected by the queue manager. This reason code occurs only on OS/390. Corrective action: Relinkedit the system parameter module (CSQZPARM) to ensure that it is at the correct level. If the problem persists, contact your IBM support center.
2274	X'08E2'	MQRC_OPTION_ENVIRONMENT_ERROR An MQGET call with the MQGMO_MARK_SKIP_BACKOUT option specified was issued from a DB2 Stored Procedure. The call failed because the MQGMO_MARK_SKIP_BACKOUT option cannot be used from a DB2 Stored Procedure. This reason code occurs only on OS/390. Corrective action: Remove the MQGMO_MARK_SKIP_BACKOUT option from the MQGET call.

2277	X'08E5'	MQRC_CD_ERROR
		An MQCONNX call was issued to connect to a queue manager, but the MQCD channel definition structure addressed by the *ClientConnOffset* or *ClientConnPtr* field in MQCNO contains data that is not valid. Consult the MQSeries error log for more information about the nature of the error.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Ensure that required input fields in the MQCD structure are set correctly.
2278	X'08E6'	MQRC_CLIENT_CONN_ERROR
		An MQCONNX call was issued to connect to a queue manager, but the MQCD channel definition structure is not specified correctly. One of the following applies: • *ClientConnOffset* is not zero and *ClientConnPtr* is not zero and not the null pointer. • *ClientConnPtr* is not a valid pointer. • *ClientConnPtr* or *ClientConnOffset* points to storage that is not accessible.
		This reason code occurs in the following environments: AIX, HP-UX, OS/2, AS/400, Sun Solaris, Windows NT, plus MQSeries clients connected to these systems.
		Corrective action: Ensure that at least one of *ClientConnOffset* and *ClientConnPtr* is zero. Ensure that the field used points to accessible storage.
2279	X'08E7'	MQRC_CHANNEL_STOPPED_BY_USER
		This condition is detected when the channel has been stopped by an operator. The reason qualifier identifies the reasons for stopping.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2280	X'08E8'	MQRC_HCONFIG_ERROR
		The configuration handle *Hconfig* specified on the MQZEP call issued by an installable service is not valid.
		On OS/390, this reason code does not occur.
		Corrective action: Specify the configuration handle that was provided to the installable service's configuration function on the component initialization call. See the *MQSeries Programmable System Management* book for information about installable services.
2281	X'08E9'	MQRC_FUNCTION_ERROR
		The function identifier *Function* specified on the MQZEP call issued by an installable service is not valid for the service being configured.
		On OS/390, this reason code does not occur.
		Corrective action: Specify an MQZID_* value that is valid for the installable service being configured. Refer to the description of the MQZEP call in the *MQSeries Programmable System Management* book to determine which values are valid.
2282	X'08EA'	MQRC_CHANNEL_STARTED
		One of the following has occurred: • An operator has issued a Start Channel command. • An instance of a channel has been successfully established. This condition is detected when Initial Data negotiation is complete and resynchronization has been performed where necessary such that message transfer can proceed.
		Corrective action: None. This reason code is only used to identify the corresponding event message.

Completion and reason codes

2283	X'08EB'	MQRC_CHANNEL_STOPPED
		This condition is detected when the channel has been stopped. The reason qualifier identifies the reasons for stopping.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2284	X'08EC'	MQRC_CHANNEL_CONV_ERROR
		This condition is detected when a channel is unable to do data conversion and the MQGET call to get a message from the transmission queue resulted in a data conversion error. The conversion reason code identifies the reason for the failure.
		Corrective action: None. This reason code is only used to identify the corresponding event message.
2285	X'08ED'	MQRC_SERVICE_NOT_AVAILABLE
		This reason should be returned by an installable service component when the requested action cannot be performed because the required underlying service is not available.
		On OS/390, this reason code does not occur.
		Corrective action: Make the underlying service available.
2286	X'08EE'	MQRC_INITIALIZATION_FAILED
		This reason should be returned by an installable service component when the component is unable to complete initialization successfully.
		On OS/390, this reason code does not occur.
		Corrective action: Correct the error and retry the operation.
2287	X'08EF'	MQRC_TERMINATION_FAILED
		This reason should be returned by an installable service component when the component is unable to complete termination successfully.
		On OS/390, this reason code does not occur.
		Corrective action: Correct the error and retry the operation.
2288	X'08F0'	MQRC_UNKNOWN_Q_NAME
		This reason should be returned by the MQZ_LOOKUP_NAME installable service component when the name specified for the *QName* parameter is not recognized.
		On OS/390, this reason code does not occur.
		Corrective action: None. See the *MQSeries Programmable System Management* book for information about installable services.
2289	X'08F1'	MQRC_SERVICE_ERROR
		This reason should be returned by an installable service component when the component encounters an unexpected error.
		On OS/390, this reason code does not occur.
		Corrective action: Correct the error and retry the operation.

2290	X'08F2'	MQRC_Q_ALREADY_EXISTS
		This reason should be returned by the MQZ_INSERT_NAME installable service component when the queue specified by the *QName* parameter is already defined to the name service.
		On OS/390, this reason code does not occur.
		Corrective action: None. See the *MQSeries Programmable System Management* book for information about installable service.
2291	X'08F3'	MQRC_USER_ID_NOT_AVAILABLE
		This reason should be returned by the MQZ_FIND_USERID installable service component when the user ID cannot be determined.
		On OS/390, this reason code does not occur.
		Corrective action: None. See the *MQSeries Programmable System Management* book for information about installable services.
2292	X'08F4'	MQRC_UNKNOWN_ENTITY
		This reason should be returned by the authority installable service component when the name specified by the *EntityName* parameter is not recognized.
		On OS/390, this reason code does not occur.
		Corrective action: Ensure that the entity is defined.
2293	X'08F5'	MQRC_UNKNOWN_AUTH_ENTITY
		This reason should be returned by the authority installable service component when the name specified by the *AuthEntityName* parameter is not recognized.
		On OS/390, this reason code does not occur.
		Corrective action: Ensure that the entity is defined.
2294	X'08F6'	MQRC_UNKNOWN_REF_OBJECT
		This reason should be returned by the MQZ_COPY_ALL_AUTHORITY installable service component when the name specified by the *RefObjectName* parameter is not recognized.
		On OS/390, this reason code does not occur.
		Corrective action: Ensure that the reference object is defined. See the *MQSeries Programmable System Management* book for information about installable services.
2295	X'08F7'	MQRC_CHANNEL_ACTIVATED
		This condition is detected when a channel that has been waiting to become active, and for which a Channel Not Activated event has been generated, is now able to become active because an active slot has been released by another channel.
		This event is not generated for a channel that is able to become active without waiting for an active slot to be released.
		Corrective action: None. This reason code is only used to identify the corresponding event message.

Completion and reason codes

2296	X'08F8'	MQRC_CHANNEL_NOT_ACTIVATED This condition is detected when a channel is required to become active, either because it is starting or because it is about to make another attempt to establish connection with its partner. However, it is unable to do so because the limit on the number of active channels has been reached. • On OS/390, the maximum number of active channels is given by the ACTCHL parameter in CSQXPARM. • In other environments, the maximum number of active channels is given by the MaxActiveChannels parameter in the qm.ini file. The channel waits until it is able to take over an active slot released when another channel ceases to be active. At that time a Channel Activated event is generated. Corrective action: None. This reason code is only used to identify the corresponding event message.
2297	X'08F9'	MQRC_UOW_CANCELED An MQI call was issued, but the unit of work (TM/MP transaction) being used for the MQSeries operation had been canceled. This may have been done by TM/MP itself (for example, due to the transaction running for too long, or exceeding audit trail sizes), or by the application program issuing an ABORT_TRANSACTION. All updates performed to MQSeries resources are backed out. Corrective action: Refer to the operating system's *Transaction Management Operations Guide* to determine how the Transaction Manager can be tuned to avoid the problem of system limits being exceeded.
2298	X'08FA'	MQRC_FUNCTION_NOT_SUPPORTED The function requested is not available in the current environment. Corrective action: Remove the call from the application.
2299	X'08FB'	MQRC_SELECTOR_TYPE_ERROR The *Selector* parameter has the wrong data type; it must be of type Long. Corrective action: Declare the *Selector* parameter as Long.
2300	X'08FC'	MQRC_COMMAND_TYPE_ERROR The mqExecute call was issued, but the value of the MQIASY_TYPE data item in the administration bag is not MQCFT_COMMAND. Corrective action: Ensure that the MQIASY_TYPE data item in the administration bag has the value MQCFT_COMMAND.
2301	X'08FD'	MQRC_MULTIPLE_INSTANCE_ERROR The *Selector* parameter specifies a system selector (one of the MQIASY_* values), but the value of the *ItemIndex* parameter is not MQIND_NONE. Only one instance of each system selector can exist in the bag. Corrective action: Specify MQIND_NONE for the *ItemIndex* parameter.
2302	X'08FE'	MQRC_SYSTEM_ITEM_NOT_ALTERABLE A call was issued to modify the value of a system data item in a bag (a data item with one of the MQIASY_* selectors), but the call failed because the data item is one that cannot be altered by the application. Corrective action: Specify the selector of a user-defined data item, or remove the call.

2303	X'08FF'	MQRC_BAG_CONVERSION_ERROR
		The mqBufferToBag or mqGetBag call was issued, but the data in the buffer or message could not be converted into a bag. This occurs when the data to be converted is not valid PCF.
		Corrective action: Check the logic of the application that created the buffer or message to ensure that the buffer or message contains valid PCF.
		If the message contains PCF that is not valid, the message cannot be retrieved using the mqGetBag call: • If one of the MQGMO_BROWSE_* options was specified, the message remains on the queue and can be retrieved using the MQGET call. • In other cases, the message has already been removed from the queue and discarded. If the message was retrieved within a unit of work, the unit of work can be backed out and the message retrieved using the MQGET call.
2304	X'0900'	MQRC_SELECTOR_OUT_OF_RANGE
		The *Selector* parameter has a value that is outside the valid range for the call. If the bag was created with the MQCBO_CHECK_SELECTORS option: • For the mqAddInteger call, the value must be within the range MQIA_FIRST through MQIA_LAST. • For the mqAddString call, the value must be within the range MQCA_FIRST through MQCA_LAST.
		If the bag was not created with the MQCBO_CHECK_SELECTORS option: The value must be zero or greater.
		Corrective action: Specify a valid value.
2305	X'0901'	MQRC_SELECTOR_NOT_UNIQUE
		The *ItemIndex* parameter has the value MQIND_NONE, but the bag contains more than one data item with the selector value specified by the *Selector* parameter. MQIND_NONE requires that the bag contain only one occurrence of the specified selector.
		This reason code also occurs on the mqExecute call when the administration bag contains two or more occurrences of a selector for a required parameter that permits only one occurrence.
		Corrective action: Check the logic of the application that created the bag. If correct, specify for *ItemIndex* a value that is zero or greater, and add application logic to process all of the occurrences of the selector in the bag.
		Review the description of the administration command being issued, and ensure that all required parameters are defined correctly in the bag.
2306	X'0902'	MQRC_INDEX_NOT_PRESENT
		The specified index is not present: • For a bag, this means that the bag contains one or more data items that have the selector value specified by the *Selector* parameter, but none of them has the index value specified by the *ItemIndex* parameter. The data item identified by the *Selector* and *ItemIndex* parameters must exist in the bag. • For a namelist, this means that the index parameter value is too large, and outside the range of valid values.
		Corrective action: Specify the index of a data item that does exist in the bag or namelist. Use the mqCountItems call to determine the number of data items with the specified selector that exist in the bag, or the nameCount method to determine the number of names in the namelist.

Completion and reason codes

2307	X'0903'	MQRC_STRING_ERROR The *String* parameter is not valid. Either the parameter pointer is not valid, or it points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Correct the parameter.
2308	X'0904'	MQRC_ENCODING_NOT_SUPPORTED The *Encoding* field in the message descriptor MQMD contains a value that is not supported: • For the mqPutBag call, the field in error resides in the *MsgDesc* parameter of the call. • For the mqGetBag call, the field in error resides in: – The *MsgDesc* parameter of the call if the MQGMO_CONVERT option was specified. – The message descriptor of the message about to be retrieved if MQGMO_CONVERT was *not* specified. Corrective action: The value must be MQENC_NATIVE. If the value of the *Encoding* field in the message is not valid, the message cannot be retrieved using the mqGetBag call: • If one of the MQGMO_BROWSE_* options was specified, the message remains on the queue and can be retrieved using the MQGET call. • In other cases, the message has already been removed from the queue and discarded. If the message was retrieved within a unit of work, the unit of work can be backed out and the message retrieved using the MQGET call.
2309	X'0905'	MQRC_SELECTOR_NOT_PRESENT The *Selector* parameter specifies a selector that does not exist in the bag. Corrective action: Specify a selector that does exist in the bag.
2310	X'0906'	MQRC_OUT_SELECTOR_ERROR The *OutSelector* parameter is not valid. Either the parameter pointer is not valid, or it points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Correct the parameter.
2311	X'0907'	MQRC_STRING_TRUNCATED The string returned by the call is too long to fit in the buffer provided. The string has been truncated to fit in the buffer. Corrective action: If the entire string is required, provide a larger buffer. On the mqInquireString call, the *StringLength* parameter is set by the call to indicate the size of the buffer required to accommodate the string without truncation.

2312	X'0908'	MQRC_SELECTOR_WRONG_TYPE
		A data item with the specified selector exists in the bag, but has a data type that conflicts with the data type implied by the call being used. For example, the data item might have an integer data type, but the call being used might be mqSetString, which implies a character data type.
		This reason code also occurs on the mqBagToBuffer, mqExecute, and mqPutBag calls when mqAddString or mqSetString was used to add the MQIACF_INQUIRY data item to the bag.
		Corrective action: For the mqSetInteger and mqSetString calls, specify MQIND_ALL for the *ItemIndex* parameter to delete from the bag all existing occurrences of the specified selector before creating the new occurrence with the required data type.
		For the mqInquireBag, mqInquireInteger, and mqInquireString calls, use the mqInquireItemInfo call to determine the data type of the item with the specified selector, and then use the appropriate call to determine the value of the data item.
		For the mqBagToBuffer, mqExecute, and mqPutBag calls, ensure that the MQIACF_INQUIRY data item is added to the bag using the mqAddInteger or mqSetInteger calls.
2313	X'0909'	MQRC_INCONSISTENT_ITEM_TYPE
		The mqAddInteger or mqAddString call was issued to add another occurrence of the specified selector to the bag, but the data type of this occurrence differed from the data type of the first occurrence.
		This reason can also occur on the mqBufferToBag and mqGetBag calls, where it indicates that the PCF in the buffer or message contains a selector that occurs more than once but with inconsistent data types.
		Corrective action: For the mqAddInteger and mqAddString calls, use the call appropriate to the data type of the first occurrence of that selector in the bag.
		For the mqBufferToBag and mqGetBag calls, check the logic of the application that created the buffer or sent the message to ensure that multiple-occurrence selectors occur with only one data type. A message that contains a mixture of data types for a selector cannot be retrieved using the mqGetBag call: • If one of the MQGMO_BROWSE_* options was specified, the message remains on the queue and can be retrieved using the MQGET call. • In other cases, the message has already been removed from the queue and discarded. If the message was retrieved within a unit of work, the unit of work can be backed out and the message retrieved using the MQGET call.
2314	X'090A'	MQRC_INDEX_ERROR
		An index parameter to a call or method has a value that is not valid. The value must be zero or greater. For bag calls, certain MQIND_* values can also be specified: • For the mqDeleteItem, mqSetInteger and mqSetString calls, MQIND_ALL and MQIND_NONE are valid. • For the mqInquireBag, mqInquireInteger, mqInquireString, and mqInquireItemInfo calls, MQIND_NONE is valid.
		Corrective action: Specify a valid value.
2315	X'090B'	MQRC_SYSTEM_BAG_NOT_ALTERABLE
		A call was issued to add a data item to a bag, modify the value of an existing data item in a bag, or retrieve a message into a bag, but the call failed because the bag is one that had been created by the system as a result of a previous mqExecute call. System bags cannot be modified by the application.
		Corrective action: Specify the handle of a bag created by the application, or remove the call.

Completion and reason codes

2316	X'090C'	MQRC_ITEM_COUNT_ERROR The mqTruncateBag call was issued, but the *ItemCount* parameter specifies a value that is not valid. The value is either less than zero, or greater than the number of user-defined data items in the bag. This reason also occurs on the mqCountItems call if the parameter pointer is not valid, or points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Specify a valid value. Use the mqCountItems call to determine the number of user-defined data items in the bag.
2317	X'090D'	MQRC_FORMAT_NOT_SUPPORTED The *Format* field in the message descriptor MQMD contains a value that is not supported: • For the mqPutBag call, the field in error resides in the *MsgDesc* parameter of the call. • For the mqGetBag call, the field in error resides in the message descriptor of the message about to be retrieved. Corrective action: The value must be one of the following: MQFMT_ADMIN MQFMT_EVENT MQFMT_PCF If the value of the *Format* field in the message is none of these values, the message cannot be retrieved using the mqGetBag call: • If one of the MQGMO_BROWSE_* options was specified, the message remains on the queue and can be retrieved using the MQGET call. • In other cases, the message has already been removed from the queue and discarded. If the message was retrieved within a unit of work, the unit of work can be backed out and the message retrieved using the MQGET call.
2318	X'090E'	MQRC_SELECTOR_NOT_SUPPORTED The *Selector* parameter specifies a value that is a system selector (a value that is negative), but the system selector is not one that is supported by the call. Corrective action: Specify a selector value that is supported.
2319	X'090F'	MQRC_ITEM_VALUE_ERROR The mqInquireBag or mqInquireInteger call was issued, but the *ItemValue* parameter is not valid. Either the parameter pointer is not valid, or it points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Correct the parameter.
2320	X'0910'	MQRC_HBAG_ERROR A call was issued that has a parameter that is a bag handle, but the handle is not valid. For output parameters, this reason also occurs if the parameter pointer is not valid, or points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) Corrective action: Correct the parameter.
2321	X'0911'	MQRC_PARAMETER_MISSING An administration message requires a parameter that is not present in the administration bag. This reason code occurs only for bags created with the MQCBO_ADMIN_BAG or MQCBO_REORDER_AS_REQUIRED options. Corrective action: Review the description of the administration command being issued, and ensure that all required parameters are present in the bag.

2322	X'0912'	MQRC_CMD_SERVER_NOT_AVAILABLE
		The command server that processes administration commands is not available.
		Corrective action: Start the command server.
2323	X'0913'	MQRC_STRING_LENGTH_ERROR
		The *StringLength* parameter is not valid. Either the parameter pointer is not valid, or it points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		Corrective action: Correct the parameter.
2324	X'0914'	MQRC_INQUIRY_COMMAND_ERROR
		The mqAddInquiry call was used previously to add attribute selectors to the bag, but the command code to be used for the mqBagToBuffer, mqExecute, or mqPutBag call is not recognized. As a result, the correct PCF message cannot be generated.
		Corrective action: Remove the mqAddInquiry calls and use instead the mqAddInteger call with the appropriate MQIACF_*_ATTRS or MQIACH_*_ATTRS selectors.
2325	X'0915'	MQRC_NESTED_BAG_NOT_SUPPORTED
		A bag that is input to the call contains nested bags. Nested bags are supported only for bags that are output from the call.
		Corrective action: Use a different bag as input to the call.
2326	X'0916'	MQRC_BAG_WRONG_TYPE
		The *Bag* parameter specifies the handle of a bag that has the wrong type for the call. The bag must be an administration bag, that is, it must be created with the MQCBO_ADMIN_BAG option specified on the mqCreateBag call.
		Corrective action: Specify the MQCBO_ADMIN_BAG option when the bag is created.
2327	X'0917'	MQRC_ITEM_TYPE_ERROR
		The mqInquireItemInfo call was issued, but the *ItemType* parameter is not valid. Either the parameter pointer is not valid, or it points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		Corrective action: Correct the parameter.
2328	X'0918'	MQRC_SYSTEM_BAG_NOT_DELETABLE
		An mqDeleteBag call was issued to delete a bag, but the call failed because the bag is one that had been created by the system as a result of a previous mqExecute call. System bags cannot be deleted by the application.
		Corrective action: Specify the handle of a bag created by the application, or remove the call.
2329	X'0919'	MQRC_SYSTEM_ITEM_NOT_DELETABLE
		A call was issued to delete a system data item from a bag (a data item with one of the MQIASY_* selectors), but the call failed because the data item is one that cannot be deleted by the application.
		Corrective action: Specify the selector of a user-defined data item, or remove the call.

Completion and reason codes

2330	X'091A'	MQRC_CODED_CHAR_SET_ID_ERROR
		The *CodedCharSetId* parameter is not valid. Either the parameter pointer is not valid, or it points to read-only storage. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.)
		Corrective action: Correct the parameter.
2331	X'091B'	MQRC_MSG_TOKEN_ERROR
		An MQGET call was issued to retrieve a message using the message token as a selection criterion, but the options specified are not valid, for one of the following reasons: • MQMO_MATCH_MSG_TOKEN was specified, but the queue is not indexed by message token (that is, the queue's *IndexType* attribute does not have the value MQIT_MSG_TOKEN). • MQMO_MATCH_MSG_TOKEN was specified with either MQGMO_WAIT or MQGMO_SET_SIGNAL.
		This reason code occurs only on OS/390.
		Corrective action: Do one of the following: • Modify the queue definition so that the queue is indexed by message token. • Remove the MQMO_MATCH_MSG_TOKEN option from the MQGET call.
2332	X'091C'	MQRC_MISSING_WIH
		An MQPUT or MQPUT1 call was issued to put a message on a queue whose *IndexType* attribute had the value MQIT_MSG_TOKEN, but the *Format* field in the MQMD was not MQFMT_WORK_INFO_HEADER. This error occurs only when the message arrives at the destination queue manager.
		This reason code occurs only on OS/390.
		Corrective action: Modify the application to ensure that it places an MQWIH structure at the start of the message data, and sets the *Format* field in the MQMD to MQFMT_WORK_INFO_HEADER.
2333	X'091D'	MQRC_WIH_ERROR
		An MQPUT or MQPUT1 call was issued to put a message on a queue whose *IndexType* attribute had the value MQIT_MSG_TOKEN, but the message data did not begin with a valid MQWIH structure.
		This reason code occurs only on OS/390.
		Corrective action: Modify the application to ensure that it places a valid MQWIH structure at the start of the message data.
2334	X'091E'	MQRC_RFH_ERROR
		The message contains an MQRFH structure, but the structure is not valid.
		Corrective action: Modify the application that generated the message to ensure that it places a valid MQRFH structure in the message data.

2335	X'091F'	MQRC_RFH_STRING_ERROR
		The contents of the *NameValueString* field in the MQRFH structure are not valid. *NameValueString* must adhere to the following rules: • The string must consist of zero or more name/value pairs separated from each other by one or more blanks; the blanks are not significant. • If a name or value contains blanks that are significant, the name or value must be enclosed in double-quote characters. • If a name or value itself contains one or more double-quote characters, the name or value must be enclosed in double-quote characters, and each embedded double-quote character must be doubled. • A name or value can contain any characters other than the null, which acts as a delimiter. The null and characters following it, up to the defined length of *NameValueString*, are ignored. The following is a valid *NameValueString*: `Famous_Words "The program displayed ""Hello World"""` Corrective action: Modify the application that generated the message to ensure that it places in the *NameValueString* field data that adheres to the rules listed above. Check that the *StrucLength* field is set to the correct value.
2336	X'0920'	MQRC_RFH_COMMAND_ERROR
		The message contains an MQRFH structure, but the command name contained in the *NameValueString* field is not valid. Corrective action: Modify the application that generated the message to ensure that it places in the *NameValueString* field a command name that is valid.
2337	X'0921'	MQRC_RFH_PARM_ERROR
		The message contains an MQRFH structure, but a parameter name contained in the *NameValueString* field is not valid for the command specified. Corrective action: Modify the application that generated the message to ensure that it places in the *NameValueString* field only parameters that are valid for the specified command.
2338	X'0922'	MQRC_RFH_DUPLICATE_PARM
		The message contains an MQRFH structure, but a parameter occurs more than once in the *NameValueString* field when only one occurrence is valid for the specified command. Corrective action: Modify the application that generated the message to ensure that it places in the *NameValueString* field only one occurrence of the parameter.
2339	X'0923'	MQRC_RFH_PARM_MISSING
		The message contains an MQRFH structure, but the command specified in the *NameValueString* field requires a parameter that is not present. Corrective action: Modify the application that generated the message to ensure that it places in the *NameValueString* field all parameters that are required for the specified command.
2340	X'0924'	MQRC_CHAR_CONVERSION_ERROR
		This reason code is returned by the Java MQQueueManager constructor when a required character-set conversion is not available. The conversion required is between two nonUnicode character sets. This reason code occurs in the following environment: MQSeries Classes for Java on OS/390. Corrective action: Ensure that the National Language Resources component of the OS/390 Language Environment® is installed, and that conversion between the IBM-1047 and ISO8859-1 character sets is available.

Completion and reason codes

2341	X'0925'	MQRC_UCS2_CONVERSION_ERROR
		This reason code is returned by the Java MQQueueManager constructor when a required character-set conversion is not available. The conversion required is between the UCS-2 Unicode character set and the queue-manager's character set. IBM-500 is used for the queue-manager's character set if no specific value is available.
		This reason code occurs in the following environment: MQSeries classes for Java on OS/390.
		Corrective action: Ensure that the relevant Unicode conversion tables are installed, and that they are available to the OS/390 Language Environment. The conversion tables should be installed as part of the OS/390 C/C++ optional feature. Refer to the *OS/390 C/C++ Programming Guide* for more information about enabling UCS-2 conversions.
2342	X'0926'	MQRC_DB2_NOT_AVAILABLE
		An MQOPEN, MQPUT1, or MQSET call was issued to access a shared queue, but the call failed because the queue manager is not connected to a DB2 subsystem. As a result, the queue manager is unable to access the object definition relating to the shared queue.
		This reason code occurs only on OS/390.
		Corrective action: Configure the DB2 subsystem so that the queue manager can connect to it.
2343	X'0927'	MQRC_OBJECT_NOT_UNIQUE
		An MQOPEN or MQPUT1 call was issued to access a queue, but the call failed because the queue specified in the MQOD structure cannot be resolved unambiguously. There exists a shared queue with the specified name, and a nonshared queue with the same name.
		This reason code occurs only on OS/390.
		Corrective action: One of the queues must be deleted. If the queue to be deleted contains messages, use the MQSC command MOVE QLOCAL to move the messages to a different queue, and then use the command DELETE QLOCAL to delete the queue.
2344	X'0928'	MQRC_CONN_TAG_NOT_RELEASED
		An MQDISC call was issued when there was a unit of work outstanding for the connection handle. For CICS, IMS, and RRS connections, the MQDISC call does not commit or back out the unit of work. As a result, the connection tag associated with the unit of work is not yet available for reuse. The tag becomes available for reuse only when processing of the unit of work has been completed.
		This reason code occurs only on OS/390.
		Corrective action: Do not try to reuse the connection tag immediately. If the MQCONNX call is issued with the same connection tag, and that tag is still in use, the call fails with reason code MQRC_CONN_TAG_IN_USE.
2345	X'0929'	MQRC_CF_NOT_AVAILABLE
		An MQOPEN or MQPUT1 call was issued to access a shared queue, but the allocation of the coupling-facility structure specified in the queue definition failed because there is no suitable coupling facility to hold the structure, based on the preference list in the active CFRM policy.
		This reason code occurs only on OS/390.
		Corrective action: Make available a coupling facility with one of the names specified in the CFRM policy, or modify the CFRM policy to specify the names of coupling facilities that are available.

2346	X'092A'	MQRC_CF_STRUC_IN_USE
		An MQI call was issued to operate on a shared queue, but the call failed because the coupling-facility structure specified in the queue definition is temporarily unavailable. The coupling-facility structure can be unavailable because a structure dump is in progress, or new connectors to the structure are currently inhibited, or an existing connector to the structure failed or disconnected abnormally and clean-up is not yet complete.
		This reason code occurs only on OS/390.
		Corrective action: The problem is temporary; wait a short while and then retry the operation.
2347	X'092B'	MQRC_CF_STRUC_LIST_HDR_IN_USE
		An MQGET, MQOPEN, MQPUT1, or MQSET call was issued to access a shared queue, but the call failed because the list header associated with the coupling-facility structure specified in the queue definition is temporarily unavailable. The list header is unavailable because it is undergoing recovery processing.
		This reason code occurs only on OS/390.
		Corrective action: The problem is temporary; wait a short while and then retry the operation.
2348	X'092C'	MQRC_CF_STRUC_AUTH_FAILED
		An MQOPEN or MQPUT1 call was issued to access a shared queue, but the call failed because the user is not authorized to access the coupling-facility structure specified in the queue definition.
		This reason code occurs only on OS/390.
		Corrective action: Modify the security profile for the user identifier used by the application so that the application can access the coupling-facility structure specified in the queue definition.
2349	X'092D'	MQRC_CF_STRUC_ERROR
		An MQOPEN or MQPUT1 call was issued to access a shared queue, but the call failed because the coupling-facility structure name specified in the queue definition is not defined in the CFRM data set, or is not the name of a list structure.
		This reason code occurs only on OS/390.
		Corrective action: Modify the queue definition to specify the name of a coupling-facility list structure that is defined in the CFRM data set.
2350	X'092E'	MQRC_CONN_TAG_NOT_USABLE
		An MQCONNX call was issued specifying one of the MQCNO_*_CONN_TAG_* options, but the call failed because the connection tag specified by *ConnTag* in MQCNO is being used by the queue manager for recovery processing, and this processing is delayed pending recovery of the coupling facility.
		This reason code occurs only on OS/390.
		Corrective action: The problem is likely to persist. Consult the system programmer to ascertain the cause of the problem.

Completion and reason codes

2351	X'092F'	MQRC_GLOBAL_UOW_CONFLICT An attempt was made to use inside a global unit of work a connection handle that is participating in another global unit of work. This can occur when an application passes connection handles between objects where the objects are involved in different DTC transactions. Because transaction completion is asynchronous, it is possible for this error to occur *after* the application has finalized the first object and committed its transaction. This error does not occur for nontransactional MQI calls. This reason code occurs only on Windows NT. Corrective action: Check that the "MTS Transaction Support" attribute defined for the object's class is set correctly. If necessary, modify the application so that the connection handle is not used by objects participating in different units of work.
2352	X'0930'	MQRC_LOCAL_UOW_CONFLICT An attempt was made to use inside a global unit of work a connection handle that is participating in a queue-manager coordinated local unit of work. This can occur when an application passes connection handles between objects where one object is involved in a DTC transaction and the other is not. This error does not occur for nontransactional MQI calls. This reason code occurs only on Windows NT. Corrective action: Check that the "MTS Transaction Support" attribute defined for the object's class is set correctly. If necessary, modify the application so that the connection handle is not used by objects participating in different units of work.
2353	X'0931'	MQRC_HANDLE_IN_USE_FOR_UOW An attempt was made to use outside a unit of work a connection handle that is participating in a global unit of work. This error can occur when an application passes connection handles between objects where one object is involved in a DTC transaction and the other is not. Because transaction completion is asynchronous, it is possible for this error to occur *after* the application has finalized the first object and committed its transaction. This error can also occur when a single object that was created and associated with the transaction loses that association whilst the object is running. The association is lost when DTC terminates the transaction independently of MTS. This might be because the transaction timed out, or because DTC shut down. This error does not occur for nontransactional MQI calls. This reason code occurs only on Windows NT. Corrective action: Check that the "MTS Transaction Support" attribute defined for the object's class is set correctly. If necessary, modify the application so that objects executing within different units of work do not try to use the same connection handle.

2354	X'0932'	MQRC_UOW_ENLISTMENT_ERROR
		This reason code can occur for a variety of reasons. The most likely reason is that an object created by a DTC transaction does not issue a transactional MQI call until after the DTC transaction has timed out. (If the DTC transaction times out after a transactional MQI call has been issued, reason code MQRC_HANDLE_IN_USE_FOR_UOW is returned by the failing MQI call.)
		Another cause of MQRC_UOW_ENLISTMENT_ERROR is incorrect installation; Windows NT Service pack must be installed after the Windows NT Option pack.
		This reason code occurs only on Windows NT.
		Corrective action: Check the DTC "Transaction timeout" value. If necessary, verify the NT installation order.
2355	X'0933'	MQRC_UOW_MIX_NOT_SUPPORTED
		The mixture of calls used by the application to perform operations within a unit of work is not supported. In particular, it is not possible to mix within the same process a local unit of work coordinated by the queue manager with a global unit of work coordinated by DTC (Distributed Transaction Coordinator).
		An application may cause this mixture to arise if some objects in a package are coordinated by DTC and others are not. It can also occur if transactional MQI calls from an MTS client are mixed with transactional MQI calls from a library package transactional MTS object.
		No problem arises if all transactional MQI calls originate from transactional MTS objects, or all transactional MQI calls originate from nontransactional MTS objects. But when a mixture of styles is used, the first style used fixes the style for the unit of work, and subsequent attempts to use the other style within the process fail with reason code MQRC_UOW_MIX_NOT_SUPPORTED.
		When an application is run twice, scheduling factors in the operating system mean that it is possible for the queue-manager-coordinated transactional calls to fail in one run, and for the DTC-coordinated transactional calls to fail in the other run.
		This reason code occurs only on Windows NT.
		Corrective action: Check that the "MTS Transaction Support' attribute defined for the object's class is set correctly. If necessary, modify the application so that objects executing within different units of work do not try to use the same connection handle.
2356	X'0934'	MQRC_WXP_ERROR
		An MQXCLWLN call was issued from a cluster workload exit to obtain the address of the next record in the chain, but the workload exit parameter structure *ExitParms* is not valid, for one of the following reasons: • The parameter pointer is not valid. (It is not always possible to detect parameter pointers that are not valid; if not detected, unpredictable results occur.) • The *StrucId* field is not MQWXP_STRUC_ID. • The *Version* field is not MQWXP_VERSION_2.
		Corrective action: Ensure that the parameter specified for *ExitParms* is the MQWXP structure that was passed to the exit when the exit was invoked.

Completion and reason codes

2357	X'0935'	MQRC_CURRENT_RECORD_ERROR
		An MQXCLWLN call was issued from a cluster workload exit to obtain the address of the next record in the chain, but the address specified by the *CurrentRecord* parameter is not the address of a valid record. *CurrentRecord* must be the address of a destination record (MQWDR), queue record (MQWQR), or cluster record (MQWCR) residing within the cluster cache.
		Corrective action: Ensure that the cluster workload exit passes the address of a valid record residing in the cluster cache.
2358	X'0936'	MQRC_NEXT_OFFSET_ERROR
		An MQXCLWLN call was issued from a cluster workload exit to obtain the address of the next record in the chain, but the offset specified by the *NextOffset* parameter is not valid. *NextOffset* must be the value of one of the following fields: • *ChannelDefOffset* field in MQWDR • *ClusterRecOffset* field in MQWDR • *ClusterRecOffset* field in MQWQR • *ClusterRecOffset* field in MQWCR
		Corrective action: Ensure that the value specified for the *NextOffset* parameter is the value of one of the fields listed above.
2359	X'0937'	MQRC_NO_RECORD_AVAILABLE
		An MQXCLWLN call was issued from a cluster workload exit to obtain the address of the next record in the chain, but the current record is the last record in the chain.
		Corrective action: None.
2360	X'0938'	MQRC_OBJECT_LEVEL_INCOMPATIBLE
		An MQOPEN or MQPUT1 call was issued, but the definition of the object to be accessed is not compatible with the queue manager to which the application has connected. The object definition was created or modified by a different version of the queue manager.
		If the object to be accessed is a queue, the incompatible object definition could be the object specified by the MQOD structure on the call, or one of the object definitions used to resolve the specified object (for example, the base queue to which an alias queue resolves, or the transmission queue to which a remote queue or queue-manager alias resolves).
		This reason code occurs only on OS/390.
		Corrective action: The application must be run on a queue manager that is compatible with the object definition. Refer to the *MQSeries for OS/390 Concepts and Planning Guide* and the *MQSeries for OS/390 System Setup Guide* for information about compatibility and migration between different versions of the queue manager.
6100	X'17D4'	MQRC_REOPEN_EXCL_INPUT_ERROR
		An open object does not have the correct ImqObject **open options** and requires one or more additional options. An implicit reopen is required but closure has been prevented.
		Closure has been prevented because the queue is open for exclusive input and closure might result in the queue being accessed by another process or thread, before the queue is reopened by the process or thread that presently has access.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: set the **open options** explicitly to cover all eventualities so that implicit reopening is not required.

6101	X'17D5'	MQRC_REOPEN_INQUIRE_ERROR
		An open object does not have the correct ImqObject **open options** and requires one or more additional options. An implicit reopen is required but closure has been prevented.
		Closure has been prevented because one or more characteristics of the object need to be checked dynamically prior to closure, and the **open options** do not already include MQOO_INQUIRE.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: set the **open options** explicitly to include MQOO_INQUIRE.
6102	X'17D6'	MQRC_REOPEN_SAVED_CONTEXT_ERR
		An open object does not have the correct ImqObject **open options** and requires one or more additional options. An implicit reopen is required but closure has been prevented.
		Closure has been prevented because the queue is open with MQOO_SAVE_ALL_CONTEXT, and a destructive get has been performed previously. This has caused retained state information to be associated with the open queue and this information would be destroyed by closure.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: set the **open options** explicitly to cover all eventualities so that implicit reopening is not required.
6103	X'17D7'	MQRC_REOPEN_TEMPORARY_Q_ERROR
		An open object does not have the correct ImqObject **open options** and requires one or more additional options. An implicit reopen is required but closure has been prevented.
		Closure has been prevented because the queue is a local queue of the definition type MQQDT_TEMPORARY_DYNAMIC, that would be destroyed by closure.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: set the **open options** explicitly to cover all eventualities so that implicit reopening is not required.
6104	X'17D8'	MQRC_ATTRIBUTE_LOCKED
		An attempt has been made to change the value of an attribute of an object while that object is open, or, for an ImqQueueManager object, while that object is connected. Certain attributes cannot be changed in these circumstances. Close or disconnect the object (as appropriate) before changing the attribute value.
		An object may have been connected and/or opened unexpectedly and implicitly in order to perform an MQINQ call. Check the attribute cross-reference table in the *MQSeries Using C++* book to determine whether any of your method invocations result in an MQINQ call.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: include MQOO_INQUIRE in the ImqObject **open options** and set them earlier.
6105	X'17D9'	MQRC_CURSOR_NOT_VALID
		The browse cursor for an open queue has been invalidated since it was last used by an implicit reopen.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: set the ImqObject **open options** explicitly to cover all eventualities so that implicit reopening is not required.

Completion and reason codes

6106	X'17DA'	MQRC_ENCODING_ERROR
		The encoding of the (next) message item needs to be MQENC_NATIVE for pasting.
		This reason code occurs in the MQSeries C++ environment.
6107	X'17DB'	MQRC_STRUC_ID_ERROR
		The structure id for the (next) message item, which is derived from the 4 characters beginning at the data pointer, is either missing or is inconsistent with the class of object into which the item is being pasted.
		This reason code occurs in the MQSeries C++ environment.
6108	X'17DC'	MQRC_NULL_POINTER
		A null pointer has been supplied where a nonnull pointer is either required or implied.
		This reason code occurs in the MQSeries C++ environment.
6109	X'17DD'	MQRC_NO_CONNECTION_REFERENCE
		The **connection reference** is null. A connection to an ImqQueueManager object is required.
		This reason code occurs in the MQSeries C++ environment.
6110	X'17DE'	MQRC_NO_BUFFER
		No buffer is available. For an ImqCache object, one cannot be allocated, denoting an internal inconsistency in the object state that should not occur.
		This reason code occurs in the MQSeries C++ environment.
6111	X'17DF'	MQRC_BINARY_DATA_LENGTH_ERROR
		The length of the binary data is inconsistent with the length of the target attribute. Zero is a correct length for all attributes.
		• The correct length for an **accounting token** is MQ_ACCOUNTING_TOKEN_LENGTH.
		• The correct length for an **alternate security id** is MQ_SECURITY_ID_LENGTH.
		• The correct length for a **correlation id** is MQ_CORREL_ID_LENGTH.
		• The correct length for a **facility token** is MQ_FACILITY_LENGTH.
		• The correct length for a **group id** is MQ_GROUP_ID_LENGTH.
		• The correct length for a **message id** is MQ_MSG_ID_LENGTH.
		• The correct length for an **instance id** is MQ_OBJECT_INSTANCE_ID_LENGTH.
		• The correct length for a **transaction instance id** is MQ_TRAN_INSTANCE_ID_LENGTH.
		• The correct length for a **message token** is MQ_MSG_TOKEN_LENGTH.
		This reason code occurs in the MQSeries C++ environment.
6112	X'17E0'	MQRC_BUFFER_NOT_AUTOMATIC
		A user-defined (and managed) buffer cannot be resized. A user-defined buffer can only be replaced or withdrawn. A buffer must be automatic (system-managed) before it can be resized.
		This reason code occurs in the MQSeries C++ environment.
6113	X'17E1'	MQRC_INSUFFICIENT_BUFFER
		There is insufficient buffer space available after the data pointer to accommodate the request. This might be because the buffer cannot be resized.
		This reason code occurs in the MQSeries C++ environment.

6114	X'17E2'	MQRC_INSUFFICIENT_DATA
		There is insufficient data after the data pointer to accommodate the request.
		This reason code occurs in the MQSeries C++ environment.
6115	X'17E3'	MQRC_DATA_TRUNCATED
		Data has been truncated when copying from one buffer to another. This might be because the target buffer cannot be resized, or because there is a problem addressing one or other buffer, or because a buffer is being downsized with a smaller replacement.
		This reason code occurs in the MQSeries C++ environment.
6116	X'17E4'	MQRC_ZERO_LENGTH
		A zero length has been supplied where a positive length is either required or implied.
		This reason code occurs in the MQSeries C++ environment.
6117	X'17E5'	MQRC_NEGATIVE_LENGTH
		A negative length has been supplied where a zero or positive length is required.
		This reason code occurs in the MQSeries C++ environment.
6118	X'17E6'	MQRC_NEGATIVE_OFFSET
		A negative offset has been supplied where a zero or positive offset is required.
		This reason code occurs in the MQSeries C++ environment.
6119	X'17E7'	MQRC_INCONSISTENT_FORMAT
		The format of the (next) message item is inconsistent with the class of object into which the item is being pasted.
		This reason code occurs in the MQSeries C++ environment.
6120	X'17E8'	MQRC_INCONSISTENT_OBJECT_STATE
		There is an inconsistency between this object, which is open, and the referenced ImqQueueManager object, which is not connected.
		This reason code occurs in the MQSeries C++ environment.
6121	X'17E9'	MQRC_CONTEXT_OBJECT_NOT_VALID
		The ImqPutMessageOptions **context reference** does not reference a valid ImqQueue object. The object has been previously destroyed.
		This reason code occurs in the MQSeries C++ environment.
6122	X'17EA'	MQRC_CONTEXT_OPEN_ERROR
		The ImqPutMessageOptions **context reference** references an ImqQueue object that could not be opened to establish a context. This may be because the ImqQueue object has inappropriate **open options**. Inspect the referenced object **reason code** to establish the cause.
		This reason code occurs in the MQSeries C++ environment.
6123	X'17EB'	MQRC_STRUC_LENGTH_ERROR
		The length of a data structure is inconsistent with its content. For an MQRMH, the length is insufficient to contain the fixed fields and all offset data.
		This reason code occurs in the MQSeries C++ environment.

Completion and reason codes

6124	X'17EC'	MQRC_NOT_CONNECTED
		A method failed because a required connection to a queue manager was not available, and a connection cannot be established implicitly because the IMQ_IMPL_CONN flag of the ImqQueueManager **behavior** class attribute is FALSE.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: establish a connection to a queue manager and retry.
6125	X'17ED'	MQRC_NOT_OPEN
		A method failed because an MQSeries object was not open, and opening cannot be accomplished implicitly because the IMQ_IMPL_OPEN flag of the ImqObject **behavior** class attribute is FALSE.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: open the object and retry.
6126	X'17EE'	MQRC_DISTRIBUTION_LIST_EMPTY
		An ImqDistributionList failed to open because there are no ImqQueue objects referenced.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: establish at least one ImqQueue object in which the **distribution list reference** addresses the ImqDistributionList object, and retry.
6127	X'17EF'	MQRC_INCONSISTENT_OPEN_OPTIONS
		A method failed because the object is open, and the ImqObject **open options** are inconsistent with the required operation. The object cannot be reopened implicitly because the IMQ_IMPL_OPEN flag of the ImqObject **behavior** class attribute is false.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: open the object with appropriate ImqObject **open options** and retry.
6128	X'17FO'	MQRC_WRONG_VERSION
		A method failed because a version number specified or encountered is either incorrect or not supported.
		For the ImqCICSBridgeHeader class, the problem is with the **version** attribute.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: If you are specifying a version number, use one that is supported by the class. If you are receiving message data from another program, ensure that both programs are using consistent and supported version numbers.
6129	X'17F1'	MQRC_REFERENCE_ERROR
		An object reference is invalid.
		There is a problem with the address of a referenced object. At the time of use, the address of the object is nonnull, but is invalid and cannot be used for its intended purpose.
		This reason code occurs in the MQSeries C++ environment.
		Corrective action: Check that the referenced object is neither deleted nor out of scope, or remove the reference by supplying a null address value.

Notices

This information was developed for products and services offered in the United States. IBM may not offer the products, services, or features discussed in this information in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this information. The furnishing of this information does not give you any license to these patents. You can send license inquiries, in writing, to:

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

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

    IBM World Trade Asia Corporation
    Licensing
    2-31 Roppongi 3-chome, Minato-ku
    Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the information. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this information at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Notices

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

AIX	AS/400	CICS
DB2	FFST	IBM
IBMLink	Language Environment	MQSeries
MVS	OS/390	S/390
VSE/ESA		

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

Active Directory, Microsoft, Windows, and Windows NT are trademarks of Microsoft 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.

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

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

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 ®