IBM MQ multi-instance informational-error messages, events, return codes encountered during strmqm and endmqm

https://www.ibm.com/support/pages/node/598101

Date last updated: 16-Mar-2023

Angel Rivera – rivera@us.ibm.com
IBM MQ Support

+++ Objective

The objective of this document is to capture the following items that are relevant during the start and stop tasks, when working with active and standby instances of an IBM MQ multi-instance queue manager:
- all the error and information messages,
- return codes,
- configuration events,
- locked files,
- qmstatus.ini

There are 15 scenarios in total:
- 5 scenarios for strmqm
- 9 scenarios for endmqm
- 1 scenario for incomplete upgrade

+ 15-Mar-2023
Overall revision.

The scope for this document is for the strmqm, endmqm and dspmq commands.

If you encountered an FDC related to multi-instance, please consult the following technote:
https://www.ibm.com/support/pages/node/473409
Probeid ZX155001 component zxcFileLockMonitorThread error
lrcE_S_Q_MGR_UNRESPONSIVE

How best use this document:
Search the error number or phrase, and you will be taken to one of the tables at the beginning of the document.
The tables provide a summary and refer to specific scenarios, which are described in more details later on in the document.

Note about the flag -i or -is for endmqm for a graceful switchover from Active:
The scenarios for stopping a queue manager (endmqm -i and endmqm -is) use the recommended use of the -i option (immediate), which is not aggressive but allows the queue manager to end even when there are clients connected, that is, the queue manager waits for any current put/get to end:

```
endmqm -is QmgrName
```

Note for ending a Standby:
The -i option is not compatible with the -x option. Thus, when ending a Standby, the command "endmqm -ix" is not valid and the following should be used:

```
endmqm -x QmgrName
```

Summary of important scenarios:

Scenario 1: strmqm (no -x option) to start Active in host1. No standby in host2. OK.
- This is the way to start a STANDALONE queue manager.
- It does NOT allow a Standby instance to start.

.
Scenario 3: strmqm -x to start Active in host1. No standby in host2. OK.
This is the way to start the Active instance for a queue manager for multi-instance.

.
Scenario 5: strmqm -x to start Standby in host2, while Active is running in host1. OK.
This is the way to start the Standby instance for a queue manager for multi-instance.

.
Scenario 6: endmqm -i to stop Active in host1. No Standby in host2. OK.
This is the way to end a STANDALONE queue manager.

.
Scenario 7: endmqm -i to stop Active in host1 and to stop Standby in host2. OK.
- This is the way to stop BOTH instances of the queue manager.
- No switchover is intended to happen.

.
Scenario 10: endmqm -is to stop Active in host1, and Standby running in host2, causing switchover to Standby. OK.
- This is the way to do a graceful SWITCHOVER from the Active in host1 to the Standby in host2.
- This will end the Active in host1 and the Standby in host2 will become the new Active.

.
Scenario 14: endmqm -x to end Standby, but Active continues to run. OK.
This is the way to end a STANDBY instance, without disturbing the Active. The Active continues to run.

++ (02-Oct-2019) Summary of valid ending scenarios

```
Scenario 6: Active in host1. No Standby in host2. It is in a sense, a Standalone queue
manager.
Objective: To stop Active in host1 for a standalone queue manager:
host1:   endmqm -i QmgrName
host2:   <not applicable>
.
Scenario 7: Active in host1. Standby in host2. Typical Multi-instance queue manager.
Objective: To stop BOTH instances and NO switchover is intended to happen.
host1:   endmqm -i QmgrName
host2:   <not applicable>
.
Scenario 10: Active in host1. Standby in host2. Typical Multi-instance queue manager.
Objective: To stop Active in host1, causing graceful switchover to Standby in host2.
This will end the Active in host1 and the Standby in host2 will become the new Active.
host1:   endmqm -is QmgrName
host2:   <not applicable>
.
Scenario 14: Active in host1. Standby in host2. Typical Multi-instance queue manager.
Objective: To stop Standby in host2, without without disturbing the Active. The Active
continues to run. (May need to do maintenace in host2).
host1:   <not applicable>
host2:   endmqm -x QmgrName
```

++ There are 3 appendixes:

Appendix 1: Entries in qmstatus.ini regarding Multi-Instance
Appendix 2: Highlights of file locking when using multi-instance
Appendix 3: Return codes for strmqm and endmqm

Note:
This list of scenarios is an extension of the Table 1 shown at the bottom of the following web page from the online MQ manual:

https://www.ibm.com/docs/en/ibm-mq/9.3?topic=reference-endmqm-end-queue-manager
IBM MQ > IBM MQ 9.3 > IBM MQ > Reference > Administration reference > IBM MQ Control commands > The control commands >
endmqm (end queue manager)

+++ Topology and versions of software

- Using 3 hosts with RHEL 8.6 (x86_64):
  host1 => use the exported file system from host3
  host2 => use the exported file system from host3
  host3 => exports the NFS V4

- MQ version 9.3
  Queue manager name: QMMI1 (port 1421)
  DataPath=/mqha/qmgrs/QMMI1

The following MQ command provides the details from the mqs.ini about this queue manager:

```
mqm@host1: /var/mqm
$ dspmqinf QMMI1
QueueManager:
   Name=QMMI1
   Directory=QMMI1
   Prefix=/var/mqm
   DataPath=/mqha/qmgrs/QMMI1
   InstallationName=Installation1
```

The DataPath indicates that the error logs will be located at the following shared directory structure:
  /mqha/qmgrs/QMMI1/errors

In addition, there is a file that has status information that is used by the instances:
  /mqha/qmgrs/QMMI1/qmstatus.ini

The MQ FDCs will be located in each host under:
 /var/mqm/errors

+++ Baseline scenario

There are no instances running on either of the hosts:

+ In host1:

```
mqm@host1: /home/mqm
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Ended immediately)
.
$ ps -ef | grep mq | grep QMMI1
mqm      16576 16457  0 04:41 pts/1    00:00:00 grep QMMI1
```

+ In host2:

```
mqm@host2: /mqha/qmgrs/QMMI1/errors
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Ended immediately)

mqm@host2: /mqha/qmgrs/QMMI1/errors
$ ps -ef | grep mq | grep QMMI1
mqm      16949 13871  0 11:52 pts/1    00:00:00 grep QMMI1
```

+++ Scenarios for Starting an instance (strmqm)

#	Command	Local machine	Remote machine	Results
1	strmqm To start Active in host1, but not using -x	host1 No instance is running	host2 No instance is running	host1: Active starts qmstatus.ini: PermitStandby=No Message: Informational: AMQ8003I Generated event: Failover Not Permitted [31] . host2: No Standby.
2	strmqm or strmqm -x Trying to start queue manager in host2	host2 No instance is running	host1 Active was started: strmqm with no -x (Scen #1)	host1: Active continues to run. . host2: Standby does not start. RC: 43 The queue manager already has an active instance. The queue manager does not permit standby instances. . Solution: Stop Active in host1. Restart it with -x. See Scen #3: strmqm -x QmgrName
3	strmqm -x To start Active in host1. Using -x	host1 No instance is running	host2 No instance is running	host1: Active starts qmstatus.ini: PermitStandby=Yes Message: Informational: AMQ8003I Generated event: Failover Permitted [30] . host2: No Standby.
4	strmqm To start Standby in host2, but not using -x	host2 No instance is running	host1 Active was started: strmqm -x (Scen #3)	host1: Active instance continues to run. . host2: Standby cannot start. Message, error: IBM MQ queue manager 'QMMI1' is already running elsewhere. It permits standby instances. RC: 31 . Solution: Restart Standby in host2. Scen #5: strmqm -x QmgrName
5	strmqm -x To start Standby in host2. Using -x	host2 No instance is running	host1 Active was started: strmqm -x (Scen #3)	host1: Active instance continues to run. . host2: Standby started successfully in host2. Message, informational: - A standby instance of queue manager 'QMMI1' has been started. The active instance is running elsewhere. - AMQ8060I RC: 30

+++ Scenarios for Ending an instance (enqmqm)

#	Command	Local machine	Remote machine	Results
6	endmqm -i To end Active in host1	host1 Active is running. Scen #1 Scen #3	host2 No instance is running	host1: Active ends qmstatus.ini: CurrentStatus=EndedImmediately Message: Informational: AMQ8004I . host2: No Standby.
7	endmqm -i To end Active in host1 and Standby in host2	host1 Active is running. Scen #3	host2 Standby is running. Scen #5	host1: Active ends qmstatus.ini: CurrentStatus=EndedImmediately Message: Informational: AMQ8004I . host2: Standby ends Message: Informational: AMQ8355E
8	endmqm -i To try to end Standby in host2	host2 Standby is running. Scen #5	host1 Active is running. Scen #3	host1: Active continues to run. . host2: Standby continues to run. It is NOT ended. Message: Informational: AMQ8368E RC 90 . Solution: To only end the Standby, see Scen #14. Use in host2: endmqm -x QmgrName
9	endmqm -is To try to end Active in host1. Specifying -s for switchover.	host1 Active is running. Scen #3	host2 No instance is running	host1: Active continues to run because there is no Standby to do a switchover. Message: Informational: AMQ7276E RC 77 . host2: No Standby. . Solution: To end the Active, see Scen #6. Do not specify -is, use only -i. Use in host1: endmqm -i QmgrName
10	endmqm -is To switchover the Active from host1 to host2	host1 Active is running. Scen #3	host2 Standby is running. Scen #5	host1: Active ends. Message: Informational: AMQ8004I . host2: Standby becomes the new Active Message: Informational: AMQ8352I and AMQ8003I Generated event: Standby Activated [32]

11	endmqm -is To try to end Standby in host2	host2 Standby is running. Scen #5	host1 Active is running. Scen #3	host1: Active continues to run. . host2: Standby continues to run. It is NOT ended. Message: Informational: AMQ8368E RC 90 . Solution: To only end the Standby, see Scen #14. Use in host2: endmqm -x QmgrName
12	endmqm -x To try to end Active in host1.	host1 Active is running. Scen #3	host2 No instance is running	host1: Active continues to run. Message: Informational: AMQ8367E RC 79 . host2: No Standby. . Solution: To end the Active, see Scen #6. Do not specify -x, use only -i. Use in host1: endmqm -i QmgrName
13	endmqm -x To try to end Active in host1.	host1 Active is running. Scen #3	host2 Standby is running. Scen #5	host1: Active continues to run. Message: Informational: AMQ8367E RC 79 . host2: No Standby. . Solution: 1) If you want to stop both the Active and the Standby, see Scen #7. Do not specify -x, use -i. Use in host1: endmqm -i QmgrName 2) If you want to switchover the Active from host1 to host2, see Scen #10. Do not specify -x, use -is. Use in host1: endmqm -is QmgrName
14	endmqm -x To end Standby in host2	host2 Standby is running. Scen #5	host1 Active is running. Scen #3	host1: Active continues to run. . host2: Standby ends. Message: Informational: AMQ8004I

Return codes (RCs):

30 : A standby instance of the queue manager started. The active instance is running elsewhere
31 : The queue manager already has an active instance. The queue manager permits standby instances.
43 : The queue manager already has an active instance. The queue manager does not permit standby instances.
77 : IBM MQ queue manager cannot switch over
79 : Active instance of IBM MQ queue manager QmgrName not ended
90 : Standby instance of IBM MQ queue manager QmgrName not ended

Error and Information messages in error log of queue manager:

AMQ7276E: IBM MQ queue manager cannot switch over.

AMQ8003I: IBM MQ queue manager 'QMMI1' started using V9.3.2.0

AMQ8004I: IBM MQ queue manager 'QMMI1' ended.

AMQ8060I: IBM MQ queue manager 'QMMI1' started as a standby instance.

AMQ8352I: IBM MQ queue manager 'QMMI1' becoming the active instance.

AMQ8355E: IBM MQ standby queue manager instance 'QMMI1' not permitted to become the active instance.
EXPLANATION:
IBM MQ standby queue manager instance 'QMMI1' obtained a lock on its data in the file-system but was not permitted to become the active instance.
The most likely cause is that the active queue manager instance was stopped without permitting a switchover.

AMQ8367E: Active instance of IBM MQ queue manager 'QMMI1' not ended.

AMQ8368E: Standby instance of IBM MQ queue manager 'QMMI1' not ended.

+++ Scenario 1: strmqm (no -x option) to start Active in host1. No standby in host2. OK.

Notes:
- This is the way to start a STANDALONE queue manager.
- It does NOT allow a Standby instance to start.

Prerequisite conditions:
host1: No Active instance.
host2: No Standby instance.

Result:
host1: Active started successfully in host1.
host2: Because -x was NOT used, then the Standby will not start in Scenario #2.

+ host1:

Start command:
$ strmqm QMMI1

Output from command:
IBM MQ queue manager 'QMMI1' starting.
The queue manager is associated with installation 'Installation1'.
6 log records accessed on queue manager 'QMMI1' during the log replay phase.
Log replay for queue manager 'QMMI1' complete.
Transaction manager state recovered for queue manager 'QMMI1'.
Plain text communication is enabled.
IBM MQ queue manager 'QMMI1' started using V9.3.2.0.

Return code:
$ echo $?
0

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running)
    INSTANCE(host1) MODE(Active)

Entries from qmstatus.ini:
Notice: **PermitStandby=No**

```
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
   CurrentStatus=Running
   PermitStandby=No
   PermitFailover=Yes
   PlatformSignature=20497
   PlatformString=Linux 4.18.0-425.13.1.el8_7.x86_64
   Version=9.3.2.0
   MinimumRequiredVersion=9.3.2.0
```

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG

```
03/15/2023 01:09:46 PM - Process(158206.1) User(mqm) Program(amqzxma0)
                    Host(riggioni1.fyre.ibm.com) Installation(Installation1)
                    VRMF(9.3.2.0) QMgr(QMMI1)
                    Time(2023-03-15T20:09:46.680Z)
                    CommentInsert1(9.3.2.0)
                    CommentInsert3(QMMI1)
AMQ8003I: IBM MQ queue manager 'QMMI1' started using V9.3.2.0.
EXPLANATION:
IBM MQ queue manager 'QMMI1' started using V9.3.2.0.
ACTION:
None.
```

Generated event:

```
SYSTEM.ADMIN.QMGR.EVENT: 1 events
  (date) Queue Manager Active [2222]
     Event Type  : Queue Manager
     Queue Manager Name : QMMI1
     Host Name          : host1
     Reason Qualifier   : Failover Not Permitted [31]
```

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:

```
lsof /mqha/qmgrs/QMMI1/master
  See Appendix 2, section "++ Looking at the locks for the Active instance"
lsof /mqha/qmgrs/QMMI1/active
  See Appendix 2, section "++ Looking at the locks for the Active instance"
lsof /mqha/qmgrs/QMMI1/standby
  No locks
```

+ host2:

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                   STATUS(Running elsewhere)
    INSTANCE(host1) MODE(Active)
```

+++ Scenario 2: strmqm to try to start queue manager in host2, while Active is running in host1 (but not started with -x). FAILS.

Prerequisite conditions:
host1: Active instance running, NOT started with -x, which will not allow a standby.
host2: No Standby running.

Result:
host1: Active instance continues to run.
host2: The instance did not start in host2, because Active was not started with the -x flag.

Solution:
Stop Active in host1 and restart it with -x. See Scenario #3.
Use: strmqm -x QmgrName

+ host1:

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                   STATUS(Running)
    INSTANCE(host1) MODE(Active)
```

Entries from qmstatus.ini:
Notice: **PermitStandby=No**

```
QueueManagerStatus:
   CurrentStatus=Running
   PermitStandby=No
   PermitFailover=Yes
   PlatformSignature=20497
   PlatformString=Linux 4.18.0-425.13.1.el8_7.x86_64
   Version=9.3.2.0
   MinimumRequiredVersion=9.3.2.0
```

+ host2:

Notice that qmstatus.ini shows:
PermitStandby=No

Try to start the queue manager (it does not matter if the flag -x is used or not).
There are 2 cases, one without -x and the other with the -x

+ Case 1: Start command: Not using -x:
$ strmqm QMMI1

Output from command:
IBM MQ queue manager 'QMMI1' starting.
The queue manager is associated with installation 'Installation1'.
IBM MQ queue manager 'QMMI1' is already running elsewhere. It does not
permit standby instances.

Return Code:
$ echo $?
43

RC 43 means: The queue manager already has an active instance. The queue manager does not permit standby instances.

+ Case 2: Start command: Using -x:
$ strmqm -x QMMI1

Output from command:
IBM MQ queue manager 'QMMI1' starting.
The queue manager is associated with installation 'Installation1'.
IBM MQ queue manager 'QMMI1' is already running elsewhere. It does not
permit standby instances.

Return Code:
$ echo $?
43

Output of dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running elsewhere)
INSTANCE(host1) MODE(Active)

Entries from qmstatus.ini:
No change

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
No new entries

Generated event:
No new entries.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No lock was acquired by the strmqm in host2.

+++ Scenario 3: strmqm -x to start Active in host1. No standby in host2. OK.

Note: This is the way to start the Active instance for a queue manager for multi-instance.

Prerequisite conditions:
host1: No Active instance running.
host2: No Standby instance running.

Result:
host1: Active stated successfully in host1.
host2: Because -x was used, then this will allow the Standby to start in Scenario 5.

+ host1:

Start command:
$ strmqm -x QMMI1

Output from command:
IBM MQ queue manager 'QMMI1' starting.
The queue manager is associated with installation 'Installation1'.
6 log records accessed on queue manager 'QMMI1' during the log replay phase.
Log replay for queue manager 'QMMI1' complete.
Transaction manager state recovered for queue manager 'QMMI1'.
Plain text communication is enabled.
IBM MQ queue manager 'QMMI1' started using V9.3.2.0.

Return code:
$ echo $?
0

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running)
INSTANCE(host1) MODE(Active)

Entries from qmstatus.ini:
Notice: **PermitStandby=Yes**

$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
CurrentStatus=Running
PermitStandby=Yes
PermitFailover=Yes
PlatformSignature=20497
PlatformString=Linux 4.18.0-425.13.1.el8_7.x86_64

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
AMQ8003I: IBM MQ queue manager 'QMMI1' started using V9.3.2.0.

Generated event:
SYSTEM.ADMIN.QMGR.EVENT: 1 events
(date) Queue Manager Active [2222]
Event Type : Queue Manager
Queue Manager Name : QMMI1
Host Name : host1
Reason Qualifier : Failover Permitted [30]

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
lsof /mqha/qmgrs/QMMI1/master
See Appendix 2, section "++ Looking at the locks for the Active instance"
lsof /mqha/qmgrs/QMMI1/active
See Appendix 2, section "++ Looking at the locks for the Active instance"
lsof /mqha/qmgrs/QMMI1/standby
No locks

+ host2:

$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running elsewhere)
INSTANCE(host1) MODE(Active)

+++ Scenario 4: strmqm (no -x) to start Standby in host2, while Active is running in host1. FAILS.

Prerequisite conditions:
host1: Active instance running in host1, started with -x (permits standby).
host2: No Standby running.

Result:
host1: Active instance continues to run.
host2: Standby cannot start.

Solution:
Because Active was started in host1 with -x, it is necessary to start the Standby in host2 with -x. See Scenario #5. Use: strmqm -x QmgrName

+ host1:

See:
Scenario 3: strmqm -x to start Active in host1. No standby in host2.
Resulting in Active instance running in host1 and no Standby in host2.

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running)
    INSTANCE(host1) MODE(Active)

+ host2:

Notice that -x is not used in this scenario.
This will cause a failure, because without the -x, the instance tries to be an Active and will not accept to become a Standby.

Start command:
$ strmqm QMMI1

Output from command:
IBM MQ queue manager 'QMMI1' starting.
The queue manager is associated with installation 'Installation1'.
IBM MQ queue manager 'QMMI1' is already running elsewhere. It permits
standby instances.

Return code:
echo $?
31

This RC 31 means:
The queue manager already has an active instance. The queue manager permits standby instances.

Output from dspmq:
```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                             STATUS(Running elsewhere)
    INSTANCE(host1) MODE(Active)
```

Entries from qmstatus.ini: (no change)
```
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
   CurrentStatus=Running
   PermitStandby=Yes
   PermitFailover=Yes
   PlatformSignature=20497
   PlatformString=Linux 4.18.0-425.13.1.el8_7.x86_64
```

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
No new entries

Generated event:
No new events

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
```
lsof /mqha/qmgrs/QMMI1/standby
```
See Appendix 2, section "++ Looking at the locks for the Standby instance"

+++ Scenario 5: strmqm -x to start Standby in host2, while Active is running in host1. OK.

Note: This is the way to start the Standby instance for a queue manager for multi-instance.

Prerequisite conditions:
host1: Active instance running in host1, started with -x (permits standby).
host2: No Standby running.

Result:
host1: Active instance continues to run.
host2: Standby started successfully in host2.

+ host1:

See:
Scenario 3: strmqm -x to start Active in host1. No standby in host2.
Resulting in Active instance running in host1 and no Standby in host2.

Output from dspmq:

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running)
   INSTANCE(host1) MODE(Active)

$ dspmq -xf -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running)
   INSTANCE(host1) MODE(Active)
      master(host1.fyre.ibm.com,7210871600286817189)
      active(host1.fyre.ibm.com,7210871600286817189)
```

File Locking:
Note: After Standby starts in host2, a lock will be seen for the file 'standby' in host1.
See Appendix 2, section "++ Looking at the locks for the Active instance"

+ host2:

Start command:

```
$ strmqm -x QMMI1
```

Output from command:

```
IBM MQ queue manager 'QMMI1' starting.
The queue manager is associated with installation 'Installation2'.
Plain text communication is enabled.
A standby instance of queue manager 'QMMI1' has been started. The active
instance is running elsewhere.
```

Return code:
$ echo $?
30

This RC 30 means:
A standby instance of the queue manager started. The active instance is running elsewhere

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running as standby)
INSTANCE(host1) MODE(Active)
INSTANCE(host2) MODE(Standby)

$ dspmq -xf -m QMMI1
QMNAME(QMMI1) STATUS(Running as standby)
INSTANCE(host1) MODE(Active)
INSTANCE(host2) MODE(Standby)
master(host1,7210871600286817189)
active(host1,7210871600286817189)
standby(host2,7210872480757888503)

Entries from qmstatus.ini: (no change)
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
CurrentStatus=Running
PermitStandby=Yes
PermitFailover=Yes
PlatformSignature=20497
PlatformString=Linux 2.6.27.19-5-default

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
AMQ8060I: IBM MQ queue manager 'QMMI1' started as a standby instance.
EXPLANATION:
Queue manager 'QMMI1' started as a standby instance, ready to become the active instance if the existing active instance fails.

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
lsof /mqha/qmgrs/QMMI1/standby
See Appendix 2, section "++ Looking at the locks for the Standby instance"

+++ Scenario 6: endmqm -i to stop Active in host1. No Standby in host2. OK.

Note: This is the way to end a STANDALONE queue manager.

Prerequisite conditions:
host1: Active running in host1
Either, started without -x (Scenario #1),
or started with -x (Scenario #3)
host2: No Standby running.

Result:
host1: Active stopped in host1.
host2: No Standby in host2.

+ host1:

End command:
$ endmqm -i QMMI1

Output from command:
Waiting for queue manager 'QMMI1' to end.
IBM MQ queue manager 'QMMI1' ending.
IBM MQ queue manager 'QMMI1' ended.

Return code:
$ echo $?
0

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Ended immediately)

Entries from qmstatus.ini (when Active started without -x)
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
CurrentStatus=EndedImmediately
PermitStandby=No
PermitFailover=No

Note: If the Active had started with -x, then the qmstatus.ini after the endmqm will look like:
QueueManagerStatus:
CurrentStatus=EndedImmediately
PermitStandby=Yes
PermitFailover=No

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
AMQ8004I: IBM MQ queue manager 'QMMI1' ended.
EXPLANATION:
IBM MQ queue manager 'QMMI1' ended.

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No locks.

+ host2:

$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Ended immediately)

+++ Scenario 7: endmqm -i to stop Active in host1 and to stop Standby in host2. OK.

Notes:
- This is the way to stop BOTH instances of the queue manager.
- No switchover is intended to happen.

Prerequisite conditions:
host1: Active instance running. Started with -x (Scenario #3)
host2: Standby instance running. Started with -x (Scenario #5)

Result:
host1: Active stopped.
host2: Standby stopped.

+ host1:

End command:
$ endmqm -i QMMI1

Output from command:
Waiting for queue manager 'QMMI1' to end.
IBM MQ queue manager 'QMMI1' ending.
IBM MQ queue manager 'QMMI1' ended.

Return code:
$ echo $?
0

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1)   STATUS(Ended immediately)

Entries from qmstatus.ini:
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
  CurrentStatus=EndedImmediately
  PermitStandby=Yes
  PermitFailover=No

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG

There are 2 entries, one from each host:

For host1:

```
03/15/2023 01:41:19 PM - Process(158649.1) User(mqm) Program(amqzxma0)
                Host(host1) Installation(Installation1)
                VRMF(9.3.2.0) QMgr(QMMI1)
                Time(2023-03-15T20:41:19.405Z)
                CommentInsert3(QMMI1)
AMQ8004I: IBM MQ queue manager 'QMMI1' ended.
EXPLANATION:
IBM MQ queue manager 'QMMI1' ended.
```

For host2:

```
03/15/2023 01:41:19 PM - Process(178091.1) User(mqm) Program(amqzxma0)
                Host(host2) Installation(Installation2)
                VRMF(9.3.2.0) QMgr(QMMI1)
                Time(2023-03-15T20:41:19.727Z)
                CommentInsert3(QMMI1)
AMQ8355E: IBM MQ standby queue manager instance 'QMMI1' not permitted to become
the active instance.
EXPLANATION:
IBM MQ standby queue manager instance 'QMMI1' obtained a lock on its data in
the file-system but was not permitted to become the active instance. The most
likely cause is that the active queue manager instance was stopped without
permitting a switchover.
```

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No locks.

+ host2:

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                   STATUS(Ended immediately)
```

+++ Scenario 8: endmqm -i to try to stop Standby in host2. Active running in host1. FAILS.

Note: This is similar to Scenario #11, but only using 2 flag: -i

Prerequisite conditions:
host1: Active instance running. Started with -x (Scenario #3)
host2: Standby instance running. Started with -x (Scenario #5)

Result:
host1: Active continues to run.
host2: Standby continues to run. It was not stopped.
RC 90 Standby instance of IBM MQ queue manager QmgrName not ended

Solution:
If you want to stop the Standby in host2, see Scenario #14.
Use: endmqm -x QmgrName

+ host1:

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running)
   INSTANCE(host1) MODE(Active)
   INSTANCE(host2) MODE(Standby)
```

+ host2:

End command:
```
$ endmqm -i QMMI1
```

Output from command:
```
AMQ8368E: Standby instance of IBM MQ queue manager 'QMMI1' not ended.
```

Return code:
```
$ echo $?
90
```

The RC 90 means:
Standby instance of IBM MQ queue manager QmgrName not ended

Output from dspmq:
```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running as standby)
   INSTANCE(host1) MODE(Active)
   INSTANCE(host2) MODE(Standby)
```

Entries from qmstatus.ini: (no change)
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
  CurrentStatus=Running
  PermitStandby=Yes
  PermitFailover=Yes

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
No new entries.

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No change in file locks.

+++ Scenario 9: endmqm -is to try to stop Active in host1. No Standby in host2. FAILS.

Note: The use of the -s flag is for doing a switchover, but if there is no Standby running, then there will not be a switchover.

Prerequisite conditions:
host1: Active instance running. Started with -x (Scenario #3)
host2: No Standby running.

Result:
host1: Active did not stop and continues to run.
RC 77 IBM MQ queue manager cannot switch over
host2: No Standby in host2.

Solution:
If you want to stop the Active in host1, see Scenario #6.
Do not specify -is, use only -i.
Use: endmqm -i QmgrName

+ host1:

Displaying status:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running)
INSTANCE(host1) MODE(Active)

End command:
$ endmqm -is QMMI1

Output from command:
AMQ7276E: IBM MQ queue manager cannot switch over.

Return code:
$ echo $?
77

The RC 77 means:
IBM MQ queue manager cannot switch over

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running)
INSTANCE(host1) MODE(Active)

Entries from qmstatus.ini: (no change)
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
  CurrentStatus=Running
  PermitStandby=Yes
  PermitFailover=Yes

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
No new entries.

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No change in file locks

+ host2:

$ dspmq -x -m QMMI1
QMNAME(QMMI1)   STATUS(Running elsewhere)
   INSTANCE(host1) MODE(Active)

+++ Scenario 10: endmqm -is to stop Active in host1, and Standby running in host2, causing switchover to Standby. OK.

Notes:
- This is the way to do a graceful SWITCHOVER from the Active in host1 to the Standby in host2.
- This will end the Active in host1 and the Standby in host2 will become the new Active.

Prerequisite conditions:
host1: Active instance running. Started with -x (Scenario #3)
host2: Standby instance running. Started with -x (Scenario #5)

Result:
host1: Active stopped.
host2: Standby became the new Active.

+ host1:

Displaying status:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running)
INSTANCE(host1) MODE(Active)
INSTANCE(host2) MODE(Standby)

End command:
$ endmqm -is QMMI1

Output from command:
IBM MQ queue manager 'QMMI1' ending.
IBM MQ queue manager 'QMMI1' ended, permitting switchover to a standby instance.

Return code:
$ echo $?
0

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running elsewhere)
INSTANCE(host2) MODE(Active)

Entries from qmstatus.ini: (no change)
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
  CurrentStatus=Running
  PermitStandby=Yes
  PermitFailover=Yes

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG

There are 3 relevant entries regarding the switchover:

1) The current Active in host1 was ended:

03/15/2023 01:50:52 PM - Process(158840.1) User(mqm) Program(amqzxma0)
                Host(host1) Installation(Installation1)
                VRMF(9.3.2.0) QMgr(QMMI1)
                Time(2023-03-15T20:50:52.331Z)
                CommentInsert3(QMMI1)
AMQ8004I: IBM MQ queue manager 'QMMI1' ended.
EXPLANATION:
IBM MQ queue manager 'QMMI1' ended.

2) The Standby in host2 became the new Active

03/15/2023 01:50:52 PM - Process(178193.1) User(mqm) Program(amqzxma0)
                Host(host2) Installation(Installation2)
                VRMF(9.3.2.0) QMgr(QMMI1)
                Time(2023-03-15T20:50:52.649Z)
                CommentInsert3(QMMI1)
AMQ8352I: IBM MQ queue manager 'QMMI1' becoming the active instance.
EXPLANATION:
The standby instance of queue manager instance 'QMMI1' is becoming the active instance.

3) The new Active in host2 is ready after the switchover:
03/15/2023 01:50:53 PM - Process(178193.1) User(mqm) Program(amqzxma0)
                Host(host2) Installation(Installation2)
                VRMF(9.3.2.0) QMgr(QMMI1)
                Time(2023-03-15T20:50:53.570Z)
                CommentInsert1(9.3.2.0)
                CommentInsert3(QMMI1)
AMQ8003I: IBM MQ queue manager 'QMMI1' started using V9.3.2.0.
EXPLANATION:
IBM MQ queue manager 'QMMI1' started using V9.3.2.0.

Generated event:
SYSTEM.ADMIN.QMGR.EVENT: 1 events
  (date) Queue Manager Active [2222]
    Event Type : Queue Manager
    Queue Manager Name : QMMI1
    Host Name   : host2
    Reason Qualifier   : Standby Activated [32]

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
There are no locks for the queue manager QMMI1 in host1.

+ host2:

$ dspmq -x -m QMMI1
QMNAME(QMMI1)   STATUS(Running)
    INSTANCE(host2) MODE(Active)

File Locking:
lsof /mqha/qmgrs/QMMI1/master and lsof /mqha/qmgrs/QMMI1/active
  See Appendix 2, section "++ Looking at the locks for the Active instance"
lsof /mqha/qmgrs/QMMI1/standby
  No locks.

+++ Scenario 11: endmqm -is to try to stop Standby in host2, and Active running host1. FAILS.

Note: This is similar to Scenario #8, but using 2 flags: -is

Prerequisite conditions:
host1: Active instance running. Started with -x (Scenario #3)
host2: Standby instance running. Started with -x (Scenario #5)

Result:
host1: Active continues to run.
host2: Standby continues to run. It was not stopped.
RC 90 Standby instance of IBM MQ queue manager QmgrName not ended

Solution:
If you want to stop the Standby in host2, see Scenario #14.
Use: endmqm -x QmgrName

+ host1:

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running)
   INSTANCE(host1) MODE(Active)
   INSTANCE(host2) MODE(Standby)
```

+ host2:

End command:
$ endmqm -is QMMI1

Output from command:
AMQ8368E: Standby instance of IBM MQ queue manager 'QMMI1' not ended.

Return code:
$ echo $?
90

The RC 90 means:
Standby instance of IBM MQ queue manager QmgrName not ended

Output from dspmq:

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running as standby)
   INSTANCE(host1) MODE(Active)
   INSTANCE(host2) MODE(Standby)
```

Entries from qmstatus.ini: (no change)
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
  CurrentStatus=Running
  PermitStandby=Yes
  PermitFailover=Yes

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
No new entries.

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No change in file locks.

+++ Scenario 12: endmqm -x to try to stop Active in host1. No Standby in host2. FAILS.

Note: This is similar to Scenario #13, which has a Standby running in host2.
The difference is that in Scenario #12, there is no Standby in host2.

Prerequisite conditions:
host1: Active instance running. Started with -x (Scenario #3)
host2: No Standby running.

Result:
host1: Active not stopped and continues to run.
host2: No Standby in host2.

Solution:
If you want to stop the Active, see Scenario #6.
Do not specify -x, use only -i.
Use in host1: endmqm -i QmgrName

+ host1:

Displaying status:
```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running)
    INSTANCE(host1) MODE(Active)
```

End command:
```
$ endmqm -x QMMI1
```

Output from command:
```
AMQ8367E: Active instance of IBM MQ queue manager 'QMMI1' not ended.
```

Return code:
```
$ echo $?
79
```

The RC 79 means:
Active instance of IBM MQ queue manager QmgrName not ended

Output from dspmq:
```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running)
    INSTANCE(host1) MODE(Active)
```

Entries from qmstatus.ini: (no change)
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
  CurrentStatus=Running
  PermitStandby=Yes
  PermitFailover=Yes

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
No new entries

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No change in file locks

+ host2:

$ dspmq -x -m QMMI1
QMNAME(QMMI1)   STATUS(Running elsewhere)
   INSTANCE(host1) MODE(Active)

+++ Scenario 13: endmqm -x to try to stop Active in host1. Standby running in host2. FAILS.

Note: This is similar to Scenario #12, which does not have a Standby in host2.
The difference is that in Scenario #13, there is a Standby running in host2.

Prerequisite conditions:
host1: Active instance running. Started with -x (Scenario #3)
host2: Standby instance running. Started with -x (Scenario #5)

Result:
host1: Active not stopped and continues to run.
host2: Standby continues to run.

Solution:

1) If you want to stop both the Active and the Standby, see Scenario #7.
Do not specify -x, use only -i.
Use in host1: endmqm -i QmgrName

2) If you want to switchover the Active from host1 to host2, see Scenario #10
Do not specify -x, use -is.
Use in host1: endmqm -is QmgrName

+ host1:

Displaying status:
```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running)
   INSTANCE(host1) MODE(Active)
   INSTANCE(host2) MODE(Standby)
```

End command:
```
$ endmqm -x QMMI1
```

Output from command:
AMQ8367E: Active instance of IBM MQ queue manager 'QMMI1' not ended.

Return code:
```
$ echo $?
79
```

The RC 79 means: Active instance of IBM MQ queue manager QmgrName not ended

Output from dspmq:
$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running)
INSTANCE(host1) MODE(Active)
INSTANCE(host2) MODE(Standby)

Entries from qmstatus.ini: (no change)
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
CurrentStatus=Running
PermitStandby=Yes
PermitFailover=Yes

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG
No new entries

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No change in file locks

+ host2:

$ dspmq -x -m QMMI1
QMNAME(QMMI1) STATUS(Running as standby)
INSTANCE(host1) MODE(Active)
INSTANCE(host2) MODE(Standby)

+++ Scenario 14: endmqm -x to end Standby, but Active continues to run. OK.

Note: This is the way to end a STANDBY instance, without disturbing the Active. The Active continues to run.

Prerequisite conditions:
host1: Active instance running. Started with -x (Scenario #3)
host2: Standby instance running. Started with -x (Scenario #5)

Result:
host1: Active continues to run.
host2: Standby stopped.

+ host1:

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running)
    INSTANCE(host1) MODE(Active)
    INSTANCE(host2) MODE(Standby)
```

+ host2:

End command:
```
$ endmqm -x QMMI1
```

Output from command:
```
Quiesce request accepted. The queue manager will stop when all outstanding work
is complete.
```

Return code:
```
$ echo $?
0
```

Output from dspmq:
```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                 STATUS(Running elsewhere)
    INSTANCE(host1) MODE(Active)
```

Entries from qmstatus.ini: (no change)
```
$ head /mqha/qmgrs/QMMI1/qmstatus.ini
QueueManagerStatus:
   CurrentStatus=Running
   PermitStandby=Yes
   PermitFailover=Yes
```

Entries from Error log: /mqha/qmgrs/QMMI1/errors/AMQERR01.LOG

03/15/2023 02:06:05 PM - Process(159125.1) User(mqm) Program(amqzxma0)
Host(host2) Installation(Installation1)
VRMF(9.3.2.0) QMgr(QMMI1)
Time(2023-03-15T21:06:05.339Z)
CommentInsert3(QMMI1)
AMQ8004I: IBM MQ queue manager 'QMMI1' ended.
EXPLANATION:
IBM MQ queue manager 'QMMI1' ended.

Generated event:
No new events.

Any new FDC in /var/mqm/errors?
No FDCs were generated.

File Locking:
No file locks for Standby.

+++ Scenario 15: host1 was upgraded to 9.3.2 but host2 remained at 9.2.0.6. FAILS.

You are using MQ 9.2 in both host1 and host2.
You upgrade host1 to MQ 9.3 according to the recommended sequence from:
https://www.ibm.com/support/pages/node/1284952
Multi-Stage migration of MQ multi-instance queue managers

… but you were interrupted with the process and forgot to do the upgrade in host2.
Thus, you have:
Host1: MQ 9.3.2.0
Host2: MQ 9.2.0.6

The active was started successfully in host1 (MQ 9.3.2).

```
mqm@host1: /home/mqm
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                   STATUS(Running)
    INSTANCE(host1) MODE(Active)
```

The file:
/mqha/qmgrs/QMMI1/qmstatus.ini

… shows that the minimum required version of MQ is: 9.3.2.0

```
QueueManagerStatus:
   CurrentStatus=Running
   PermitStandby=Yes
   PermitFailover=Yes
   PlatformSignature=20497
   PlatformString=Linux 4.18.0-425.13.1.el8_7.x86_64
   Version=9.3.2.0
   MinimumRequiredVersion=9.3.2.0
   UpdateTime=2023-03-15 11:44:23.332723
```

But host2 is still at MQ 9.2.0.6 and you try to start the queue manager and it will not work, because the old level of the code (MQ 9.2) CANNOT run a queue manager that recently was running under a newer release (MQ 9.3).

You will get an entry in the error log of the queue manager:

```
03/15/2023 04:12:54 AM - Process(113747.1) User(mqm) Program(amqzxma0)
                    Host(host2) Installation(Installation1)
                    VRMF(9.2.0.6) QMgr(QMMI1)
                    Time(2023-03-15T11:12:54.352Z)
                    ArithInsert1(932)
                    CommentInsert1(QMMI1)

AMQ7204E: IBM MQ queue manager 'QMMI1' cannot be started by this installation.
It has previously been started by a newer release of IBM MQ.

EXPLANATION:
The queue manager has previously been started by a newer release of IBM MQ at
command level 932. This installation is not compatible with the newer release's
data. Migration between these releases is not possible.
ACTION:
If the queue manager's data is shared using networked storage, ensure that all
installations used to start the queue manager are of the same release. The
queue manager can be started by installing a release of IBM MQ which supports
command level 932 or higher.
```

+++ Appendix 1: Entries in qmstatus.ini regarding Multi-Instance

The file /var/mqm/qmgrs/QMGR/qmstatus.ini contains several values related to multi-instance:

PermitStandby = Yes | No
  Indicates whether the active instance was started permitting standby instances (that is, strmqm -x)
 This is checked when the execution controller wants to become a standby instance

PermitFailover = Yes | No
  Indicates whether a standby instance is permitted to failover when active crashes
  This is used to prevent a queue manager which crashes as it starts up from doing it again

PlatformSignature = <numeric>
  Indicates which platform owns the data
  Prevents failover between different architectures and OSes

PlatformString = <string>
  A string version of the platform signature used when reporting a mismatch between the running code and the qmstatus.ini

+++ Appendix 2: Highlights of file locking when using multi-instance

The following 3 files in the queue manager directory structure are used to coordinate the starting of standby and the failover from the active to the standby:
DataPath_QMgr/active
DataPath_QMgr/master
DataPath_QMgr/standby

Question: Is there a way to find out if the "active" file is locked without taking the MQ trace?

The full path of the file active file for the queue manager in this document is:
/mqha/qmgrs/QMMI1/active

The Unix tool "lsof" can be used to show the file locks.
.
lsof is a tool that shows only the LOCAL processes that have locked a file.
In other words, if you login to the host that has the NFS server, lsof will NOT show those files that have been locked in remote hosts where the NFS file system has been mounted.
You will need to visit each host to find out the locked files.

Login as root in order to execute the operating system "lsof" to find out which applications have a lock on the files
(if you can execute successfully "lsof" as user "mqm" or an MQ administrator, then it is OK to proceed with the user "mqm").

In this scenario, the Active instance is running in host1.

The meaning of the 4th column (FD) is the one that is of interest for us regarding file locking:
If the first letter is the lower case "r" means: r for read access;
If the first letter is the lower case "u" means: u for read and write access;
If the last character is the upper case "W" means: W for a write lock on the entire file;
If the last character is the upper case"R" means: R for a read lock on the entire file;
If the last character is a space, then there is no lock.
.
Thus:
uW means - read and write access, write lock on entire file
rW means - read access, write lock on entire file
rR means - read access, read lock on entire file
r means - read access, no lock

++ Looking at the locks for the Active instance

Let's list the locks for the files used in multi-instance in host1:

In host1 where the Active instance is running, issue the following to list the locks for the 'master' file.
Under the column FD notice the value of uW:
uW => which means that the file is opened with read and write access, and with **write lock** on entire file

```
$ lsof /mqha/qmgrs/QMMI1/master
COMMAND    PID USER   FD   TYPE DEVICE SIZE/OFF    NODE NAME
amqzxma0 28562  mqm   12uW  REG   0,21       52 3504413 /mqha/qmgrs/QMMI1/master
```

List the locks for the 'active' file.
Under the column FD notice the value of rR:
rR => which means that the file is opened with read access mode, and with **read lock** on entire file

```
$ lsof /mqha/qmgrs/QMMI1/active
COMMAND    PID USER   FD   TYPE DEVICE SIZE/OFF    NODE NAME
amqzxma0 28562  mqm   13rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqzfuma 28567  mqm    4rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqzmuc0 28575  mqm    4rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqzmuc0 28575  mqm    5rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqzmur0 28580  mqm    4rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqzmuf0 28595  mqm    4rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqzmuf0 28595  mqm    5rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqrrmfa 28599  mqm    4rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqrrmfa 28599  mqm    5rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqfqpub 28612  mqm    4rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqfcxba 28632  mqm    4rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
amqzlaa0 28645  mqm    4rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
```

Notice that the Execution Controller (EC) for the queue manager is called "amqzxma0":

```
amqzxma0 28562  mqm   13rR  REG   0,21       52 3504414 /mqha/qmgrs/QMMI1/active
```

You can use the following to find out the command and arguments for the EC:

```
$ ps -ef | grep 28562
mqm      28562     1  0 04:00 ?       00:00:00 /opt/mqm/bin/amqzxma0 -m QMMI1 -x -u mqm
```

List the locks for the 'standby' file.
Under the column FD notice the value of r:
r => which means that the process has opened the file in read access mode, but there is NO lock on the file

```
$ lsof /mqha/qmgrs/QMMI1/standby
COMMAND    PID USER   FD   TYPE DEVICE SIZE/OFF    NODE NAME
amqzxma0 28562  mqm    4r   REG   0,21       54 3504509 /mqha/qmgrs/QMMI1/standby
```

The MQ processes for the queue manager are:

```
mqm@host1: /mqha/qmgrs/QMMI1/errors
$ ps -ef | grep mq
mqm      28562     1  0 04:00 ?       00:00:00 /opt/mqm/bin/amqzxma0 -m QMMI1 -x -u mqm
mqm      28567 28562  0 04:00 ?       00:00:00 /opt/mqm/bin/amqzfuma -m QMMI1
mqm      28575 28562  0 04:00 ?       00:00:00 /opt/mqm/bin/amqzmuc0 -m QMMI1
mqm      28580 28562  0 04:00 ?       00:00:00 /opt/mqm/bin/amqzmur0 -m QMMI1
mqm      28595 28562  0 04:00 ?       00:00:00 /opt/mqm/bin/amqzmuf0 -m QMMI1
mqm      28599 28562  0 04:00 ?       00:00:00 /opt/mqm/bin/amqrrmfa -m QMMI1 -
t2332800 -s2592000 -p2592000 -g5184000 -c3600
mqm      28608 28562  0 04:00 ?       00:00:00 /opt/mqm/bin/amqzmgr0 -m QMMI1
mqm      28612 28595  0 04:00 ?       00:00:00 /opt/mqm/bin/amqfqpub -mQMMI1
mqm      28627 28608  0 04:00 ?       00:00:00 /opt/mqm/bin/runmqchi -m QMMI1 -q
SYSTEM.CHANNEL.INITQ -r
mqm      28629 28608  0 04:00 ?       00:00:00 /opt/mqm/bin/amqpcsea QMMI1
mqm      28632 28612  0 04:00 ?       00:00:00 /opt/mqm/bin/amqfcxba -m QMMI1
mqm      28636 28608  0 04:00 ?       00:00:00 /opt/mqm/bin/runmqlsr -r -m QMMI1 -t TCP -
p 1421
mqm      28645 28562  0 04:00 ?       00:00:00 /opt/mqm/bin/amqzlaa0 -mQMMI1 -fip0
```

++ Looking at the locks for the Standby instance

Let's list the locks for the files used in multi-instance in host2:

In host2 where the Standby instance is running, issue the following to list the locks for the 'master' file:

```
$ lsof /mqha/qmgrs/QMMI1/master
(no output - which means, that there are no locks)
```

List the locks for the 'active' file:

```
$ lsof /mqha/qmgrs/QMMI1/active
(no output - which means, that there are no locks)
```

List the locks for the 'standby' file:
Under the column FD notice the value of r:
  r => which means that the process has opened the file in read access mode, but there is NO lock on the file
The value for FD includes rW and uW:
  rW => which means that the file is open with read access, and with **write lock** on entire file
  uW => which means that the file is open with read and write access, with **write lock** on entire file

```
$ lsof /mqha/qmgrs/QMMI1/standby
COMMAND   PID USER   FD   TYPE DEVICE SIZE/OFF    NODE NAME
amqzxma0 1287  mqm    4rW  REG   0,21       54 3504509 /mqha/qmgrs/QMMI1/standby
amqzxma0 1287  mqm    5uW  REG   0,21       54 3504509 /mqha/qmgrs/QMMI1/standby
```

The MQ processes for the queue manager are:

```
0$ ps -ef | grep mq | grep QMMI1
mqm       1287     1  0 03:55 ?        00:00:00 /opt/mqm/bin/amqzxma0 -m QMMI1 -x -u mqm
mqm       1295  1287  0 03:55 ?        00:00:00 /opt/mqm/bin/amqzfuma -m QMMI1
```

+++ Appendix 3: Return codes for strmqm and endmqm

https://www.ibm.com/docs/en/ibm-mq/9.3?topic=reference-strmqm-start-queue-manager
IBM MQ > 9.3 > IBM MQ > Reference > Administration reference > IBM MQ Control commands > The control commands >
strmqm (start queue manager)

Table at the bottom of the web page:

Return codes

Return code	Description
0	Queue manager started.
1	The location chosen for the queue manager data directory is invalid
3	Queue manager being created.
5	Queue manager running.
16	Queue manager does not exist.
23	Log not available.
24	A process that was using the previous instance of the queue manager has not yet disconnected.
30	A standby instance of the queue manager started. The active instance is running elsewhere.
31	The queue manager already has an active instance. The queue manager permits standby instances.
39	Invalid parameter specified.
43	The queue manager already has an active instance. The queue manager does not permit standby insta
47	The queue manager already has the maximum number of standby instances.
49	Queue manager stopping.
58	Inconsistent use of installations detected.
62	The queue manager is associated with a different installation.
69	Storage not available.
71	Unexpected error.
72	Queue manager name error.
74	The IBM MQ service is not started.
91	The command level is outside the range of acceptable values.
92	The queue manager's command level is greater or equal to the specified value.
94	A replica instance of the queue manager has been started.
100	Log location invalid.
114	Invalid `qm.ini` file stanza.
119	User not authorized to start the queue manager.

Table 1. Return code identifiers and descriptions

https://www.ibm.com/docs/en/ibm-mq/9.3?topic=reference-endmqm-end-queue-manager
IBM MQ > IBM MQ 9.3 > IBM MQ > Reference > Administration reference > IBM MQ Control commands > The control commands >
endmqm (end queue manager)

Table at the bottom of the web page:

Return codes

Return code	Description
0	Queue manager ended
3	Queue manager being created
V 9.3.2 4	V 9.3.2 Queue Manager is being started
16	Queue manager does not exist
39	Invalid parameter specified
40	Queue manager not available
49	Queue manager stopping
58	Inconsistent use of installations detected
62	The queue manager is associated with a different installation
69	Storage not available
71	Unexpected error
77	IBM MQ queue manager cannot switch over
79	Active instance of IBM MQ queue manager *QmgrName* not ended
90	Standby instance of IBM MQ queue manager *QmgrName* not ended
119	Permission denied

Table 1. Return code identifiers and descriptions

+++ Note: 8.0.0.3 extended start events

Besides 8.0.0.3 " Extended start events for multi-instance queue managers ", no new features or improvements on functionality have been done in MQ 9.0 or later.

https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_8.0.0/com.ibm.mq.pro.doc/q123800_.htm
IBM MQ > IBM MQ 8.0.0 > IBM MQ > Product overview > What's changed in IBM MQ Version 8.0 > Extended start events for multi-instance queue managers

Extended start events for multi-instance queue managers
From IBM MQ Version 8.0.0, Fix Pack 3, start events for multi-instance queue managers are extended to allow system monitoring applications to see when a multi-instance queue manager has failed over and where it is now running.
.
Reason code in MQCFH:
MQRC_Q_MGR_ACTIVE (2222, X'8AE').
Queue manager active.
Event description: This condition is detected when a queue manager becomes active.
Event type: Start And Stop.
Values:
MQRQ_FAILOVER_PERMITTED
Queue manager has started normally and allows a standby instance.
.
MQRQ_FAILOVER_NOT_PERMITTED
Queue manager has started normally but does not allow a standby instance.
.
MQRQ_STANDBY_ACTIVATED
Queue manager has moved out of standby mode into active mode.

+++ end