

## Different ways to get the Current Depth (CURDEPTH) for a queue in an IBM MQ queue manager

<https://www.ibm.com/support/pages/node/6209225>

Date last updated: 20-Jun-2024

Angel Rivera - [rivera@us.ibm.com](mailto:rivera@us.ibm.com)  
IBM MQ Support

### +++ Objective

You would like to know the different ways to obtain the Current Depth (CURDEPTH) for a queue in an MQ queue manager. That is, you want to know how many messages are available in the queue.

The purpose of the article is to explain several ways that can be used to get the Current Depth (CURDEPTH) for a queue:

- a) Traditional usage of 'echo', 'runmqsc', 'grep' (local and remote)
- b) MQ Explorer
- c) Java PCF
- d) Administrative MQ REST API for queue and mqsc (requires MQ Web Server 9.1 or higher)
- e) MQ Web Console
- f) Queue statistics records: viewed by sample amqsmon

In the examples examined in this article, there are 3 messages in the local queue Q1 in the queue manager QM92.

++ About the "WHERE" clause in the DISPLAY command

One refinement to queries that could be helpful to you, is the use of the WHERE clause in the DISPLAY command.

The following runmqsc command shows all the local queues that have messages, where the integer attribute for Current Depth is greater than 0 (zero).

```
DISPLAY QLOCAL(*) WHERE(CURDEPTH GT 0)
```

For more details on the WHERE clause see:

[https://www.ibm.com/docs/en/ibm-mq/9.2?topic=reference-display-queue-display-queue-attributes#q086270\\_2](https://www.ibm.com/docs/en/ibm-mq/9.2?topic=reference-display-queue-display-queue-attributes#q086270_2)

IBM MQ / 9.2 /

DISPLAY QUEUE (display queue attributes)

Use the MQSC command DISPLAY QUEUE to display the attributes of one or more queues of any type.

Parameter descriptions for DISPLAY QUEUE

WHERE => Specify a filter condition to display only those queues that satisfy the selection criterion of the filter condition. The filter condition is in three parts: filter-keyword, operator, and filter-value

+++ a) Traditional usage of 'echo', 'runmqsc', 'grep' (local and remote)

a.1) Local

For more details see the article:

<https://www.ibm.com/support/pages/node/394113>

How to obtain the value for an attribute from the output of runmqsc in MQ, such as CURDEPTH for a Queue

Here is a summary:

Unix: display the attribute name and the value

```
$ echo "DISPLAY QL(Q1) CURDEPTH" | runmqsc QM92 | grep 'CURDEPTH('
CURDEPTH(3)
```

Unix: display only the value

```
$ echo "DISPLAY QL(Q1) CURDEPTH" | runmqsc QM92 | grep 'CURDEPTH(' | sed
's/.*CURDEPTH//' | tr -d '()'
3
```

Windows:

```
C:\> echo DISPLAY QL(Q1) CURDEPTH | runmqsc QM92 | findstr "CURDEPTH("
CURDEPTH(3)
```

a.2) Remote (and non Administrator user)

Example:

Queue manager QM92 is in host "orizaba" and in that host, there is a user called 'bob' who belongs to the group "mqusers" but it is NOT an MQ Administrator  
From the remote host "fortin", the user 'bob' sets the MQSERVER variable for the queue manager QM92 in host 'orizaba'.

From host 'orizaba', it is necessary that an MQ Administrator authorizes the user 'bob' to do remote access and to use runmqsc:

```
setmqaut -m QM92 -t qmgr -g mqusers +connect +inq +dsp
setmqaut -m QM92 -t q -n SYSTEM.DEFAULT.MODEL.QUEUE -g mqusers +inq +browse +get +dsp
setmqaut -m QM92 -t q -n SYSTEM.ADMIN.COMMAND.QUEUE -g mqusers +inq +put +dsp
setmqaut -m QM92 -t q -n SYSTEM.MQEXPLORER.REPLY.MODEL -g mqusers +inq +browse +get +dsp
+put
setmqaut -m QM92 -t q -n MY.Q -g mqusers +inq +browse +get +put +dsp
```

The following is needed when using runmqsc:

```
setmqaut -m QM92 -t q -n SYSTEM.MQSC.REPLY.QUEUE -g mqusers +inq +browse +get +dsp
```

For more details see the article:

<https://www.ibm.com/support/pages/node/616403>

New features added to the runmqsc command in IBM MQ Version 8.0

See Page 7:

++ Providing remote access to non MQ Administrators

To provide access to a non MQ administrator to the queue manager via command line (runmqsc)

or the MQ Explorer issue the 4 setmqaut commands shown below.

.

See Page 8:

ADDITIONAL authorities are needed:

The above 4 commands, just allow for the connection to happen.

Once the user is connected, then it is necessary to provide additional authorities in order to do useful work, such as putting messages into a queue.

The following command gives "inquiry" and "display" authority for queue Q1.

```
setmqaut -m MYQMGR -t q -n Q1 -p myuser +inq +dsp
```

.

See Page 10:

Chapter 3: Non-administrators (not in the group 'mqm') can now use runmqsc

.

See Pages 12-15

+ Scenario 2: user can connect, needs to see current depth in queue Q1

+ Scenario 4: Detailed scenario of remote user who is not an MQ administrator

In addition, for more details on dedicated server-connection channels, see:

<https://www.ibm.com/support/pages/node/1135522>

Configuring MQ to use a dedicated Listener, Channel and Queue in Linux

Example:

```
bob@fortin1.fyre.ibm.com: /home/bob
```

```
$ export MQSERVER=MY.CHANNEL/TCP/orizaba1.fyre.ibm.com(1414)
```

```
$ runmqsc -c -u bob QM92
```

```
5724-H72 (C) Copyright IBM Corp. 1994, 2020.
```

```
Enter password:
```

```
*****
```

```
Starting MQSC for queue manager QM92.
```

```
display ql(Q1) curdepth
```

```
1 : display ql(Q1) curdepth
```

```
AMQ8409I: Display Queue details.
```

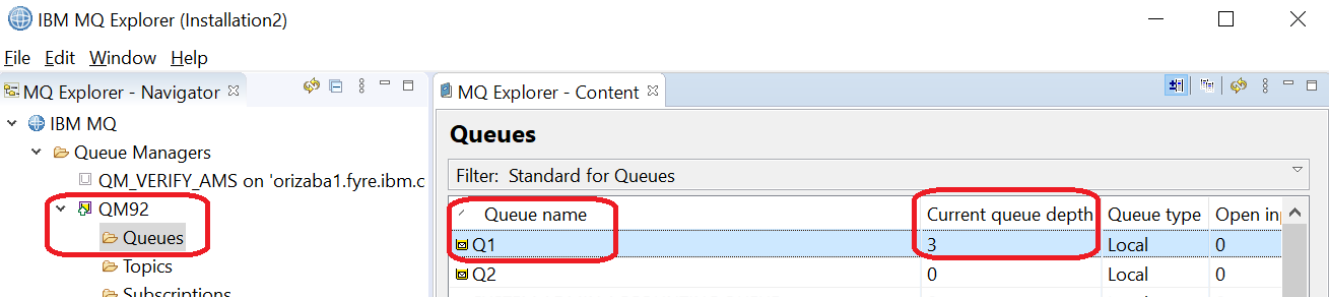
```
QUEUE(Q1)                                TYPE(QLOCAL)
```

```
CURDEPTH(3)
```

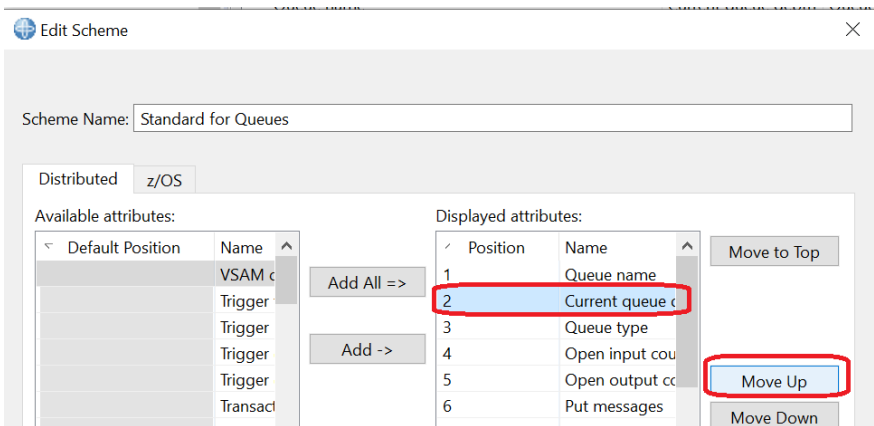
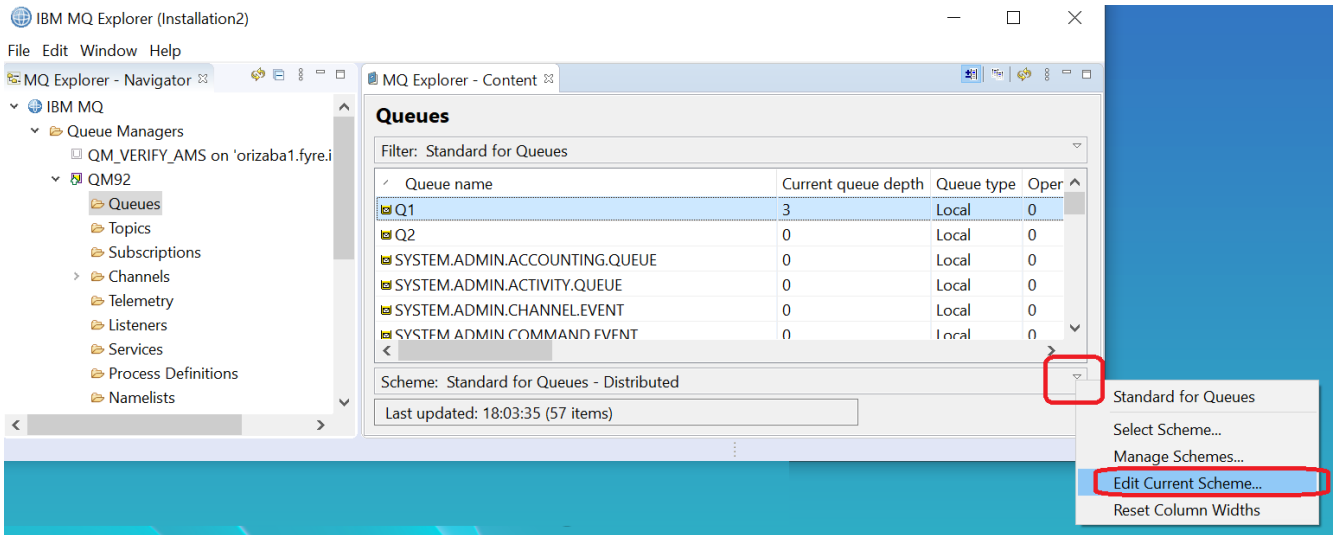
+++ b) MQ Explorer

The column named “Current queue depth” is not the 2<sup>nd</sup> column, instead it is the 5<sup>th</sup> column (from left to right).

This article shows how the 2<sup>nd</sup> column shows the “Current queue depth” for a queue.



You can customize the order of the columns by editing the “Scheme”



### +++ c) Java PCF

Note:

The PCF samples provided with MQ do NOT include the obvious sample of inquiring for the CURDEPTH of a queue.

However, there are 2 public samples from Roger Lacroix(CapitalWare)

[https://www.capitalware.com/rl\\_blog/?p=5992](https://www.capitalware.com/rl_blog/?p=5992)

Simple Monitoring of the Current Queue Depth

June 4, 2020 Roger Lacroix

run a Java/MQ/PCF application to get both the current and maximum depths and compare the values. Here is a simple Java/MQ/PCF program that will request all queues with a queue depth greater than zero and then compare the current depth to the maximum depth and output a message if needed.

You can download the source code from:

<http://www.capitalware.com/dl/code/java/mqcurrentdepthmonitor01.zip>

[https://www.capitalware.com/rl\\_blog/?p=5472](https://www.capitalware.com/rl_blog/?p=5472)

Java MQ Code to List All Local Queues Filtering by Current Depth

May 30, 2019 Roger Lacroix

If you have done the following runmqsc command to display all local queues filtering by current queue depth of a queue manager:

```
DIS QL(*) WHERE (CURDEPTH GT 0)
```

... And you wanted to do the same thing via a program, here is a fully functioning Java MQ example that will connect to a remote queue manager, issue a PCF "Inquire Queue" command filtering the current queue depth (CURDEPTH), get the PCF response messages, loop through the PCF responses and output the information. You can download the source code from:

<https://www.capitalware.com/dl/code/java/mqlistqueuewithfilter01.zip>

+++ d) Administrative MQ REST API for queue and mqsc (requires MQ Web Server 9.1 or higher)

See the following article and search for "CURDEPTH" and "currentDepth" for actual examples on different ways to get the Current Depth (or number of messages) in a Queue.

<https://www.ibm.com/support/pages/node/6208006>

Using the MQ 9.1.5 CD REST API in Linux with no security (for Testing environments)

Here is the summary:

Page 26

+ Show the status for a queue

```
$ curl -s -k
```

```
"https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QMORI/queue/Q1?status=*" -X GET
```

```
{"queue": [{"name": "Q1", "status": {"currentDepth": 1, "lastGet": "", "lastPut": "", "mediaRecoveryLogExtent": "", "monitoringRate": "off", "oldestMessageAge": -1, "onQueueTime": {"longSamplePeriod": -1, "shortSamplePeriod": -1}, "openInputCount": 0, "openOutputCount": 0, "uncommittedMessages": 0}, "type": "local"}]}
```

+ Get the value for currentDepth

(You can use the above output to help you refine the argument for jq)

```
$ curl -s -k
```

```
"https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QMORI/queue/Q1?status=*" -X GET |
```

```
jq '.queue | .[] | .status.currentDepth'
```

```
1
```



Pages 36 and 37:

+ Display the CUDEPTH for Q5

```
$ curl -s -k
"https://orizaba1.fyre.ibm.com:9443/ibmmq/rest/v1/admin/qmgr/QMFOR910/queue/Q5
?status=status.currentDepth" -X GET
{"queue": [{
  "name": "Q5",
  "status": {"currentDepth": 0},
  "type": "local"
}]}
```

```
$ curl -s -k
"https://orizaba1.fyre.ibm.com:9443/ibmmq/rest/v2/admin/action/qmgr/QMORI/mqsc" -
X POST -H "Content-Type: application/json" --data '{"type":"runCommand",
"parameters": {"command": "DISPLAY QLOCAL(Q5) CURDEPTH" } }'
{
  "commandResponse": [{
    "completionCode": 0,
    "reasonCode": 0,
    "text": ["AMQ8409I: Display Queue details. QUEUE(Q5)
TYPE(QLOCAL) CURDEPTH(0) "]
  }],
  "overallCompletionCode": 0,
  "overallReasonCode": 0
}
```

Page 39:

+ Get the CURDEPTH for the queue Q1

```
$ curl -s -k
"https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QMORI/queue/Q1?status=*" -X GET |
jq '.queue | .[] | .status.currentDepth'
1
```

Page 40

```
$ curl -s -k
"https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QMORI/queue/Q1?status=*" -X GET |
jq '.queue | .[] | .status.currentDepth'
1
```

e) MQ Web Console

For more information on how to setup the MQ Web Server and for basic usage of the MQ Web Console, see:

The following tutorial includes also the installation and setup for the REST API:

<https://www.ibm.com/support/pages/node/6118000>

Configuring MQ 9.1 Web Server in Linux and in Windows with No Security (for Testing the MQ Web Console)

Use a web browser and visit the MQ Web Console.

On the left panel, select:  
Manage

And in the right panel, select:  
Local queue managers

Then select the desired queue manager.  
In this example is: QM93WIN

The screenshot shows the MQ Web Console interface. On the left, a dark sidebar contains navigation options: Home, Manage (highlighted with a red box), Learning resources, and Settings. The main content area is titled 'Manage' (highlighted with a red box) and 'Queue managers'. Under 'Queue managers', there are two sub-sections: 'Remote queue managers' and 'Local queue managers' (highlighted with a red box). The 'Local queue managers' section contains a table with the following data:

| Unique name ↑ | Status  |
|---------------|---------|
| QM93          | Stopped |
| QM931         | Stopped |
| QM93WIN       | Running |

The 'QM93WIN' row is highlighted with a red box.

Click on the desired queue manager and you will see another page.

In this example, one of the queues (Q1) has reached its Maximum Queue Depth, and you will see an indicator that 1 queue is full.

The screenshot shows the IBM MQ console dashboard. On the left is a navigation menu with 'Home', 'Manage', 'Learning resources', and 'Settings'. The main area contains three summary cards:
 

- Queue depth full:** 1 (highlighted with a red box and star), 1 user-defined, 0 system queues.
- Queue depth warning:** 0, 0 user-defined, 0 system queues.
- Queue depth normal:** 57, 10 user-defined, 47 system queues.

In the bottom part of the right panel, you will see more details on the queues.

In particular, you will see an icon (red circle) for Q1 to indicate that it is full. It has 5 messages and the maximum depth is 5 (of course, this is an extreme low number, in order to illustrate the feature).

| Name                             | Type  | Depth % | Maximum depth | Current connections | Attached apps     | Last put                   | Last get      |
|----------------------------------|-------|---------|---------------|---------------------|-------------------|----------------------------|---------------|
| Q1                               | Local | 100%    | 5/5           | Input 0, Output 0   | -                 | Mar 20, 2023 at 2:45:09 PM | -             |
| Q2                               | Local | 40%     | 2/5           | Input 0, Output 0   | -                 | 5 hours ago                | -             |
| AMQ.6405797A203E9401             | Local | 0%      | 0/5000        | Input 0, Output 0   | -                 | -                          | -             |
| AMQ.MQEXPLORE R.641738F521DA8D02 | Local | 0%      | 0/5000        | Input 1, Output 0   | MQ Explorer 9.3.2 | 1 seconds ago              | 1 seconds ago |

f) Queue statistics records: viewed by sample amqsmon

For more details see:

Gathering and Displaying Statistics in WebSphere MQ V7

<https://www.ibm.com/support/pages/node/590415>

This WebSphere Support Technical Exchange is designed to present how to gather and display statistics in WebSphere MQ V7, by using the sample amqsmon.

```
C:\> amqsmon -m QM93WIN -t statistics -q Q1 -b
```

```
MonitoringType: QueueStatistics
QueueManager: 'QM93WIN'
IntervalStartDate: '2023-03-20'
IntervalStartTime: '14.35.19'
IntervalEndDate: '2023-03-20'
IntervalEndTime: '15.05.19'
CommandLevel: 932
ObjectCount: 11
QueueStatistics: 0
  QueueName: 'Q1'
  CreateDate: '2022-07-22'
  CreateTime: '09.09.39'
  QueueType: Local
  QueueDefinitionType: Predefined
  QMinDepth: 3
  QMaxDepth: 5
  AverageQueueTime: [0, 0]
  PutCount: [2, 0]
  PutFailCount: 0
  Put1Count: [0, 0]
  Put1FailCount: 0
  PutBytes: [6, 0]
  GetCount: [0, 0]
  GetBytes: [0, 0]
  GetFailCount: 0
  BrowseCount: [0, 0]
  BrowseBytes: [0, 0]
  BrowseFailCount: 0
  NonQueuedMsgCount: 0
  ExpiredMsgCount: 0
  PurgeCount: 0
```

```
+++ end +++
```