Have you ever wanted to use dspmq on z/OS?

gwydiontudur |Dec 14 2016 Updated

If you’ve used MQ on distributed platforms, you’re likely to be familiar with the dspmq utility. This utility displays the names and details of the queue managers on the system. There hasn’t been an equivalent utility on z/OS, until now.

One of the new features in MQ V9.0.1 is the addition of the CSQUDSPM utility, which provides equivalent function to dspmq on other platforms.

**What does it do?**

The CSQUDSPM utility lists information about all queue managers on the LPAR where it is running, regardless of the version of MQ the queue manager is associated with.

There are parameters that can be specified to request information for a specific queue manager, or a subset of queue managers, and to control the amount of information that is displayed for each queue manager. These parameters are documented in the [IBM MQ V9.0 Knowledge Center](http://www.ibm.com/support/knowledgecenter/SSFKSJ_9.0.0/com.ibm.mq.ref.adm.doc/q083185_.htm).

The information displayed for each queue manager is similar to what is produced by the dspmq utility on distributed platforms, with some differences as some fields are not relevant to all platforms. Below is an example of the output you can expect from the CSQUDSPM utility on z/OS.

You’ll notice in the example above that the version for some of the queue managers is displayed as “0.0.0”. This is because those queue managers haven’t been started since the system was last IPLed. CSQUDSPM can only display the version of queue managers that have been started since the system was IPLed.

**How do I run it?**

There are two ways to run the CSQUDSPM utility. It can be run as a batch job. Sample JCL, CSQ4DSPM, is provided with MQ V9.0.1 in the SCSQPROC library to help you do this. The JCL needed to run the utility is very simple, and will be similar to this example.

There is an alias named DSPMQ in the SCSQAUTH library, where the CSQUDSPM utility is packaged. This means that you can run the utility with either the CSQUDSPM or DSPMQ command.

The other way to run the utility is in USS. To do this you’ll need a file in USS with the same name as the program (dspmq or csqudspm), with the sticky bit on. You can create this file by issuing the following commands:

touch dspmq

chmod 755 dsmq

chmod +t dspmq

If you’ve installed the new “Unix System Services Web Components” feature, which lets you use the [IBM MQ Console](https://www.ibm.com/developerworks/community/blogs/messaging/entry/IBM_MQ_Console_on_z_OS) and [administrative REST API](https://www.ibm.com/developerworks/community/blogs/messaging/entry/BIteSize_Blogging_MQ_9_0_1_Administrative_REST_API) in MQ V9.0.1, you’ll already have a file named dspmq with the correct permissions in the web/bin directory under your MQ installation path.

As long as you have the MQ V9.0.1 SCSQAUTH library in your STEPLIB concatenation, CSQUDSPM will be called when the file is executed in USS.

Remember that all the libraries in your STEPLIB concatenation need to be APF authorized. If not, dspmq will abend with code 047!