



Infrastructure Series

TechDoc
IBM MQ

IBM MQ Administration
z/OS Installation/Upgrade Verification





Table of Contents

Introduction	3
Document Version	3
Initial MQ Installation Verification Steps	4
Review Logs.....	4
Review MQ ISPF Panels.....	4
Verify the Functional Level of the Queue Manager.....	4
The CSQ4IVP1 Verification Program	5
Introduction	5
Preparation	5
IVP Test Execution.....	5
CSQ4IVP1 Sample Report.....	6
The CSQ4IVPX Verification Program	7
Introduction	7
Preparation	7
IVP Test Execution.....	8
CSQ4IVPX Sample Report.....	8
Verifying Queue Sharing Groups.....	10
Introduction	10
Preparation	10
IVP Test Execution.....	10
References	11



Introduction

Document Version

This document describes how to verify the installation or upgrade of IBM MQ software on the z/OS platform. The installations verified include:

- MQ
- MQ Managed File Transfer (MQ MFT)
- MQ Advanced Message Security (MQ AMS)

The contents of this document should be generally applicable to both current and previous versions of IBM Integration Bus. They have been specifically verified on the following production versions:

- IBM MQ v9.0.0.0

This documentation has been created and maintained by:

- Glen Brumbaugh

This documentation has been reviewed by:

- Glen Brumbaugh

This documentation was last updated:

- Version 1.0 July 2017



Initial MQ Installation Verification Steps

Review Logs

After z/OS MQ software has been installed or upgraded, the first verification step is to review the logs from the relevant Address Spaces to ensure that there are no unexpected error messages. The following steps are required:

- Review the “MSTR” Address Space logs
- Review the “CHIN” Address Space logs
- Review the “AMSM” Address Space logs (if using AMS)
- Review the MQ MFT Address Space logs (if using MFT)

Review MQ ISPF Panels

After the Address Space logs have been reviewed, the MQ TSO/ISPF panels should be reviewed to verify their functionality. The following steps are required:

- Verify the functionality of the MQ ISPF panels (CSQOREXX)

Verify the Functional Level of the Queue Manager

The functional level of the newly installed MQ software is controlled by the *OPMODE* parameter in the *CSQ6SYSP* macro. There are no new functions available in MQ v9.0.0 that can be enabled by this parameter, but there are new functions in MQ v8.x that are controlled by this parameter. The current functional level of the system can be determined by executing the following MQSC command:

```
DISPLAY SYSTEM
```

Ensure that the Queue Manager is functioning at the correct level. If necessary, modify and recompile the *CSQ6SYSP* macro using one of the options listed below.

- `OPMODE=(COMPAT,vrm)` “*vr*m” = Version Release Modification level (e.g. 710)
- `OPMODE=(NEWFUNC,vrm)` “*vr*m” = Version Release Modification level (e.g. 900)

Note that if “COMPAT” mode is chosen, additional verification may be required each time a “NEWFUNC” setting is used. Also note that once a “NEWFUNC” level has been selected, the Queue Manager may not be migrated back to an earlier level.



The CSQ4IVP1 Verification Program

Introduction

Once it has been ascertained that the MQ subsystem has no obvious errors, some more formal verification programs can be run. The first of these is the CSQ4IVP1 program. This is an assembler program that runs in batch mode to verify the MQ subsystem (“MSTR”). This program does not require any of the provided sample language programs (C, COBOL, etc.). The CSQ4IVP1 program is link-edited during the SMP/E process. The load modules are contained in the following dataset:

hlq.v9r0m0.SCSQLOAD

The CSQ4IVP1` program tests the following functions:

- Tests the Message Queue Interface (MQI) calls
- Tests the Command Server component
- Tests the Triggering component
- Tests dynamic Queue processing
- Tests message Expiry processing
- Tests message Commit processing

Preparation

To prepare to execute the CSQ4IVP1 program, take the following steps:

- Verify that the IVP entries are in the “CSQINP2” DD concatenation. The IVP entries are described in the following dataset:

hlq.v9r0m0.SCSQPROC(CSQ4IVPQ)

- Modify the JCL to execute the CSQ4IVP1 program. This JCL is found in the following dataset:

hlq.v9r0m0.SCSQPROC(CSQ4IVPR)

- Ensure that the User ID that the JCL will be submitted under has adequate authority, as defined in the ESM, to access the necessary MQ objects. The following access is required:

Access	Resource Profile	Resource Class
Read	<i>qmgr</i> .BATCH	MQCONN
Read	<i>qmgr</i> .DISPLAY.PROCESS	MQCMDS
Update	<i>qmgr</i> .SYSTEM.COMMAND.INPUT	MQQUEUE
Update	<i>qmgr</i> .SYSTEM.COMMAND.REPLY.MODEL	MQQUEUE
Update	<i>qmgr</i> .CSQ4IVP1.**	MQQUEUE

IVP Test Execution

Submit the JCL to execute the CSQ4IVP1 program. This verification is split into ten “stages”. Each stage must successfully complete (e.g. Return Code of zero) before the next stage can be run. These ten stages are:

- 1) Connect to the Queue Manager.
- 2) Verify Command Server functionality.



- 3) Verify Trigger Monitor functionality.
- 4) Create a Dynamic Queue.
- 5) Test cross-feature functionality.
- 6) Test cross-feature functionality.
- 7) Delete a Dynamic Queue.
- 8) Verify Message Expiry functionality.
- 9) Verify Commit/Rollback functionality.
- 10) Disconnect from the Queue Manager.

Trouble-shoot any "stage" with a non-zero return code.

CSQ4IVP1 Sample Report

```
DATE : 2005.035                      IBM MQ for z/OS - V6
PAGE : 0001
INSTALLATION VERIFICATION PROGRAM
PARAMETERS ACCEPTED. PROGRAM WILL CONNECT TO : CSQ1
,OBJECT QUALIFER : CSQ4IVP1
INSTALLATION VERIFICATION BEGINS :
STAGE 01 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR BRIDGE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS EXCP FOR CHANNEL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR SSL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR INHIBITED EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR LOCAL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR PERFORMANCE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR REMOTE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR START/STOP EVENTS
STAGE 02 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
SYSTEM.COMMAND.INPUT
STAGE 03 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 04 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
CSQ4IVP1.BAB9810EFEAC8980
STAGE 05 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 06 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 07 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 08 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
CSQ4IVP1.BAB9810F0070E645
STAGE 09 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
CSQ4IVP1.BAB9812BA8706803
STAGE 10 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
>>>>>>>>>  END OF REPORT  <<<<<<<<<<<
```



The CSQ4IVPX Verification Program

Introduction

After the CSQ4IVP1 verification program has been executed, a second verification program can be run. This is the CSQ4IVPX program. This is an assembler program that runs in batch mode to verify the MQ Channel Initiator subsystem (“CHIN”). As with the CSQ4IVP1 program, this program also does not require any of the provided sample language programs (C, COBOL, etc.). The CSQ4IVPX program is also link-edited during the SMP/E process. The load modules are contained in the following dataset:

hlq.v9r0m0.SCSQLOAD

The CSQ4IVPX` program tests the following functions:

- Starts the Channel Initiator
- Executes the “DISPLAY CHINIT” command

Preparation

To prepare to execute the CSQ4IVPX program, take the following steps:

- Modify the JCL to execute the CSQ4IVPX program. This JCL is found in the following dataset:

hlq.v9r0m0.SCSQPROC(CSQ4IVPX)

- Ensure that the User ID that the JCL will be submitted under has adequate authority, as defined in the ESM, to access the necessary MQ objects. The following access is required:

Access	Resource Profile	Resource Class
Read	<i>qmgr</i> .BATCH	MQCONN
Control	<i>qmgr</i> .START.CHINIT	MQCMDS
Control	<i>qmgr</i> .STOP.CHINIT	MQCMDS
Read	<i>qmgr</i> .DISPLAY.CHINIT	MQCMDS
Update	<i>qmgr</i> .SYSTEM.COMMAND.INPUT	MQQUEUE
Update	<i>qmgr</i> .SYSTEM.CSQUTIL.*	MQQUEUE

- Ensure that the Channel Initiator (“CHIN”) Address Space has adequate authority, as defined in the ESM, to access the necessary MQ objects. The following access is required:

Access	Resource Profile	Resource Class
Read	<i>qmgr</i> .CHIN	MQCONN
Control	<i>qmgr</i> .CONTEXT.**	MQADMIN
Update	<i>qmgr</i> .SYSTEM.COMMAND.INPUT	MQQUEUE
Update	<i>qmgr</i> .SYSTEM.CHANNEL.INITQ	MQQUEUE
Update	<i>qmgr</i> .SYSTEM.CHANNEL.SYNCQ	MQQUEUE



Access	Resource Profile	Resource Class
Update	<i>qmgr</i> .SYSTEM.CLUSTER.TRANSMIT.QUEUE	MQQUEUE
Alter	<i>qmgr</i> .SYSTEM.CLUSTER.COMMAND.QUEUE	MQQUEUE
Alter	<i>qmgr</i> .SYSTEM.CLUSTER.REPOSITORY.QUEUE	MQQUEUE

IVP Test Execution

Submit the JCL to execute the CSQ4IVPX program. This verification comprises the execution of three commands. These commands are:

- 1) START CHINIT
- 2) DISPLAY CHINIT
- 3) STOP CHINIT

Each of these commands should display a "Normal" completion message. The program should complete with a return code of zero. Trouble-shoot any abnormal completion codes.

CSQ4IVPX Sample Report

```
CSQU000I CSQUTIL IBM MQ for z/OS - V6
CSQU001I CSQUTIL Queue Manager Utility - 2005-05-09 09:06:48
COMMAND
CSQU127I CSQUTIL Executing COMMAND using input from CSQUCMD
data set
CSQU120I CSQUTIL Connecting to queue manager CSQ1
CSQU121I CSQUTIL Connected to queue manager CSQ1
CSQU055I CSQUTIL Target queue manager is CSQ1
START CHINIT
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQM138I +CSQ1 CSQMSCHI CHANNEL INITIATOR STARTING
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQXCRPS ' START CHINIT' NORMAL COMPLETION
DISPLAY CHINIT
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQM137I +CSQ1 CSQMDDQM DISPLAY CHINIT COMMAND ACCEPTED
CSQN205I COUNT= 12, RETURN=00000000, REASON=00000000
CSQX830I +CSQ1 CSQXRDM Channel initiator active
CSQX002I +CSQ1 CSQXRDM Queue-sharing group is QSG1
CSQX831I +CSQ1 CSQXRDM 8 adapter subtasks started, 8
requested
CSQX832I +CSQ1 CSQXRDM 5 dispatchers started, 5 requested
CSQX833I +CSQ1 CSQXRDM 0 SSL server subtasks started, 0
requested
CSQX840I +CSQ1 CSQXRDM 0 channel connections current, maximum
200
CSQX841I +CSQ1 CSQXRDM 0 channel connections active, maximum
200,
including 0 paused
```




```
CSQX842I +CSQ1 CSQXRQDM 0 channel connections starting,  
0 stopped, 0 retrying  
CSQX836I +CSQ1 Maximum channels - TCP/IP 200, LU 6.2 200  
CSQX845I +CSQ1 CSQXRQDM TCP/IP system name is TCPIP  
CSQX848I +CSQ1 CSQXRQDM TCP/IP listener INDISP=QMGR not  
started  
CSQX848I +CSQ1 CSQXRQDM TCP/IP listener INDISP=GROUP not  
started  
CSQX849I +CSQ1 CSQXRQDM LU 6.2 listener INDISP=QMGR not  
started  
CSQX849I +CSQ1 CSQXRQDM LU 6.2 listener INDISP=GROUP not  
started  
CSQ9022I +CSQ1 CSQXCRPS ' DISPLAY CHINIT' NORMAL COMPLETION  
STOP CHINIT  
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004  
CSQM137I +CSQ1 CSQMTCHI STOP CHINIT COMMAND ACCEPTED  
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000  
CSQ9022I +CSQ1 CSQXCRPS ' STOP CHINIT' NORMAL COMPLETION  
CSQU057I CSQUCMDS 3 commands read  
CSQU058I CSQUCMDS 3 commands issued and responses received, 0  
failed  
CSQU143I CSQUTIL 1 COMMAND statements attempted  
CSQU144I CSQUTIL 1 COMMAND statements executed successfully  
CSQU148I CSQUTIL Utility completed, return code=0
```



Verifying Queue Sharing Groups

Introduction

When the CSQ4IVP1 program, as described above, was executed its API tests were executed against local (non-shared) queues. The CSQ4IVP1 program can also be configured to execute against shared queues, and therefore validate:

- Coupling Facility setup
- Queue Sharing Group setup
- Shared Queue processing

Preparation

To prepare to execute the CSQ4IVP1 program for Shared Queues, take the following additional steps:

- Update the Coupling Facility Resource (CFRM) policy dataset to include the Coupling Facility structure used by the IVP program. By default, a structure named “APPLICATION1” is used.
- Verify that the Queue Sharing Group IVP entries are in the “CSQINP2” DD concatenation. The IVP entries are described in the following dataset:

hlq.v9r0m0.SCSQPROC(CSQ4IVPG)

- Modify the JCL to execute the CSQ4IVP1 program. This JCL is found in the following dataset:

hlq.v9r0m0.SCSQPROC(CSQ4IVPS)

- Ensure that the User ID that the JCL will be submitted under has adequate authority, as defined in the ESM, to access the necessary MQ objects. The following additional access for the Shared Queue resources is required:

Access	Resource Profile	Resource Class
Update	<i>qmgr</i> .CSQ4IVPG.**	MQQUEUE

IVP Test Execution

Submit the JCL to execute the CSQ4IVP1 program. The execution of this program is identical to the original execution except that Shared Queues are used to test the MQI calls. Trouble-shoot any “stage” with a non-zero return code.



References

- IBM – Knowledge Center – z/OS Installation Verification (v9.0)
https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_9.0.0/com.ibm.mq.con.doc/q019380_.htm
- IBM – Knowledge Center – Installation Verification Program (CSQ4IVP1)
https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_9.0.0/com.ibm.mq.con.doc/q019390_.htm