Using IBM MQ in CICS applications

Jamie Squibb (jamie_squibb@uk.ibm.com)
IBM

01/11/2016
Session JE
Agenda

- MQ-CICS adapter
- MQ-CICS bridge
- New start and stop facilities
- Summary
CICS adapter
CICS adapter

- MQ adapters provide MQI support to application environments
  - Separate adapters for the channel initiator, CICS, IMS, BATCH and RRSBATCH
- Applications (CICS programs) link with a stub
  - The stub loads/calls the adapter, which communicates with local queue managers
  - Separate stub for each adapter
    - For CICS the stub is CSQCSTUB
- The adapter processes MQI requests for the application
  - For example, establishes a connection to the queue manager for MQCONN
  - CICS transactions do not need to issue MQCONN or MQDISC
    - The adapter maintains a shared connection for the CICS region
- Commit and rollback are handled by the CICS syncpoint coordinator
Example CICS adapter workflow

CICS transactions link with CSQCSTUB to use the MQI.

Transactions can be triggered upon the arrival of messages or can wait for messages to arrive on queues using MQGET (wait).

Multiple CKTI instances can be started to monitor initiation (INIT) queues.

CKQC transaction shows what is happening.
Connecting CICS to MQ

- **MQCONN resource in CICS**
  - Defines connection attributes for MQ
  - One **MQCONN** resource per CICS region
  - Must install an **MQCONN** resource before starting an MQ connection
  - Attributes include:
    - Connection name - queue manager or queue-sharing group name (**MQNAME**)
    - Resynchronisation strategy for in-doubt units of work (**RESYNCMEMBER**)
    - Default initiation queue (**INITQNAME**) - automatically start a trigger monitor (CKTI) instance
  - Managed using **CEDA** and **CEMT** (or equivalents such as CICSplex SM)
Resynchronisation (RESYNCMEMBER)

- Introduced in CICS 4.1
- Applicable only when connecting using a queue-sharing group (QSG) name
- Specifies strategy for resolving outstanding (in-doubt) units of work
- Ensure all resources are in a consistent state before changing this setting

- **YES**
  - Connect only to the same queue manager (wait to become if necessary) – default value
- **NO**
  - Attempt to connect to the same queue manager once
  - If not available connect to any other available queue manager in the QSG but leave outstanding units of work unresolved – issues DFHMQ2064
- **GROUPRESYNC**
  - Connect to any available queue manager in the QSG
  - Resolve in-doubt units of work for all eligible queue managers in the QSG
  - Requires CICS 4.2 and MQ 7.1 plus MQ group units of recovery (GROUPUR) to be enabled
Example using `GROUPRESYNC`

CICS automatically reconnects to MQ1B and resolves in-doubt units of work started on MQ1A. In-flight units of work are backed out.

<table>
<thead>
<tr>
<th>CICS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transactions access shared queues</td>
</tr>
</tbody>
</table>

- **MQCONN**
- **MQNAME=QSG1**
- **RESYNCMEMBER=GROUPRESYNC**

If MQ1A and MQ1B are on different LPARs CICS must be restarted on the LPAR where MQ1B is running.

Queue manager MQ1A

Queue manager MQ1B

Same LPAR

QSG1

CF
Managing a connection using CKQC

- Provides access to control functions for the CICS adapter
  - Start/stop a connection to MQ
  - Display connection status and settings
  - Display/reset connection statistics (e.g. numbers of MQI API calls)
  - Display tasks using the MQ connection
  - Start instances of CKTI (trigger monitor / task initiator)

- Available via control panels or command line
- Can also use `SET / INQUIRE MQCONN`
Task initiator / trigger monitor (CKTI)

• Starts a CICS transaction when a trigger message is read
  1. Trigger generated when a message is put to an application queue and conditions met
     • Every message, first message on queue, etc.
  2. Trigger message generated by MQ and put to initiation queue
  3. CKTI monitors the initiation queue and gets the trigger message
  4. CKTI starts another CICS transaction to get and process the application message
     • CICS transaction defined using the MQSC command `DEFINE PROCESS`
     • The process object is named on the application queue

• Can start multiple instances of CKTI
  – Each instance of CKTI exclusively services a single initiation queue
  – CKQC STARTCKTI / CKQC STOPCKTI
Alert monitor (CKAM)

- Handles unscheduled (pending) events
  - Events occur as a result of connect requests to MQ
  - Generates messages sent to system console
- Two types of event:
  - Deferred connection:
    - Generated if CICS attempts to connect to an unavailable queue manager
    - Could be a single queue manager or a queue-sharing group name
    - When available CICS connects and the pending event is cancelled
  - Termination notification
    - Created when a connection is made to MQ
    - Event expires when queue manager ends or CKQC closes the connection from CICS
- Automatically starts and stops
CICS bridge
CICS bridge

• Provides MQ interface to legacy CICS applications
  – Transactions that are not ‘MQ aware’ – i.e. don’t use the MQI

• Two flavours:
  – Distributed program link (DPL) bridge
    • EXEC CICS LINK to program with data in the COMMAREA
  – 3270 bridge: Transactions can use Basic Mapping Support (BMS) or terminal control commands – can be conversational or part of a pseudo-conversation

• Bridge monitor:
  – Browses queue
  – Performs authority checks
  – Starts a transaction to process each message

• Multiple bridge monitors can be started
• Supports transaction routing
CICS bridge monitor task (CKBR)

• Waits for messages to arrive on configured bridge queue
• Starts a CKBP bridge task when new session requests arrive
  – CKBP performs interaction with legacy application using COMMAREA or 3270 data
    • Data passed in MQ messages
  – CKBP processes all messages for that session
  – CKBC supports using a container instead of the COMMAREA
• Can start manually or trigger on arrival of first message
• Bridge task ends when either:
  – CICS is shutdown
  – Queue manager is shutdown
  – Bridge queue is get inhibited using GET(DISABLED)
Running DPL programs under the CICS bridge

1. Client application sends a request message to run a CICS transaction.
Running DPL programs under the CICS bridge

2. The CICS bridge monitor task browses the message to examine its content
3. The bridge monitor task starts the CICS DPL bridge task (CKBP, CKBC or a custom transaction)
Running DPL programs under the CICS bridge

4. The CICS DPL bridge task retrieves the request message

Client application

Channel initiator

Queue manager

Request queue

CICS

Queue manager

CICS bridge monitor (CKBR)

EXEC CICS START

CICS DPL bridge task (CKBP)

MQPUT (request)

MQGET (browse)

MQGET

MQGET (browse)
5. The CICS DPL bridge task starts the user program, passing the payload in the COMMAREA (CKBP) or container/channel (CKBC)
Running DPL programs under the CICS bridge

6. The user program returns control to the CICS DPL bridge task, passing any response in the COMMAREA (CKBP) or container/channel (CKBC)
Running DPL programs under the CICS bridge

7. The CICS DPL bridge task puts the response as a reply message for the application to get
Running 3270 transactions under the CICS bridge

• CICS transaction runs as if it has a real 3270 terminal
• MQ messages simulate terminal interaction using CICS Link3270
• MQ messages consist of ‘vectors’ that correspond to an EXEC CICS request
  – MQ application interacts directly with the 3270 transaction
  – The vectors provide terminal-related input
• Terminal-related output from the transaction maps to ‘vectors’ in the response
• CICS bridge task ends the transaction (or waits for next input)
Running 3270 transactions under the CICS bridge

Client application

Channel initiator

Queue manager

Request queue

Reply queue

CICS

CICS bridge monitor (CKBR)

EXEC CICS START

CICS bridge task (CKBP)

3270 bridge

User transaction

MQGET (wait)

MQGET (request)

MQPUT (request)

MQGET (browse)

MQGET (reply)

MQPUT (reply)
Securing the CICS bridge (1)

- Can specify the level of authentication performed by CKBR using the AUTH parameter:
  
  - **LOCAL** (default value)
    - All bridge tasks and user programs are started using the CICS default user ID
  
  - **IDENTIFY**
    - Starts bridge tasks and user programs as the user ID specified in the MQ message descriptor (MQMD) or the CICS default user ID if not specified
    - Users treated as trusted – no authentication performed
Securing the CICS bridge (2)

- **VERIFY_UOW**
  - Authenticates the user specified in the MQMD if a password or PassTicket is provided in an accompanying MQCIH header. If authentication succeeds that user ID is used, else the request fails with MQCRC_SECURITY_ERROR.
  - CICS default user ID is used if credentials are not provided.

- **VERIFY_ALL**
  - Same as VERIFY_UOW but authenticates every request message and verifies the same user ID is used for every request message in the same unit of work.
Securing the CICS bridge (3)

- When using **IDENTIFY**, **VERIFY_UOW** or **VERIFY_ALL**
  - User IDs must be defined to RACF
  - CICS surrogate security can be used to restrict transaction and user ID combinations
  - MQ applications must use **MQOO_SET.IDENTITY_CONTEXT** and **MQPMO_SET.IDENTITY_CONTEXT**
Application message: Message descriptor

- Application messages set the following fields in the MQMD:
  - CorrelId
    - Set to MQCI_NEW_SESSION in the first/only message for a transaction
    - Set to the message ID returned by MQ for the first message
    - When getting reply messages use the message ID of the most recent request message
  - Format
    - Set to MQCICS if a MQCIH header is provided, else the format of the data
  - ReplyToQ
    - Set to the name of the queue you want reply messages to be put to
    - Reply messages have the same persistence as the request – if persistent request the you cannot use temporary dynamic reply queue (or unrecoverable shared reply queue)
  - UserIdentifier
    - The user ID to check for access to the CICS DPL program
    - Applicable only if authorisation level is IDENTIFY, VERIFY_UOW or VERIFY_ALL
Application message: MQCIH header

- Optional only if authentication mode is `LOCAL` or `IDENTIFY`
- Must use version 2 header for 3270 transactions
- Contains both input and output fields
- Key input fields include:
  - Authenticator – the user’s password or PassTicket (`VERIFY_UOW` or `VERIFY_ALL`)
  - Flags – customise expiration, sync on return and/or trailing nulls
  - Format – the format of the user data that follows the header
  - GetWaitInterval – how long the bridge task should wait for a response
  - LinkType – whether to run a DPL program or a 3270 transaction
  - TransactionId – optional transaction code for the DPL program (default is `CKBP`)
  - UOWControl – controls the unit of work processing performed by the bridge
New start and stop facilities
Today’s environment (recap)

• CICS transaction **CKQC** controls the MQ connection and **CKTI** transactions
• Use the **INITQNAME** parameter on the **MQCONN** definition to automatically start a single instance of **CKTI**
• In order to start more than one instance of the **CKTI** transaction the user enters “**CKQC STARTCKTI CSQ4SAMP.INITIATION.QUEUE**”
  – At a CICS sequential terminal
  – At a MVS console
  – Start a terminal based transaction that retrieves the data
• Alternatively write an application program that links to program **DFHMQSSQ**
Issues with today’s environment

• If the MQ connection is restarted all transactions accessing MQ queues are terminated.

• When the connection is re-established the transaction associated with the `INITPARM` parameter on the MQ connection is restarted automatically; any user initiated `CKTI` transactions must be manually restarted.

• The user ID associated with the new `CKTI` transaction is that of the user issuing the restart request not the original user.

• The default user ID associated with the transaction retrieving the application message is now that of the user who issued the restart request.

• Cannot stop the MQ bridge transaction without stopping / restarting the MQ connection.
New in CICS Transaction Server 5.4

- New CICS resource - MQMonitor
- Created using RDO, EXEC CICS CREATE, CPSM BAS, CICS Explorer

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transaction</td>
<td>Used by the transaction monitoring the MQ queue (defaults to CKTI at installation time)</td>
</tr>
<tr>
<td>MONUserid</td>
<td>User ID to be associated with the queue monitoring transaction</td>
</tr>
<tr>
<td>Qname</td>
<td>MQ queue name (defaults to &amp;applid.INITIATION.QUEUE at installation time if the transaction is CKTI)</td>
</tr>
<tr>
<td>Userid</td>
<td>Default user ID to be associated with the transaction that will retrieve the application message</td>
</tr>
<tr>
<td>MONData</td>
<td>Free-format text up to 255 characters passed to the MQ monitoring transaction</td>
</tr>
<tr>
<td>Autostart</td>
<td>Yes/No</td>
</tr>
<tr>
<td>Enabled</td>
<td>Yes/No</td>
</tr>
</tbody>
</table>
Resource definition for the CICS MQ bridge

```
OVERTYPE TO MODIFY CICS RELEASE = 0710
CEDA ALTER MQMonitor( DFHMQBR0 )
MQMonitor : DFHMQBR0
Group : MQMON
OEScription ==>
Status ==>
Enabled | Disabled
MONUserid ==>
MONData ==>
Q=FY.IYK2ZEV1.CICS.BRIDGE.QUEUE.WAIT=20,MSG=LOG
(Mixed Case) ==>
MONUserid ==>
Qname ==>
Transaction ==>
CKBR
APPLICATION ATTRIBUTES
Userid ==>
+ DEFINITION SIGNATURE
  SYSID=GMB1 APPLID=IYK2ZEV1
```

- **Autostart**: whether started automatically when connected to MQ
- **Transaction**: monitoring transaction to start, defaults to CKTI
- **MONUserid**: user ID used to start the monitoring transaction
- **MONData**: is arbitrary data that can be passed to the monitoring transaction
- **Userid**: used by the monitoring transaction for any application transactions that it starts (e.g. when CKTI starts another transaction)
The new implementation continued…

• Two user ids associated with the MQMonitor
  - MONUSERID used by the task monitoring the queue
  - USERID is the default user id for the task to be started if no alternative is available
    (defaults to CICS DFLTUSER if not specified)

• New SPI resource EXEC CICS SET / INQUIRE MQMONITOR
  - Autostart option allows associated transaction to be started when CICS connects to MQ
  - Ability to manually start and stop using CEMT, CPSM, CICS Explorer and the CICS SPI
  - Only Autostart, Enablestatus and Monitorstatus may be changed.
  - Changes to the queue name, the user data, the transaction or the user ids require that
    the resource be reinstalled
  - Cannot be discarded unless disabled and stopped
CECI or EXEC CICS SET command

```
SET MQM(DFHMQBRO) STARTED
STATUS: COMMAND EXECUTION=COMPLETE

EXEC CICS SET MQMonitor('DFHMQBRO')
  AUTOSTART [Noautostart]
  ENABLE [Disabled]
  Start [STOPped]

RESPONSE: NORMAL
EIBRESP=+0000000000 EIBRESP2=+0000000000
PF 1 HELP 2 HEX 3 END 4 EIB 5 VAR 6 USER 7 SBH 8 SFH 9 MSG 10 SB 11 SF
```
The new implementation continued…

- Responsibilities of the user written monitor transaction:
  - The transaction monitoring the MQ queue has to run in the local region
  - The user transaction is started with data in the following format:
    - Left sharp bracket ‘<
    - Characters 2 to 9: MQMonitor name
    - Characters 10 to 17: User id for starting the transaction associated with the application if no other user id available
    - Right sharp bracket “>”
    - Characters 17 to 271: User data specified on the resource
  - At transaction initialisation the user written program must set the **MQMonitor Monstatus** to **STARTED** using the SPI
  - At transaction termination the user written program must set the **MQMonitor Monstatus** to **STOPPED** using the SPI
CEMT INQUIRE MQMONITOR (DFHMQBR0)

Highlighted text is not part of the defined monitor data.

It comprises the MQMONITOR name and user ID.

It is passed in the data on the EXEC CICS START command (brackets included).
CEMT Set MQConn connected

SET MQC CONNECTED
STATUS: RESULTS --OVERTYPE TO MODIFY NORMAL
Mqname( MQD5 )
Mqqmgr(MQD5)
Mqrelease(0710)
Connectst( Connected )
Resyncmember( )
Tasks(0003)
Trigmontasks(0003)
Installtime(06/16/16 07:08:39)
Installusrid(CICSUSER)
Installagent(Grplst)
Definesource(MQCONN)
Definetime(12/03/08 05:30:57)
Changetime(04/12/16 08:44:25)
Changeusrid(CICSUSER)
Changeagent(Cedapi)
Changeagrel(0710)

RESPONSE: NORMAL
TIME: 08:59.26  DATE: 06/16/16
PF  1 HELP    3 END    5 VAR    7 SBH  8 SFH  9 MSG  10 SB  11 SF
<table>
<thead>
<tr>
<th>MQm</th>
<th>Status</th>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFHMQBRG</td>
<td>Aut</td>
<td>Ena</td>
<td>Sta Tas(0000138) Tra(CKBR)</td>
</tr>
<tr>
<td>DFHMQINI</td>
<td>Aut</td>
<td>Ena</td>
<td>Sta Tas(0000139) Tra(CKTI) Qna(IYK2ZFV1.INITIATION.QUEUE )</td>
</tr>
<tr>
<td>MQMOMON1</td>
<td>Aut</td>
<td>Ena</td>
<td>Sta Tas(0000140) Tra(CKTI) Qna(IYK2ZFV1.INITIATION.QUEUE )</td>
</tr>
<tr>
<td>MQMOMON2</td>
<td>Aut</td>
<td>Ena</td>
<td>Sta Tas(0000141) Tra(CKTI) Qna(CSQ4SAMP.INITIATION.QUEUE )</td>
</tr>
<tr>
<td>MYMQMON3</td>
<td>Aut</td>
<td>Ena</td>
<td>Sta Tas(0000142) Tra(CKTI) Qna(IYK2ZFV1.PIPE4.INITQ )</td>
</tr>
</tbody>
</table>

SYSID=GMB1 APPLID=IYK2ZFV1

RESPONSE: NORMAL
TIME: 09.05.00 DATE: 06/16/16
PF 1 HELP  3 END  5 VAR  7 SBH  8 SFH  9 MSG  10 SB  11 SF
### CEMT SET MQC NOTConnected results

<table>
<thead>
<tr>
<th>MQM</th>
<th>Status</th>
<th>Queue</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mqm(DFHMQBR0)</td>
<td>Aut</td>
<td>Sto</td>
<td>Tas(00000000) Tra(CKBR)</td>
</tr>
<tr>
<td>Mqm(DFHMQINI)</td>
<td>Aut</td>
<td>Sto</td>
<td>Tas(00000000) Tra(CKTI) Qna(IYK2ZFY1.INITIATION.QUEUE)</td>
</tr>
<tr>
<td>Mqm(MQMOMON1)</td>
<td>Aut</td>
<td>Sto</td>
<td>Tas(00000000) Tra(CKTI) Qna(IYK2ZFY1.INITIATION.QUEUE)</td>
</tr>
<tr>
<td>Mqm(MQMOMON2)</td>
<td>Aut</td>
<td>Sto</td>
<td>Tas(00000000) Tra(CKTI) Qna(CSQ4SAMP.INITIATION.QUEUE)</td>
</tr>
<tr>
<td>Mqm(MYMQMON3)</td>
<td>Aut</td>
<td>Sto</td>
<td>Tas(00000000) Tra(CKTI) Qna(IYK2ZFY1.PIPE4.INITQ)</td>
</tr>
</tbody>
</table>

**RESPONSE:** NORMAL

**TIME:** 09.07.56  **DATE:** 06/16/16

**PF 1 HELP  3 END  5 VAR  7 SBH  8 SFH  9 MSG  10 SB  11 SF**
A pictorial explanation!

MQCONN

1. Connect

MQ

2. Start

MQMONITOR

Autostart (YES)
Transaction (CKTI)
Qname (MY.INITQ)
MONUserid (BOB)
Userid (FRED)

The monitor transaction

EXEC CICS START
TRANSID (CKTI)
USERID (BOB)

The application transaction

EXEC CICS START
TRANSID (UTRN)
USERID (FRED)

3. Get message

4. Get message

5. Get message

6. Get message

MY.INITQ

MY.REALQ
Notes

1. **MQCONN** connects to MQ
2. All **MQMONITOR** objects which have **Autostart (yes)** set are started
3. Monitoring transaction (**Transaction**) started using specified user id (**MONUserid**) and queue (**Qname**)
4. In this case **CKTI** is used which opens an initiation queue and sits waiting for trigger messages
5. When a trigger message is received the message data is used to start the relevant application transaction using the specified user ID (**Userid**)
CICSPlex SM web user interface (WUI)
CICSPlex SM WUI – All installed MQ monitors

![CICSPlex SM WUI - All installed MQ monitors](image_url)
CICSPlex SM WUI – CKBR CICS MQ Bridge MQ monitor
CICS Explorer – MQMonitors (operations)

<table>
<thead>
<tr>
<th>Region</th>
<th>MQ Monitors</th>
<th>MQ Queue ID</th>
<th>Enabled Status</th>
<th>AutoStart Status</th>
<th>Monitor Status</th>
<th>Task Number</th>
<th>Monitor Trace</th>
<th>Monitor User</th>
<th>Target Userid</th>
</tr>
</thead>
<tbody>
<tr>
<td>YK2ZFPV1</td>
<td>APPID</td>
<td>NYK2ZFPV1.NYK2ZFPV1</td>
<td>ENABLED</td>
<td>NOAUTOSTART</td>
<td>STOPPED</td>
<td>000000</td>
<td>cktd</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>CSQ4SAMP</td>
<td>CSQ4SAMP</td>
<td>ENABLED</td>
<td>AUTOSTART</td>
<td>STARTED</td>
<td>0000062</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>DFHMQBRO</td>
<td>DFHMQBRO</td>
<td>ENABLED</td>
<td>AUTOSTART</td>
<td>STARTED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>DFHIQBL</td>
<td>DFHIQBL</td>
<td>ENABLED</td>
<td>AUTOSTART</td>
<td>STARTED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>DFHIQCM</td>
<td>DFHIQCM</td>
<td>ENABLED</td>
<td>AUTOSTART</td>
<td>STARTED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>DFHIQSD</td>
<td>DFHIQSD</td>
<td>ENABLED</td>
<td>AUTOSTART</td>
<td>STARTED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>DFHIQTM</td>
<td>DFHIQTM</td>
<td>ENABLED</td>
<td>AUTOSTART</td>
<td>STARTED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>EPD8RBD</td>
<td>EPD8RBD</td>
<td>DISABLED</td>
<td>AUTOSTART</td>
<td>STOPPED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>PIPE3</td>
<td>PIPE3</td>
<td>ENABLED</td>
<td>STARTED</td>
<td>STARTED</td>
<td>0000066</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>PIPE4</td>
<td>PIPE4</td>
<td>STARTED</td>
<td>STARTED</td>
<td>STARTED</td>
<td>0000067</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>PVCBI</td>
<td>PVCBI</td>
<td>DISABLED</td>
<td>AUTOSTART</td>
<td>STOPPED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>PVCBI</td>
<td>PVCBI</td>
<td>DISABLED</td>
<td>AUTOSTART</td>
<td>STOPPED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
<tr>
<td>YK2ZFPV1</td>
<td>PVCBI</td>
<td>PVCBI</td>
<td>DISABLED</td>
<td>AUTOSTART</td>
<td>STOPPED</td>
<td>000000</td>
<td>CKTD</td>
<td>CICSSUSER</td>
<td>CICSSUSER</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Type</td>
<td>LOCAL</td>
</tr>
<tr>
<td>Routeid</td>
<td>1</td>
</tr>
<tr>
<td>CMSS Name</td>
<td>YK2ZFPV1</td>
</tr>
<tr>
<td>CMSS Status</td>
<td>ACTIVE</td>
</tr>
<tr>
<td>CMSS System ID</td>
<td>GM81</td>
</tr>
<tr>
<td>MP Status</td>
<td>YES</td>
</tr>
<tr>
<td>Name</td>
<td>GM81</td>
</tr>
</tbody>
</table>
CICS Explorer – MQMonitors (definitions)
CICS Explorer MQMonitor statistics
DFH0STAT print report selection

Sample Program - CICS Statistics Print Report Selection
10/10/2016 12:11:34

Select the statistics reports required and press 'Enter' to validate

DB2 Connection ................. N
DB2 Entries ................... N
WebSphere MQ Connection .... N
WebSphere MQ Monitors ....... Y
Program Autoinstall .......... N
Terminal Autoinstall and VTAM.. N
Connections and Modenames .. N
TCP/IP ........................ N
TCP/IP Services ............... N
IPCONNs ..................... N

BUNDLeS ...................... N
URIMAps ..................... N
Virtual Hosts ................. N

Event Processing ............... N
EPADAPTERs ................... N
ATOMSERVICEs ................. N
PIPELINES ................... N
EVENTBINDINGS ............... N
WEBSERVICEs ................. N
CAPTURESPECs ................. N
DOCTEMPLATEs ................. N
XMLTRANSFORMS .............. N

F1=Help  F8=Return to Print  F7=Back  F8=Forward  F10=Save  F12=Restore
DFHSTUP resource selection

000027  //DFHSTWRK DD UNIT=VIO,SPACE=(CYL,(8,3))
000028  //SORTWK01 DD UNIT=VIO,SPACE=(CYL,(10))
000029  //DFHPRINT DD SYSOUT=* 
000030  //TRACEOUT DD SYSOUT=* 
000031  //SYSDUMP DD SYSOUT=* 
000032  //SYSABEND DD SYSOUT=* 
000033  /* DATE FORMAT IS ALWAYS MM/DD/CCYY
000034  //SYSIN DD *
000035  SELECT APPLID=IYK2ZFY1
000036  COLLECTION TYPE=ALL
000037  SELECT TYPE=(MQCONN, MQMONITOR)
000038  DATE START=09/27/2016,STOP=09/27/2016
000039  TIME START=07.00.00,STOP=08.00.00
000040  SUMMARY
000041  //
Open requests for enhancement satisfied by this item

- **22364** Expand the number of MQ initiation queues that can be defined and started by CICS
- **23115** Require the ability to start an MQ Trigger Monitor and specify a USERID that the transaction will use when running
- **25010** Enhance MQCONN resource to start CKTI with correct userid
- **25011** Ability to invoke MQ Adaptor functions from a non-terminal task
- **60442** CKTI security concern
- **65130** CICS API for MQM resources
- **67163** CICS MQ Bridge configuration
- **74231** CKTI does not restart on a MQ 2345 and 2373 error
- **84580** Starting MQ CICS connection with PLTUSR ID
Open Requests for Enhancement potentially satisfied by this item

• 220011 Make &APPLID available for MQCONN INITQNAME
Summary
Summary

• MQ-CICS adapter provides MQI support for CICS applications

• MQ-CICS bridge provides MQ support for running legacy CICS applications
  – Both DPL and 3270

• New start and stop facilities are provided in CICS Transaction Server 5.4
We value your feedback

• Tell us your priorities
  – Raise RFEs and vote on them
• Sign up to the beta program
  – Get your first look at what’s coming
• Engage with your Lab Advocates
  – Stay connected
• Tell us what you think
  – What’s important to you?
Where can I get more information?

IBM Messaging developerWorks
developer.ibm.com/messaging

IBM Messaging Youtube
https://www.youtube.com/IBMmessagingMedia

LinkedIn
Ibm.biz/ibmmessaging

Twitter
@IBMMessaging

IBM MQ Facebook
Facebook.com/IBM-MQ-8304628654/
Session feedback

• Please submit your feedback at:

  http://conferences.gse.org.uk/2016/feedback/JE

• Session is JE
Notices and disclaimers

Copyright © 2016 by International Business Machines Corporation (IBM). No part of this document may be reproduced or transmitted in any form without written permission from IBM.

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IN NO EVENT SHALL IBM BE LIABLE FOR ANY DAMAGE ARISING FROM THE USE OF THIS INFORMATION, INCLUDING BUT NOT LIMITED TO, LOSS OF DATA, BUSINESS INTERRUPTION, LOSS OF PROFIT OR LOSS OF OPPORTUNITY. IBM products and services are warranted according to the terms and conditions of the agreements under which they are provided.

Any statements regarding IBM’s future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

It is the customer’s responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer’s business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer is in compliance with any law.
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 in connection with this publication 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. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM’s products. IBM EXPRESSLY DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

IBM, the IBM logo, ibm.com, Aspera®, Bluemix, Blueworks Live, CICS, Clearcase, Cognos®, DOORS®, Emptoris®, Enterprise Document Management System™, FASP®, FileNet®, Global Business Services®, Global Technology Services®, IBM ExperienceOne™, IBM Social Business®, Information on Demand, ILOG, Maximo®, MQIntegrator®, MQSeries®, Netcool®, OMEGAMON, OpenPower, PureAnalytics™, PureApplication®, pureCluster™, PureCoverage®, PureData®, PureExperience®, PureFlex®, pureQuery®, pureScale®, PureSystems®, QRadar®, Rational®, Rhapsody®, Smarter Commerce®, SoDA, SPSS, Sterling Commerce®, StoredIQ, Tealeaf®, Tivoli®, Trusteer®, Unica®, urban(code)®, Watson, WebSphere®, Worklight®, X-Force® and System z® Z/OS, are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at: www.ibm.com/legal/copytrade.shtml.
Using IBM MQ in CICS applications

Jamie Squibb (jamie_squibb@uk.ibm.com)
IBM

01/11/2016
Session JE