Handling MQ in-doubt transactions

Girish D V |Aug 5 2015 Updated

In-doubt transactions are the incomplete transactions which are prepared but not yet committed. Each transaction will enter into in-doubt phase during its lifecycle and sometimes these transactions will not commit, remaining in-doubt. These transactions cannot be resolved automatically, so a manual user intervention is needed to resolve these transactions by either committing or backing out. These in-doubt transactions can be either internally co-ordinated or externally co-ordinated. Internally co-ordinated transactions are initiated by MQ, and externally co-ordinated transactions are initiated by external applications via a transaction manager. In a few cases some of the transactions remain in-doubt for reasons such as the channel hosting the connection to the queue manager went into an in-doubt state or the application went down. Whenever there are in-doubt transactions in the queue manager, these will be displayed as uncommitted messages on the queue. For example the following queue shows uncommitted messages:
dis QS(XXX)
     1 : dis QS(XXX)
AMQ8450: Display queue status details.
   QUEUE(XXX)                              TYPE(QUEUE)
   CURDEPTH(9)                             IPPROCS(0)
   LGETDATE( )                             LGETTIME( )
   LPUTDATE( )                             LPUTTIME( )
   MEDIALOG( )                             MONQ(OFF)
   MSGAGE( )                               OPPROCS(0)
   QTIME( , )                              UNCOM(8)

The above shows 8 uncommitted messages on the queue. These uncommitted messages will not even allow us to clear the queue. These uncommitted messages can be handled using "dspmqtrn" and "rsvmqtrn" command.

The "dspmqtrn" command can be used to display all in-doubt transactions. The following link provides information with all the options available with "dspmqtrn"
<http://www-01.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.ref.adm.doc/q083270_.htm>

All the in-doubt can be resolved using "rsvmqtrn" command. Internally co-ordinated transactions can be resolved using '-a' option. Externally co-ordinated transactions can be resolved either by commit with '-c' option or backout '-b' option. A restart on the queue manager can resolve all internal in-doubt transactions but not external in-doubt transactions. More information about resolving in-doubt transactions is available in the following link
<http://www-01.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.ref.adm.doc/q083390_.htm>

In some situations the uncommitted transactions remain or the keep increasing on the queue even after resolving in-doubt transactions. We need to identify the transactions remaining on the queue for long time as uncommitted messages and confirm that "rsvmqtrn" did actually resolve the in-doubt transactions. In such situations it is ideal to capture 2 sets of "dspmqtrn" command output with a few minutes interval. First set of "dspmqtrn" output needs to be captured before resolving in-doubt transactions and second set after resolving in-doubt transactions then compare the transaction number and gtrid values. We cannot use just the transaction number for comparing the transactions as the transaction number of an committed transaction number can be reused. This comparison will help us confirm if the transactions seen are same old in-doubt transaction or if new in-doubt transactions are being added. For example the following is a output of "dspmqtrn" command output:

AMQ7056: Transaction number 0,1.
    XID: formatID 5067085, gtrid\_length 12, bqual\_length 4
         gtrid [3291A5060000201374657374]
         bqual [00000001]

We can use the transaction number and the global transaction id values to identify the transaction as the combination of these values make a unique combination for a transaction. If more information is needed on the transactions we can collect the dumps for the transaction using the command "amqldmpa". The following commands can be used to collect the dumps using "amqldmpa" which provides information of all the available transactions and their connection details:
amqldmpa -m QMGR-NAME -c A -o 2 -f/tmp/atm
amqldmpa -m QMGR-NAME -c K -d 3 -f /tmp/kern

The atm output provides state and details of the transaction. Below is the sample piece of the atm output which shows the state, global transaction id(gtrid), transaction number(TranNum), transaction id(Tranid) and other details of the transaction:

  StructID   "ATXN"
  State      PREPARED|HARDENED|LOGGED
  XID: formatID 1463898948,gtrid\_length 36, bqual\_length 54
  gtrid [0000014BCC5C37EF00000001002E77694C389E3844
         23B04498B02F9D88AA4E680ABDF698]
  bqual [0000014BCC5C37EF00000001002E77694C389E3844
         23B04498B02F9D88AA4E680ABDF698000000010000
         000000000000000000000004]
  TranNum    0.165024653
  Tranid     2::13::13-641880
  hmtxTranData.RequestCount    4973
  FirstLSN   0.245.64916.30968
  LastLSN    0.245.64916.31204
  RestartLSN 0.245.64916.30968
  TTLockCount      4972
  TDLockCount      4972
  CreateTime 2015-04-27 15:34:18.000
    ............................

The transaction number or the transaction id can used to co-relate the atm and the kern outputs where the kern output provides the connection details of the transactions. For all the active transactions kern output provides details of the application, object details, connection id and other details. Important note is that connection details of only the active transaction will be available in the kern output. The in-doubt transactions will be in "PREPARED|HARDENED|LOGGED" state which has no information about the connection details in the kern output. In such situations queue id(Qid) of the in-doubt transactions can be of help in identifying the application that initiated the transaction. With the above information on in-doubt transactions we can check on why the transactions are moving into in-doubt state and take appropriate measures to handle those situations.