IBM Software Group

How to Automate Handling of WebSphere® MQ Dead Letter Messages

Greg Bowman, WebSphere MQ Support, IBM

WebSphere® Support Technical Exchange

Introduction

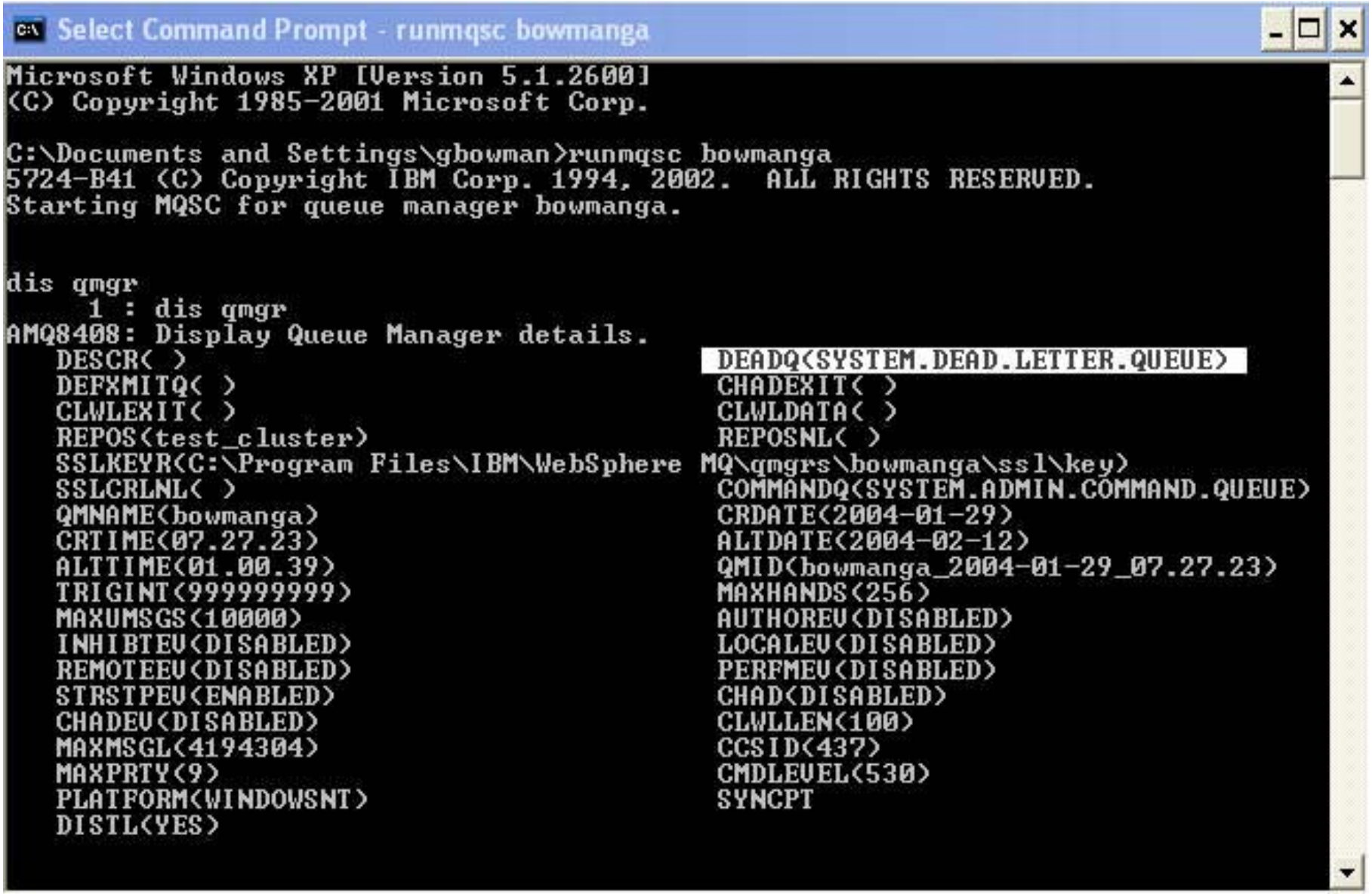
- When messages can not be delivered to their intended destination they may be sent to a Dead Letter Queue
- MQ provides an automated way of handling these Dead Letter messages
- Dead Letter Queue handler, invoked by runmqdlq command, is the default routine supplied by MQ

Agenda

- Dead Letter Queue
- Dead Letter Message and Header
- Dead Letter Queue Handler
- Sample DLQ Handler Program
- How to invoke the handler
- Rules Table
- Examples

Dead Letter Queue

- When messages can not be delivered to the intended queue, the messages may be delivered to a Dead Letter Queue (DLQ) if the queue manager has one defined.
- Can be any queue you define
- By default a queue manager does not have a DLQ but you should define one for every queue manager.


Select Command Prompt - runmqsc bowmanga
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\Documents and Settings\gbowman>runmqsc bowmanga
5724-B41 (C) Copyright IBM Corp. 1994, 2002. ALL RIGHTS RESERVED.
Starting MQSC for queue manager bowmanga.
dis qmgr
1 : dis qmgr
AMQ8408: Display Queue Manager details.
DESCR( )
DEFXMITQ( )
CLWLEXIT( )
REPOS(test_cluster)
SSLKEYR(C:\Program Files\IBM\WebSphere MQ\qmgrs\bowmanga\ssl\key)
SSLCRLNL( )
QMNAME(bowmanga)
CRTIME(07.27.23)
ALTTIME(01.00.39)
TRIGINT(999999999)
MAXUMSGS(10000)
INHIBTEV(DISABLED)
REMOTEEV(DISABLED)
STRSTPEV(ENABLED)
CHADEV(DISABLED)
MAXMSGL(4194304)
MAXPRTY(9)
PLATFORM(WINDOWSNT)
DISTL(YES)
DEADQ(SYSTEM.DEAD.LETTER.QUEUE)
CHADEXIT( )
CLWLDATA( )
REPOSNL( )
COMMANDQ(SYSTEM.ADMIN.COMMAND.QUEUE)
CRDATE(2004-01-29)
ALTDATE(2004-02-12)
QMID(bowmanga_2004-01-29_07.27.23)
MAXHANDS(256)
AUTHOREV(DISABLED)
LOCALEV(DISABLED)
PERFMEV(DISABLED)
CHAD(DISABLED)
CLWLLEN(100)
CCSID(437)
CMDLEVEL(530)
SYNCPT

Dead Letter Messages

- Messages may be placed on the DLQ by the queue manager, the channels (MCA) or by applications
- Messages should have a Dead Letter Header (DLH)
- Queue manager and MCA will generate the DLH
- Applications must create the DLH if they place messages on the DLQ

Chapter 7. MQDLH – Dead-letter header

The following table summarizes the fields in the structure.

Table 43. Fields in MQDLH

Field	Description	Page
StrucId	Structure identifier	89
Version	Structure version number	89
Reason	Reason message arrived on dead-letter queue	88
DestQName	Name of original destination queue	86
DestQMgrName	Name of original destination queue manager	85
Encoding	Numeric encoding of data that follows MQDLH	86
CodedCharSetId	Character set identifier of data that follows MQDLH	85
Format	Format name of data that follows MQDLH	86
PutApplType	Type of application that put message on dead-letter queue	87
PutApplName	Name of application that put message on dead-letter queue	86
PutDate	Date when message was put on dead-letter queue	87
PutTime	Time when message was put on dead-letter queue	87

MQRC and MQFB

- Reason code in the MQDLH will be a value from either the MQRC reason codes or MQFB feedback codes
- MQRC codes in WebSphere MQ Messages manual (Chapter 8)
- MQFB codes in WebSphere MQ Application Programming Reference (Appendix B – MQ Constants)

Windows sample (part 1)

```
AMQSBCG0 - starts here
**********************

 MQOPEN - 'SYSTEM.DEAD.LETTER.QUEUE'


 MQGET of message number 1
****Message descriptor****

  StrucId  : 'MD  '  Version : 2
  Report   : 0  MsgType : 8
  Expiry   : -1  Feedback : 0
  Encoding : 546  CodedCharSetId : 437
  Format : 'MQDEAD  '
  Priority : 0  Persistence : 1
  MsgId : X'414D51204741422020202020202020202042245B5620001A02'
  CorrelId : X'000000000000000000000000000000000000000000000000'
  BackoutCount : 0
  ReplyToQ       : '                                                '
  ReplyToQMgr    : 'GAB                                             '
  ** Identity Context
  UserIdentifier : 'mqm         '
  AccountingToken :
   X'160105150000000030D6D3BAB5CDD6E5F7E9E53EB0300000000000000000000000B'
  ApplIdentityData : '                                '
  ** Origin Context
  PutApplType    : '11'
  PutApplName    : 'WebSphere MQ\bin\amqsputc.exe'
```

Windows sample (part 2)

```
  PutDate  : '20050301'    PutTime  : '13244733'
  ApplOriginData : '    '

  GroupId : X'000000000000000000000000000000000000000000000000'
  MsgSeqNumber   : '1'
  Offset         : '0'
  MsgFlags       : '0'
  OriginalLength : '-1'

****   Message      ****

 length - 184 bytes

00000000:  444C 4820 0100 0000 0508 0000 7363 6F6F 'DLH ........scoo'
00000010:  6220 2020 2020 2020 2020 2020 2020 2020 'b               '
00000020:  2020 2020 2020 2020 2020 2020 2020 2020 '                '
00000030:  2020 2020 2020 2020 2020 2020 626F 776D '            bowm'
00000040:  616E 6761 2020 2020 2020 2020 2020 2020 'anga            '
00000050:  2020 2020 2020 2020 2020 2020 2020 2020 '                '
00000060:  2020 2020 2020 2020 2020 2020 2202 0000 '              "...'
00000070:  B501 0000 4D51 5354 5220 2020 0B00 0000 'µ...MQSTR    ....'
00000080:  6562 5370 6865 7265 204D 515C 6269 6E5C 'ebSphere MQ\bin\'
00000090:  414D 5152 4D50 5041 2E45 5845 3230 3035 'AMQRMPPA.EXE2005'
000000A0:  3033 3031 3132 3438 3036 3532 7465 7374 '0301124806 52test'
000000B0:  206D 6573 7361 6765                     ' message        '

 No more messages
 MQCLOSE
 MQDISC
```

AIX Sample (part 1)

```
AMQSBCG0 - starts here
**********************
 MQOPEN - 'SYSTEM.DEAD.LETTER.QUEUE'

 MQGET of message number 1
****Message descriptor****
StrucId : 'MD  '  Version : 2
Report   : 0  MsgType : 8
Expiry   : -1   Feedback : 0
Encoding : 273   CodedCharSetId : 819
Format : 'MQDEAD  '
Priority : 0  Persistence : 1
MsgId : X'414D5120626F776D616E676120202020524E244220001501'
CorrelId : X'000000000000000000000000000000000000000000000000'
BackoutCount : 0
ReplyToQ      : '                                '
ReplyToQMgr   : 'bowmanga                        '
** Identity Context
UserIdentifier : 'gbowman     '
AccountingToken :    X'160105150000000030D6D3BAB5CDD6E5F7E9E53EB0300000000000000000000000B'
AppIdentityData : '                            '
** Origin Context
PutApplType    : '11'
PutApplName    : 'WebSphere MQ\bin\amqsputc.exe'
```

AIX Sample (part 2)

```
PutDate  : '20050301'
PutTime  : '12044424'
ApplOriginData : '    '

GroupId : X'000000000000000000000000000000000000000000000000'
MsgSeqNumber   : '1'
Offset         : '0'
MsgFlags       : '0'
OriginalLength : '-1'

****   Message      ****

 length - 194 bytes

00000000:  444C 4820 0000 0001 0000 0805 5445 5354 'DLH ........TEST'
00000010:  2020 2020 2020 2020 2020 2020 2020 2020 '                '
00000020:  2020 2020 2020 2020 2020 2020 2020 2020 '                '
00000030:  2020 2020 2020 2020 2020 2020 4741 4220 '            GAB '
00000040:  2020 2020 2020 2020 2020 2020 2020 2020 '                '
00000050:  2020 2020 2020 2020 2020 2020 2020 2020 '                '
00000060:  2020 2020 2020 2020 2020 2020 0000 0222 '              ..."'
00000070:  0000 01B5 4D51 5354 5220 2020 0000 0006 '....µMQSTR    ....'
00000080:  616D 7172 6D70 7061 2020 2020 2020 2020 'amqrmppa        '
00000090:  2020 2020 2020 2020 2020 2020 3230 3035 '            2005'
000000A0:  3033 3031 3132 3439 3531 3735 7465 7374 '0301124951 75test'
000000B0:  206F 6620 4D65 7373 6167 6520 746F 2044 ' of Message to D'
000000C0:  4C51                                    'LQ              '
```

DLQ Handler

- Once a message arrives on the DLQ you can automate the handling of that message using the DLQ handler program
- You can have the handler running and waiting for messages to arrive on the DLQ or you can set up the DLQ to trigger the start of the handler program

Starting the DLQ Handler

- You can start the DLQ handler using the runmqdlq command.
- Documented in the WebSphere MQ System Administration Guide
- Example: runmqdlq DLQ QMGR < rules.tb
  where DLQ = name of your dead letter q
  QMGR = queue manager name and
  rules.tb is a file with your rules table

Sample Program

- MQ supplies a sample program called amqsdlq
- Provides similar function as runmqdlq
- Allows users to customize the way messages are handled from the DLQ, especially useful if you have applications placing messages directly onto DLQ

Rules Table

- Defines how the DLQ handler will process the messages on the DLQ
- Two types of entries in the rules table
  - Control Data
  - Rules
- runmqdlq can take rules table from command line (stdin) or redirect from file

Control Data

- Control Data is optional (default queue manager and it's DLQ are assumed)
  - INPUTQ – name of the DLQ from which to process messages
  - INPUTQM – name of queue manager
  - RETRYINT – retry interval (in seconds) of how long to wait before trying to process a message that could not be processed before
  - WAIT – YES|NO for messages to arrive

RULES (part 1)

- Pattern Keywords – allow you to match only certain messages on the DLQ
  - Ex: REASON – match only messages with a specified reason code
  - Ex: DESTQ – match only messages which were destined for a specified queue

Rules (part 2)

- Action Keywords describe how a matching message is processed
  - ACTION(DISCARD|IGNORE|RETRY|FWD)
  - FWDQ – name of queue where message is forwarded
  - FWDQM – name of qmgr where message is forwarded
  - HEADER(YES|NO) – do messages have DLH header
  - PUTAUT(DEF|CTX) – userid of runmqdlq program or userid from the MD of the message on the DLQ
  - RETRY – number of times to retry each matching rule

Rules Table Conventions (part 1)

- A rules table must contain at least one rule.
- Keywords can occur in any order.
- A keyword can be included only once in any rule.
- Keywords are not case-sensitive.
- A keyword and its parameter value must be separated from other keywords by at least one blank or comma.

Rules Table Conventions (part 2)

- There can be any number of blanks at the beginning or end of a rule, and between keywords, punctuation, and values.
- Each rule must begin on a new line.
- For reasons of portability, the significant length of a line must not be greater than 72 characters.
- See Chapter 12 of WebSphere MQ System Administration guide for complete list of rules and conventions.

Sample rules table

```
********************************************
* An example rules table for the runmqdlq command *
********************************************
* Control data entry
* ------------------
* If parameters are not supplied on the runmqdlq command use
* SYSTEM.DEAD.LETTER.QUEUE as the input queue, use bowmanga as the
* queue manager and set retry interval to 20 seconds
inputq('SYSTEM.DEAD.LETTER.QUEUE') inputqm('bowmanga') retryint(20)
*
* Rules entries
* -----
* Include a rule with ACTION (RETRY) first to try to deliver the
* message to the intended destination. If a message is placed on the
* DLQ because its destination queue is full, attempt to forward the
* message to its destination queue. Make 5 attempts at approximately
* 20 second intervals (the default value for RETRYINT). Also include
* the "+" just to show how to wrap a command over to the next line.
REASON(MQRC_Q_FULL) ACTION(RETRY) +
RETRY(5)
* If a message is placed on the DLQ due to put inhibited, attempt to
* forward the message to its destination queue. Make 5 attempts at
* approximately 20 second intervals (the default value for RETRYINT).
REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)
* Include a rule to handle messages which do not match any of the
* patterns specified above. Send the messages to another queue named
* DEADQ where they can be handled later.
ACTION(FWD) FWDQ('DEADQ')
```

Runmqdlq examples

- If you use the DLQ handler without redirecting stdin from a file (the rules table), the DLQ handler reads its input from the keyboard.
- In WebSphere MQ for AIX, Solaris, HP-UX, and Linux, the DLQ handler does not start to process the named queue until it receives an end_of_file (Ctrl+D) character.
- In WebSphere MQ for Windows it does not start to process the named queue until you press the following sequence of keys: Ctrl+Z, Enter, Ctrl+Z, Enter

Runmqdlq from command line

```
C:\>runmqdlq SYSTEM.DEAD.LETTER.QUEUE bowmanga
WAIT(YES) RETRYINT(20)
REASON(MQRC_Q_FULL) ACTION(RETRY) +
RETRY(5)
REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)
REASON(*) ACTION(FWD) FWDQ('DEADQ')
^Z
^Z
2005-04-08   02.20.09 AMQ8708: Dead-letter queue handler started to process
     INPUTQ(SYSTEM.DEAD.LETTER.QUEUE).
```

Runmqdlq example with bad input

```
C:\>runmqdlq SYSTEM.DEAD.LETTER.QUEUE bowmanga
REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)
WAIT(YES)
^Z
^Z

syntax error on line 2
...WAIT
2005-04-08   02.43.10 AMQ8758: 1 errors detected in input to runmqdlq.
2005-04-08   02.43.10 AMQ8709: Dead-letter queue handler ending.
```

Runmqdlq example on AIX

```
aemaix3:bowmanga 40% runmqdlq DEADQ GAB
WAIT(YES) RETRYINT(5)
REASON(*) ACTION(RETRY) RETRY(5)
^D
2005-04-08   03.46.34 AMQ8708: Dead-letter queue handler started to
     process INPUTQ(DEADQ).
```

Summary

- If messages go to your Dead Letter Queue you can automate the handling of those messages using runmqdlq.
- You can create a rules table to define how DLQ messages should be handled on an ongoing basis or you can enter input from command line for one time run.

Where to Get More Information

- WebSphere MQ System Administration Guide SC34-6088-02
  - Chapter 12
- WebSphere MQ Application Programming Reference SC34-6062-03
- MustGather: - DLQ problem – technote 1176930 www.ibm.com/support/search.wss?rs=171&tc=SSFKSJ%2BSSWHKB%2BSSFKSJ&q=1176930

Additional WebSphere Product Resources

- Discover the latest trends in WebSphere Technology and implementation, participate in technically-focused briefings, webcasts and podcasts at: www.ibm.com/developerworks/websphere/community/
- Learn about other upcoming webcasts, conferences and events: www.ibm.com/software/websphere/events_1.html
- Join the Global WebSphere User Group Community: www.websphere.org
- Access key product show-me demos and tutorials by visiting IBM Education Assistant: ibm.com/software/info/education/assistant
- Learn about the Electronic Service Request (ESR) tool for submitting problems electronically: www.ibm.com/software/support/viewlet/probsub/ESR_Overview_viewlet_swf.html
- Sign up to receive weekly technical My support emails: www.ibm.com/software/support/einfo.html
- Attend WebSphere Technical Exchange conferences or Transaction and Messaging conference: http://www-304.ibm.com/jct03001c/services/learning/ites.wss/us/en?pageType=page&c=a0011317

Questions and Answers