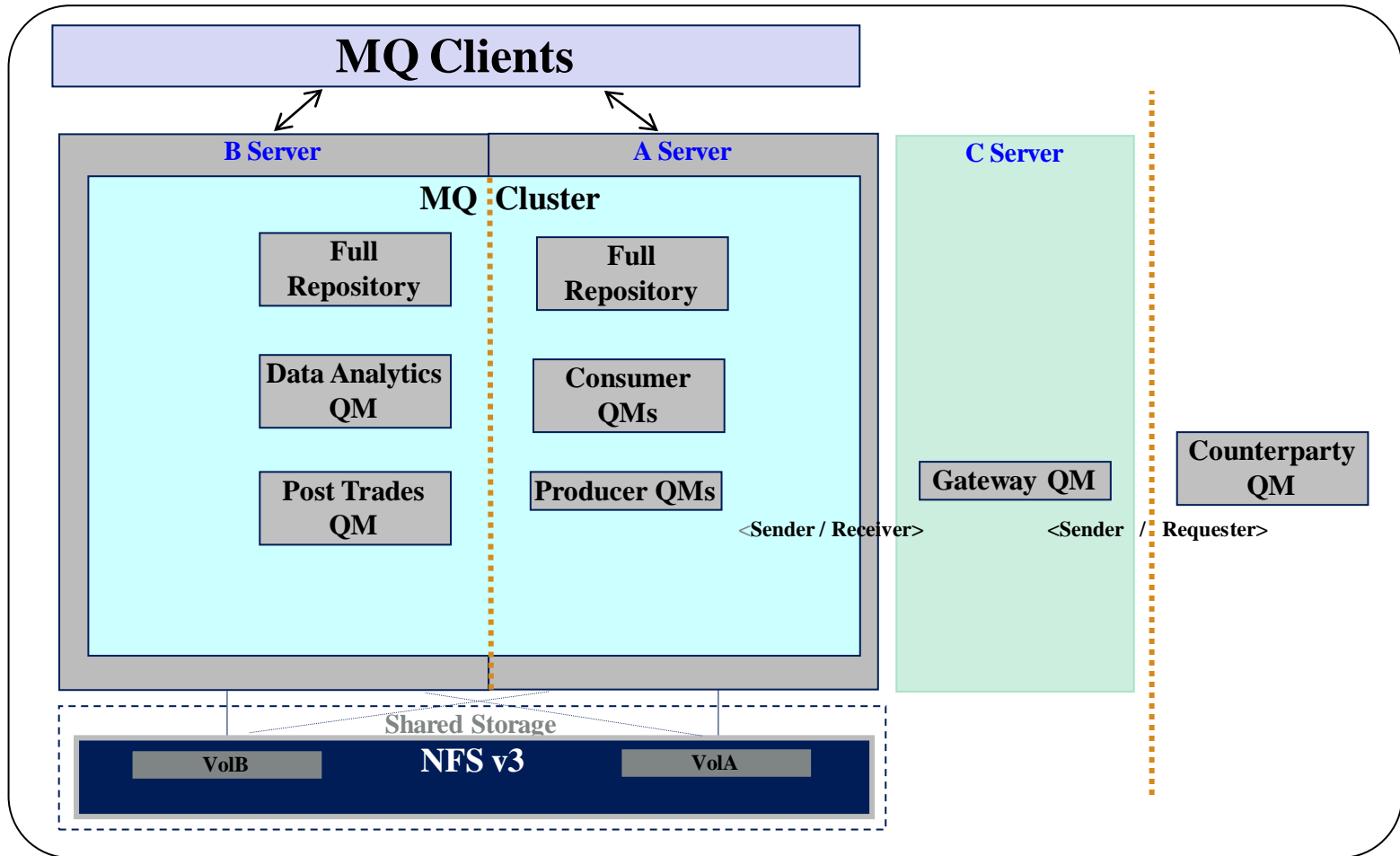


# ***MQ V7.n Client Upgrade Notes “Things That Stopped Working”***

Derek W. Hornby  
WebSphere MQ Administrator  
NB Group, LLC.

# MQ Server Architecture



# MQClient Setup: Channel Definition

<b>General</b>	<b>General</b>
Extended	Channel name: DEREK.CLIENT
MCA	Type: Server-connection
Exits	Description:
SSL	Transmission protocol: TCP
Statistics	Overall channel status: Inactive
	<b>Extended</b>
General	Maximum message length (bytes): 4194304
<b>Extended</b>	Heartbeat interval: 300
MCA	Maximum instances: 999999999
Exits	Maximum instances per client: 999999999
SSL	Message compression: None
Statistics	<input checked="" type="radio"/> Auto
	Keep alive interval (seconds): <input type="radio"/> 0
	Header compression: None
	Disconnect interval (seconds): 0
	Sharing conversations: 10

# MQClient Setup: Channel Definition (Contd.)

- General
- Extended
- MCA
- Exits
- General
- Extended
- MCA
- Exits
- SSL
- Statistics

**MCA**

MCA user ID:

---

**Exits**

Send exit name:

Send exit user data:

Receive exit name:

Receive exit user data:

Security exit name:

Security exit user data:



# MQClient Setup: MQAUSX ini File

```
# EXX01 NBXGXX1A
NoAuth=Y
License=10A9-qhEy-TN6X9999
LogMode=V
LogFile=/var/mqm/seccode/log/exx01.log
RotateLogDaily=Y
BackupLogFileCount=9
EventQueueName=SYSTEM.ADMIN.CHANNEL.EVENT
UseServerName=N
AllowUserAlterServerName=N
UseAuthOrder=N
UseLDAP=N
UseLDAPLoadBalance=N
LDAPTimeOut=10
UseLDAPSSL=N
UseLDAPSSLCert=N
SSLCertFileType=DER
UseLoginDNPrefix=N
UseANRforLDAP=N
ExtractUserIDFromANR=N
UseANRPrefix=N
UseANRPostfix=N
UseANRDelimiter=N
UseLDAPSearch=N
UseFBA=N
UseProxy=Y
ProxyFile=/var/mqm/seccode/proxxy.txt
Allowmqm=N
AllowBlankUserID=N
UseMCAUser=N
AllowMQCSPAuth=Y
UseAllowUserID=Y
AllowUserID=dhornby;svcdrengjbad;
UseRejectUserID=N
UseAllowIP=Y
AllowIP=10.10.8.88;10.10.9.99;
UseRejectIP=N
UseMCC=N
MCCRedoMinutes=720
MCCRedoCount=1000
MCCEventWarnLevel=80
MCCGetTimeOut=3
UseAllowADName=N
UseRejectADName=N
```

# MQClient Setup: CHLAUTH

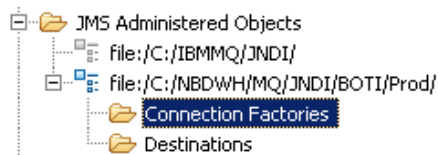
## CHLAUTH Example:

Channel authentication records:

Channel profile	Type	Peer name	Client user ID	Remote queue manager	Address	User source	MCA user ID
 *	Block User List						
 DEREK.SVRCONN	User Map		svcdrengjbad		10.99.99.99	Map	mqm

# MQClient Setup: Bindings

- JNDI is File Based (originally LDAP)



- Standard Connection Factory parameters

MQ Client Configuration dialog. The 'Connection list' field contains 'mqm-a(1416),mqm-b(1416)'. The left sidebar shows 'Reconnection' selected. The 'Reconnection' tab is active, showing 'Options' set to 'Reconnect' and 'Timeout' set to '1800'.

- Standard Destination parameters

- JMS to JMS

MQ Client Configuration dialog for JMS to JMS mapping. 'Message Body' is set to 'JMS' and 'Target client' is set to 'JMS'.

- JMS to MQ

MQ Client Configuration dialog for JMS to MQ mapping. 'Message Body' is set to 'Unspecified' and 'Target client' is set to 'MQ'.

- MQ to JMS

MQ Client Configuration dialog for MQ to JMS mapping. 'Message Body' is set to 'MQ' and 'Target client' is set to 'JMS'.

# MQClient Setup: Bindings – Connection Factory

**FDS\_CF - Properties**

**General**

Name: FDS\_CF  
Description:   
Class name: MQQueueConnectionFactory  
Messaging provider: WebSphere MQ  
Transport: Client  
Provider version: \* 7

Shared conversation allowed: Yes

**Connection**

Base queue manager: NBXGG1AX [Select...]  
Connection list: mqm-a(1416),mqm-b(1416)  
Coded character set ID: 819  
Local address:   
Connect options: Standard

**Reconnection**

Options: Reconnect  
Timeout: 1800

**Channels**

Channel: DEREK.CLIENT

Navigation tabs: General, Connection, Reconnection, Channels, SSL, Exits, Temporary queues, Extended



# MQClient Setup: Bindings - Destination

<ul style="list-style-type: none"><li>General</li><li>Message handling</li><li>Producers</li><li>Consumers</li><li>Extended</li></ul>	<h3>General</h3> <p>Name: <input type="text" value="FDS_NOTIFY_DATA"/></p> <p>Description: <input type="text"/></p> <p>Class name: <input type="text" value="MQQueue"/></p> <p>Messaging provider: <input type="text" value="WebSphere MQ and Real-time"/></p> <p>Queue manager: <input type="text" value="NBXGGX1X"/></p> <p>Queue: * <input type="text" value="FDS.NOTIFY.DATA"/></p>
<ul style="list-style-type: none"><li>General</li><li>Message handling</li><li>Producers</li><li>Consumers</li><li>Extended</li></ul>	<h3>Message handling</h3> <p>Expiry: <input checked="" type="radio"/> Application <input type="radio"/> 0</p> <p>Persistence: <input type="text" value="Application"/></p> <p>Priority: <input checked="" type="radio"/> Application <input type="radio"/> 0</p> <p>MQMD Message Context: <input type="text" value="Default"/></p> <p>MQMD Write Enabled: <input type="text" value="No"/></p> <p>MQMD Read Enabled: <input type="text" value="No"/></p> <p>Message Body: <input type="text" value="Unspecified"/></p>

# MQClient Setup: Bindings – Destination (Contd.)

General  
Message handling  
Producers  
Consumers  
Extended

### Extended

Coded character set ID: 1208

Encoding

Native encoding

Integer encoding: Normal

Decimal encoding: Normal

Floating point encoding: Normal

Mnemonic: NATIVE

Value: 273

Fail if quiescing: Yes

Target client: JMS

ReplyTo destination style: Default

Receive conversion: Client message

Receive CCSID: 1208

# "TTSW": Previously Fixed

- **MQ V7.0.1.0 MQClient XMS DotNet dll "faulty" – Fixed in V7.0.1.2 Client**
  - corrupted all messages over 32k
- **MQExplorer File based JNDI issues: – Fixed in V7.0.1.2 Client**
  - Bindings files created with MQExplorer not readable by XMS DotNet apps (consistent "2538")
  - Bindings files created with JMSAdmin cause error with JMS apps (intermittent "2538")
  - Emailing Bindings files and Certificates with Exchange/Outlook causes file corruption
- **MQClient V7.0.1.0 fails with MQServer V7.0.1.7 – Fixed in applications**
  - creates "zombie" writer apps if MQCC / MQRC not handled gracefully

# “TTSW”: “Long” Service Accounts

## MQ Server CHLAUTH “12 Character” Notes

The QMGR parameter **CHLAUTH(ENABLED)** is the default setting for Queue Managers created under V7.1 and V7.5. , which means that MQ is “locked down” OOTB

**CHLAUTH rules** can then be set which allow access, by UserId and or IP Address

- MQ running on Linux / Unix: if a UserId is passed that is **longer than 12 characters**, then **access is denied**, with a 2035 and no further explanation in the logs!
- For example, unlike Capitalware’s MQAUSX product, you cannot use a “truncated” version of the UserId to authenticate via the CHLAUTH rules!
- We see this problem with our apps that run under **IIS** (“**network service**”) and **JBOSS** (where our JBOSS Admins used to allow very long service accounts.
- To use CHLAUTH, the Service Accounts must change, or the apps must send their own UserId string at connection time! – we use both methods

# "TTSW": "Long" Service Accounts (Contd.)

User Id is "svcdrengjbadadmin"

**MQAUSX** Example: using MQAUSX ini file in SVRCONN "Exits" parms:

Security exit name:	mqausx(SecExit)
Security exit user data:	/var/mqm/testdir/DEREKT.ini

**Failed pattern:** AllowUserID=svcdrengjbadadmin;

2013/09/09 15:02:40 MQAUSX #01237 W: Regular expression **failed** on early text termination : UserID is **svcdrengjbad and pattern is svcdrengjbadadmin**

**Success Pattern:** AllowUserID=svcdrengjbad;

2013/09/09 15:24:01 MQAUSX #01207 I:

**Regular expression**

**matched UserID of svcdrengjbad to pattern: svcdrengjbad**

2013/09/09 15:24:01 MQAUSX #00180 I: FileExist() **success** :

filename=/var/mqm/mqausx/derekp.txt

2013/09/09 15:24:01 MQAUSX #00272 I: Looking up UserID: svcdrengjbad

2013/09/09 15:24:01 MQAUSX #00388 I: Using Default Proxy UserID of: derek



# "TTSW": "Long" Service Accounts (Contd.)

User Id is "svcdrengjbadmin"

**CHLAUTH** Example:

Channel authentication records: Enabled

"allowed" Id. "svcdrengjbadmin" in CHLAUTH parms:  
(Only 12 Chars allowed "svcdrengjbad")

Channel profile	Type	Peer name	Client user ID	Remote queue manager	Address	User source	MCA user ID
 *	Block User List						
 DEREK.SVRCONN	User Map		svcdrengjbad		10.99.99.99	Map	mqm

**Failed:**

09/09/2013 06:20:59 PM - Process(29802.74) User(mqm) Program(amqrmppa)  
Host(pidlwmq001.nb.com) Installation(Installation1)  
VRMF(7.5.0.0) QMgr(NBHR01AD)

AMQ9557: Queue Manager User ID initialization failed.

EXPLANATION:

The call to initialize the User ID failed with CompCode 2 and Reason 2035.

ACTION: Correct the error and try again.

# "TTSW": "Null" User Ids

## "Null Id" problem

Java programs using MQClient V7.0.1.0, V7.0.1.1 and V7.0.1.2 on Linux servers automatically passed the Id. under which they were executing to MQ during the connection process.

Under later MQClient versions, the Id. was no longer passed (null id was passed to MQ), which caused authentication via Security Exits or CHLAUTH settings to fail

### Technote:

#### For WMQ JMS V7.0.1.3 and above:

the UserId which will be used to connect to the queue manager is:

1. The MCA UserId specified on the server connection channel if present
2. If 1 is not specified, then the UserId specified as an argument to the createConnection() call in the WMQ JMS Client application code.
3. If neither 1 nor 2 is specified, then:

**(a) if `com.ibm.mq.jms.ForceUserID = true`, then the UserId with which the JVM was started will be used.**

**(b) if `com.ibm.mq.jms.ForceUserID = false` (or not specified), then a blank userid will be used.**

# "TTSW": "Null" User Ids (Contd.)

**Adding this parameter to the JVM arguments will force a UserId to be passed:**

**Java:**

```
java -Dcom.ibm.mq.jms.ForceUserID=true class
```

**JBOSS:**

append the parameters to JAVA\_OPTS:

```
export JAVA_OPTS="${JOPTS} ${JAVA_OPTS}"  
appendJavaProp "com.ibm.mq.jms.ForceUserID" "true"  
appendJavaProp "com.ibm.msg.client.wmq.receiveConversionCCSID" "1208"  
appendJavaProp () {  
    JOPTS=" ${JOPTS} -D$1=$2 "  
}
```



# "TTSW": Data Conversion

## Data Conversion Problem

MQ data originating on Windows is no longer being automatically converted when read by a JMS application on a Linux server - **changed from V7.0.1.0 → V7.0.1.8+**

Example: when a JBOSS JMS app on Linux reads a JMS message created on Windows, and writes that message to another local queue, the write fails with this error:

```
ERROR [jmsSingletonNonXAQueueConnector.dispatcher.1] mule.AbstractExceptionListener
(AbstractExceptionListener.java:394) - Caught exception in Exception Strategy: JMSCMQ1006: The
value for 'JMS_IBM_Character_Set':'UnicodeLittle' is not valid.
com.ibm.msg.client.jms.DetailedJMSEException: JMSCMQ1006: The value for
'JMS_IBM_Character_Set':'UnicodeLittle' is not valid. The value 'UnicodeLittle' for property
'JMS_IBM_Character_Set' is not correct. Check the linked WebSphere MQ exception reason and
completion code.
```

# "TTSW": Data Conversion (Contd.)

**IC72897** – changed between V7.0.1.2 → V7.0.1.3+

JMS applications upgraded from V6 may have encountered problems associated with data conversion, either the received message bytes were different in V7 to those seen in V6, or the queue manager was unable to convert the messages in V7 issuing error messages.

This APAR switches the default JMS message conversion behavior back to that of V6. the default for reading JMS Messages of "Get with MQGMO\_CONVERT" has been changed in MQClient versions > V7.0.1.2 so that conversion is no longer performed by default (the MQGMO\_CONVERT has been dropped).

**Adding this parameter to the JVM arguments will force conversion to take place (1208 is the preferred CCSID for our X86 Linux servers):**

**Java:**

```
java -Dcom.ibm.msg.client.wmq.receiveConversionCCSID=1208 class
```

**JBOSS:**

```
export JAVA_OPTS="${JOPTS} ${JAVA_OPTS}"  
appendJavaProp "com.ibm.mq.jms.ForceUserID" "true"  
appendJavaProp "com.ibm.msg.client.wmq.receiveConversionCCSID" "1208"  
appendJavaProp () {  
    JOPTS=" ${JOPTS} -D$1=$2 "  
}
```

## Java "PMO" Put problem

We have several Java applications that run normally under MQClient V7.0.1.0 JARs that fail with this error when using MQClient V7.0.1.8

```
Exception in thread "Timer-0" java.lang.StackOverflowError
  at com.ibm.mq.jmqi.internal.AbstractMqiStructure.<init>(AbstractMqiStructure.java:54)
  at com.ibm.mq.jmqi.MQPMO.<init>(MQPMO.java:141)
  at com.ibm.mq.jmqi.MQPMO.clone(MQPMO.java:962)
  at com.ibm.mq.jmqi.remote.internal.RemoteFAP.jmqiPutMessageWithProps(RemoteFAP.java:7299)
```

## IBM Answer:

### Problem summary

\*\* USERS AFFECTED: This issue affects users of the WebSphere MQ V7.0.1.8 classes for JMS who have enabled **automatic client reconnection** for either the specific connection in use or the entire Java Virtual Machine (JVM).

The fix is targeted for delivery in the following PTFs: **v7.0 Platform Fix Pack 7.0.1.9**

# "TTSW": SHARECNV(10) Problem

**IC61153: WHEN SHARING CONVERSATIONS IS ENABLED, WEBSHERE MQ V7 JMS APPLICATION THREADS HANG IF THERE ARE CONNECTION PROBLEMS**

**IZ69682: WMQ V7: CONVERSATION SHARING IS NOT HAPPENING BETWEEN A V7 QUEUE MANAGER IN HP-UX AND A MULTI-THREADED JAVA/JMS MQ CLIENT V7**

**IZ85904: V7 WEBSHERE MQ JAVA/JMS CLIENTS: WHEN CONVERSATION SHARING IS BEING USED THERE IS A DELAY IN 2059 CONNECTION TIMEOUT REPORTING**

**IV47311: WEBSHERE MQ CLASSES FOR JAVA/JMS ATTEMPT TO CREATE MORE SHARED CONVERSATIONS THAN THE CONFIGURED SHARECNV LIMIT ON A CONNECTION**

**IC59462: WebSphere MQ v7 JMS applications may create too many, or too few shared conversations per connection.**

**IZ78516: CONSIDERABLE DELAY OBSERVED DURING MQDISC CALL WHEN A V7.0 MQ CLIENT CONNECTS TO A V7.0 QUEUE MANAGER WHEN USING NON ZERO SHRCNV**

**IZ65557: WMQ 7 JAVA APPLICATION EXPERIENCES A HANG WHILE CONNECTING TO QUEUE MANAGER USING SHARED CONNECTIONS.**

**IC66174: WMQ V7.0 CLIENT CONVERSATION SHARING MEMORY ACCESS VIOLATION UNDER RESOURCE CONSTRAINT CONDITIONS**

# “TTSW”: SHARECNV(10) Problem

## MQI client: Default behavior of client-connection and server-connection

From version 7.0, the default for client and server connections is to share an MQI channel. Each channel is defined with a default of 10 threads to run up to 10 client conversations per channel instance. Before version 7.0, each conversation was allocated to a different channel instance. The change might cause migration problems for existing client applications.

You can restore **version 6.0** behavior of a server or client connection channel by setting the channel attribute, **SHARECNV**, to **0**.

If you set **SHARECNV to 1**, rather than **0**, each conversation is allocated a separate channel instance. However, the channel behaves like a new channel.  
→ **V7 Behavior but without sharing**

For example, heartbeats flow in each direction at any time, and the channel supports the following features:

- Administrator stop-quiesce
- Unrestricted heartbeats
- Read-ahead
- Asynchronous-consume by client applications

# "TTSW": Last Set of Test Results

**Server:** MQ V7.1.0.0 - RHEL6 x64

**Client:** MQ V7.1.0.0 - RHEL6 x64 / Win Server 2008 R2 x64

**Tests:** Get Messages / Put messages

- RC 2195 – "Unexpected Error"
- RC 2538 – "Host Not Available"

**Resolution:** Client Versions rolled back to V7.0.1.7 (Linux) and V7.0.1.8) (Win)

**Server:** MQ V7.5.0.0 - RHEL6 x64

**Client:** MQ V7.5.0.0 - RHEL6 x64 / Win Server 2008 R2 x64

**Tests:** Get Messages / Put messages

- Windows C - PASS
- Windows DotNet XMS - PASS
- Windows Java - PASS
- Windows JMS - PASS
- Linux C - PASS
- Linux Java - PASS
- Linux JMS - PASS

Thank You For Attending

## **Wrap Up and Questions**