Using CCDT in an Activation Specification for a Queue Manager Group of separate queue managers in WAS V7, V8.0 and V8.5

IBM Techdoc: 7035714
http://www.ibm.com/support/docview.wss?rs=171&uid=swg27035714

Date last updated: 26-Sep-2013

Angel Rivera – rivera@us.ibm.com
IBM WebSphere MQ Support

+++ Objective

This technical document describes the steps for configuring an Activation Specification in WebSphere Application Server V7, V8 and V8.5 that uses a Client Channel Definition Table (CCDT) file for a Queue Manager Group in WebSphere MQ V7 that use separate queue managers.

The last section provides information on customization of the MDB using IBM Rational Application Development for EJB 3.

If you want to use a Connection Factory, consult the following related techdoc:
http://www.ibm.com/support/docview.wss?rs=171&uid=swg27035452
Using CCDT in a Connection Factory for a Queue Manager Group of separate queue managers in WAS V7

This document explains the usage of the sample MDB (SampleMDBEJBEAR) which is available from the following techdoc, which also has the steps on how to create the JMS administrative objects and how to deploy the MDB.

Using WebSphere MQ V7 as JMS Provider for WebSphere Application Server V7
http://www.ibm.com/support/docview.wss?rs=171&uid=swg27016505

For the case of a Queue Manager Group that is composed of a multi-instance queue manager, see the techdoc:

http://www.ibm.com/support/docview.wss?rs=171&uid=swg27020701
Using a CCDT file to connect to WebSphere MQ multi-instance queue managers from WebSphere Application Server V7

Note: XA is NOT supported for separate queue managers using CCDT.

++
+++ Chapter 1: Configuration, creation of JMS administrative objects, deploy MDB
++

++ There are 2 separate queue managers in the Queue Manager Group:

Queue Manager 1:
  Name: QM_VER
  Hostname: veracruz.x
  Port: 1414
  Channel: TEST.TO.QM_VER
  Queue: Q_MDB
  Version: 7.0.1.10

Queue Manager 2:
  Name: QM_MIG
  Hostname: veracruz.x
  Port: 1421
  Channel: TEST.TO.QM_MIG
  Queue: Q_MDB
  Version: 7.5.0.1

The WAS Server (server1) is located in:
  Hostname: veracruz.x

The sample MDB code has been downloaded and it is available for deployment.

++ Creation of the CCDT

The CCDT is created in the queue manager QM_VER.

Notice that the queue manager group is the value for the attribute QMNAME:
  QMGRP

Thus, to specify it, it is necessary to use a leading asterisk when using it:
  *QMGRP

Invoke the MQ administration tool for the queue manager QM_VER

runmqsc QM_VER

DEFINE CHANNEL(TEST.TO.QM_VER) CHLTYPE(SVRCONN) TRPTYPE(TCP) REPLACE

DEFINE CHANNEL(TEST.TO.QM_MIG) CHLTYPE(CLNTCONN) TRPTYPE(TCP) CONNAME('veracruz.x.(1421)') QMNAME(QMGRP) REPLACE

DEFINE CHANNEL(TEST.TO.QM_VER) CHLTYPE(CLNTCONN) TRPTYPE(TCP) CONNAME('veracruz.x.(1414)') QMNAME(QMGRP) REPLACE

END

<u>Invoke the MQ administration tool for the queue manager QM_MIG</u>

In the other queue manager (QM_MIG) create the corresponding server-connection channel:

runmqsc QM_MIG

DEFINE CHANNEL(TEST.TO.QM_MIG) CHLTYPE(SVRCONN) TRPTYPE(TCP) REPLACE

END

+ Copy the CCDT file from QM_VER into a directory that is not associated with the queue manager in order to isolate disruptions from further updates to the client-connection channels in that queue manager. In this case it is/var/mqm
  cp /var/mqm/QM_VER/@ipcc/AMQCLCHL.TAB /var/mqm/AMQCLCHL.TAB

The specification for WAS for the CCDT will be using this generic place.
  /var/mqm/AMQCLCHL.TAB

++ Use the WAS Admin console to create JMS resources

This scenario is based on a single standalone WAS server (no other cells, no other nodes), thus, the scope used here is for the "server".

+ Destination: Queue

JMS messaging provider: WebSphere MQ Messaging Provider
Name: SampleMDBQueueNoQmgr
JNDI Name: jms/SampleMDBQueueNoQmgr
Queue Name in MQ: Q_MDB
Queue Manager: (blank)
Note: It is ESSENTIAL that the Queue Manager field must be left with blank. If you look at the corresponding entry for this connection factory in the resources.xml file, you will see that the attribute is called:

baseQueueManagerName

+ Activation Specification for a Queue:

JMS messaging provider: WebSphere MQ Messaging Provider
Name: SampleMDBQueueActSpecCCDT
JNDI Name: jms/SampleMDBQueueActSpecCCDT
Destination JNDI name: jms/SampleMDBQueueNoQmgr
Destination Type: Queue
Use a client channel definition table:
  CCDT: file:///var/mqm/AMQCLCHL.TAB
  Queue Manager: *QMGRP

CCDT URL conventions
The CCDT URL specifies the file location and is required by an MQ binding when the
connection mode is selected to use a CCDT. The format of the CCDT URL depends on
the operating system where the CCDT is located.
The file protocols can use either three slashes ("///") or one slash ("/").
- Windows:
  file:///C:/directoryPath/AMQCLCHL.TAB
or
  file:/C:/directoryPath/AMQCLCHL.TAB
- Unix or Linux:
  file:///directoryPath/AMQCLCHL.TAB
or
  file:/directoryPath/AMQCLCHL.TAB

Notice that you will get the following warning regarding the Queue Manager field:
The value *QMGRP for field Queue manager used in combination with a client channel
definition table might prevent recovery of global transactions. See the information
center for further detail.

Continue with the creation of the Activation Specification.

Click on Save.

+ Restart the server

You will need to stop and restart the WAS server in order for the new JMS resources to be available

As root issue:
  stopServer.sh server1
  startServer.sh server1

+ Deploy the MDB

From the WebSphere Application Server Administrative Console, in the left panel, select:
Applications > New Application

Then from the center window "New Application" select:
New Enterprise Application

You will see a dialog "Preparing for the application installation".
Specify the location where you have downloaded the ear file: SampleMDBEJB.ear

The sample MDB has a default binding that is associated with a Listener Port. We need to override it now.
Do not select FastPath, instead select:
(*) Detailed - Show all installation options and parameters

In "Step 1: Select installation options", select:
(*) Deploy enterprise beans

Click on "Step 6: Bind listeners for message-driven beans"
In the right-most column (titled "Listener Bindings"), click on the bottom section for:
(*) Activation Specification
… and in the field "Target Resource JNDI Name" enter:
  jms/SampleMDBQueueActSpecCCDT

Continue with the deployment of the MDB and wait for a successful deployment.
Notice that the deployment will not automatically start the MDB, do not start it yet.

ADMA5013I: Application SampleMDBEJBEAR installed successfully.
Application SampleMDBEJBEAR installed successfully.

Click on Save.

IMPORTANT! XA is NOT Supported

Please refer to the following technical note for more information on which cases are supported and which cases are NOT supported when using CCDT and XA:

http://www-01.ibm.com/support/docview.wss?uid=swg21631879
Configuring WebSphere Application Server to connect via a Client Channel Definition Table to a WMQ queue manager for Global Transaction (XA) capability.

This is the summary:

+ begin

If you are using a Client Channel Definition Table (CCDT) to connect from WebSphere Application Server to a WebSphere MQ queue manager, the following table shows the configurations in which XA is and is not supported:

Not Supported:

- Using a CCDT to connect to multiple different queue managers (active/active)
- Using a CCDT to connect to multiple multi-instance queue managers (active/active)

Supported:

- CCDT connecting to single queue manager
- CCDT connecting to a single multi-instance queue manager (active/standby)

+ end

IMPORTANT! You must disable XA

DO NOT START THE MDB AT THIS TIME!!!

+ Attention:
Before you start the MDB, it is necessary to modify the ejb-jar.xml file for this MDB, in order to turn off the XA transaction processing.
This modification is needed for EJBs 1.x and 2.x. If you are using EJB 3, see the last section in this document.

Contrary to a Connection Factory (which has an attribute to turn off the XA transactions), an Activation Specification has always turned ON the handling of XA transactions.

However, in order to use the queue manager groups facility it is necessary to turn off the XA transactions, and because it is not possible to turn it off at the level of the Activation Specification, it is necessary to turn the XA transactions at the MDB level.

As root, you need to locate the ejb-jar.xml file for the deployed MDB.
Change to the directory where the WAS profile was created and issue the following command to locate the ejb-jar.xml files.

UNIX:
  cd /opt/IBM/WebSphere/AppServer/profiles/AppSrv01
  find . -name ejb-jar.xml -print

Windows:
  cd "C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01"
or
  cd "C:\Program Files (x86)\IBM\WebSphere\AppServer\profiles\AppSrv01"
  dir /s ejb-jar.xml

There could be several such files, depending on the MDBs that are deployed in your profile. The one that we are looking for is:

Unix:
./config/cells/veracruzNode01Cell/applications/SampleMDBEJBEAR.ear/deployments/SampleMDBEJBEAR/SampleMDBEJB.jar/META-INF/ejb-jar.xml

Windows:
C:\Program Files (x86)\IBM\WebSphere\AppServer\profiles\AppSrv01\config\cells\ANGELITONode01Cell\applications\SampleMDBEJBEAR.ear\deployments\SampleMDBEJBEAR\SampleMDBEJB.jar\META-INF/ejb-jar.xml

+ The current contents of this ejb-jar.xml file is shown below (surrounded by 2 extra lines, one to indicate the beginning and the other to indicate the end):

```
// begin
<?xml version="1.0" encoding="UTF-8"?><ejb-jar id="ejb-jar_ID" version="2.1"
xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/ejb-jar_2_1.xsd">
     <display-name>
     SampleMDBEJB</display-name>
     <enterprise-beans>
        <message-driven id="SampleMDB">
            <ejb-name>SampleMDB</ejb-name>
            <ejb-class>ejbs.SampleMDBBean</ejb-class>
            <messaging-type>javax.jms.MessageListener</messaging-type>
            <transaction-type>Container</transaction-type>
            <message-destination-type>javax.jms.Queue</message-destination-type>
            <activation-config>
                 <activation-config-property>
                       <activation-config-property-
name>destinationType</activation-config-property-name>
                       <activation-config-property-
value>javax.jms.Queue</activation-config-property-value>
                 </activation-config-property>
            </activation-config>
        </message-driven>
     </enterprise-beans>
   <assembly-descriptor/>
     <ejb-client-jar>SampleMDBEJBClient.jar</ejb-client-jar>
// end
```

+ Make a backup copy of this file (preferably outside the directory structure of your WAS installation).

+ Edit the file.

a) Delete the line:

b) In the same location where the above line was located, insert these lines (again, using 2 extra lines to indicate the begin and the end)

```
+ begin
      <assembly-descriptor>
        <container-transaction>
          <method>
             <ejb-name>SampleMDB</ejb-name>
             <method-name>*</method-name>
          </method>
          <trans-attribute>NotSupported</trans-attribute>
        </container-transaction>
      </assembly-descriptor>
+ end
```

c) The complete modified file looks like these. The new lines are shown in blue:

```
// begin
<?xml version="1.0" encoding="UTF-8"?><ejb-jar id="ejb-jar_ID" version="2.1"
xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/ejb-jar_2_1.xsd">
    <display-name>
    SampleMDBEJB</display-name>
    <enterprise-beans>
     <message-driven id="SampleMDB">
          <ejb-name>SampleMDB</ejb-name>
          <ejb-class>ejbs.SampleMDBBean</ejb-class>
          <messaging-type>javax.jms.MessageListener</messaging-type>
          <transaction-type>Container</transaction-type>
          <message-destination-type>javax.jms.Queue</message-destination-type>
          <activation-config>
               <activation-config-property>
                    <activation-config-property-
name>destinationType</activation-config-property-name>
                    <activation-config-property-
value>javax.jms.Queue</activation-config-property-value>
               </activation-config-property>
          </activation-config>
     </message-driven>
    </enterprise-beans>
    <assembly-descriptor>
     <container-transaction>
      <method>
         <ejb-name>SampleMDB</ejb-name>
         <method-name>*</method-name>
      </method>
      <trans-attribute>NotSupported</trans-attribute>
     </container-transaction>
    </assembly-descriptor>
    <ejb-client-jar>SampleMDBEJBClient.jar</ejb-client-jar>
</ejb-jar
// end
```

+ Start the MDB.

Now you can start the MDB.

From the console, in the left panel, select:
Applications > Application Types > WebSphere enterprise applications

From the "Enterprise Applications" window, select "SampleMDBEJBEAR" and click on the box to the left of the name. Then click "Start".

++
+++ Chapter 2: Testing of the MDB, including failover
++

+ Open a command window prompt to put a message into the queue for the queue
manager QM_MIG:

rivera@veracruz: /var/mqm
$ amqsput Q_MDB QM_MIG
Sample AMQSPUT0 start
target queue is Q_MDB
test-msg-from-QM_MIG
Sample AMQSPUT0 end

+ Change to the directory where the WAS SystemOut.log is located. In this test it is:
  cd /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/server1

Issue:
  tail -f SystemOut.log

Notice that the bottom line is provided by the MDB and displays the contents of the
message:

… SystemOut   O +++ SAMPLE MDB: Text Message => test-msg-from-QM_MIG

+ Perform the failover.

Now that we have successfully placed a message into QM_MIG and the MDB has
consumed it, let's stop QM_MIG in order to trigger the reconnection to QM_VER.

Issue:
  endmqm -ir QM_MIG

Notice the errors in the SystemOur.log

[7/24/12 14:12:14:488 EDT] 00000027 SibMessage   W [:] CWSJY0003W: JMSCC3036:
An exception has been delivered to the connections exception listener: '
          Message : com.ibm.msg.client.jms.DetailedJMSException:
JMSWMQ1107: A problem with this connection has occurred. An error has occurred
with the WebSphere MQ JMS connection. Use the linked exception to determine the
cause of this error.
          Class : class com.ibm.msg.client.jms.DetailedJMSException

Stack :
com.ibm.msg.client.wmq.common.internal.Reason.reasonToException(Reason.java:6
08) ...
   Caused by [1] --> Message : com.ibm.mq.MQException: JMSCMQ0001: WebSphere
MQ call failed with compcode '2' ('MQCC_FAILED') reason '2009'
('MQRC_CONNECTION_BROKEN').
               Class : class com.ibm.mq.MQException
               Stack :
com.ibm.msg.client.wmq.common.internal.Reason.createException(Re
...

[7/24/12 14:12:14:506 EDT] 00000028 SibMessage   W   [:] CWSJY0003W: MQJCA4013:
A connection to a queue manager failed for activationSpec
'javax.jms.Queue:jms/SampleMDBQueueNoQmgr@*QMGRP <1379684924>'. Check the
queue manager error logs for details.
[7/24/12 14:12:19:512 EDT] 00000028 SibMessage   W   [:] CWSJY0003W: MQJCA4003:
A recoverable exception occurred in the JMS layer: 'JMSCMQ0002: The method
'MQCTL' failed.'
[7/24/12 14:12:20:650 EDT] 00000028 SibMessage   W   [:] CWSJY0003W: MQJCA4003:
A recoverable exception occurred in the JMS layer: 'JMSWMQ0019: Failed to
disconnect from queue manager '*QMGRP' using connection mode '1' and host name
'null'.'

Notice that the Activation Specification restarted fine:

[7/24/12 14:12:21:329 EDT] 00000029 SibMessage   W   [:] CWSJY0003W: MQJCA4019:
ActivationSpec 'javax.jms.Queue:jms/SampleMDBQueueNoQmgr@*QMGRP
<1379684924>' reconnected successfully.

It means that it was able to reconnect to the other queue manager (QM_VER).
In this case, the failover process took around 7 seconds.

+ Now put a new message into QM_VER (via "amqsput Q_MDB Q_VER") and notice that
the MDB is able to consume it:

SystemOut   O +++ SAMPLE MDB: Text Message => test-msg-from-QM_VER

+++ Modifying the EJB deployment descriptor

The above instructions are for EJB 2.1.

The instructions for modifying the transaction attribute of a method(s) on an MDB (an MDB is a type of EJB) in RAD are located here. This could be useful in case that you do not want to manually modify the ejb-jar.xml file and instead, you want to modify the descriptor from the deployment assembly tool, such as IBM Rational Application Developer or IBM Rational Software Architect.

http://publib.boulder.ibm.com/infocenter/rtnlhelp/v6r0m0/topic/com.ibm.etools.ejb.assembly.doc/topics/teaddcontainertran.html
Rational Software Architect
Defining container transactions for EJB modules

++ Modifications for EJB 3

If you are using EJB 3, then the way these applications are assembled has changed. EJB 3 now uses annotations rather than using deployment descriptors, so the RAD Information Center page above is only applicable to EJB 1.1 and 2.x.

The annotation he needs to add to his code is the
@TransactionAttribute(value = TransactionAttributeType.NOT_SUPPORTED)

… to either the onMessage method or the Bean as a whole.

+++
+++ Chapter 3: Properties to control retry times and retry interval
+++

If you want to modify the default retry times and retry interval for an activation specification, consult the following web page:

http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/index.jsp?
topic=/com.ibm.websphere.base.doc/info/aes/ae/tmj_adm20.html
WebSphere Application Server (Distributed operating systems), Version 7.0 >
Administering applications and their environment > Messaging resources > Managing messaging with the WebSphere MQ messaging provider > Configuring JMS resources for the WebSphere MQ messaging provider
Configuring an activation specification for the WebSphere MQ messaging provider

Under Step 4:
Note: There are four WebSphere MQ connection properties that affect the activation specification:

- maxConnections
- connectionConcurrency
- reconnectionRetryCount
- reconnectionRetryInterval

Although these properties control the behavior of the WebSphere MQ activation specification, they are configured as custom properties of the WebSphere MQ messaging provider. For more information about these four connection properties, and how to configure them, see Configuring custom properties for the WebSphere MQ messaging provider.

http://pic.dhe.ibm.com/infocenter/wasinfo/v7r0/index.jsp?topic=
%2Fcom.ibm.websphere.base.doc%2Finfo%2Faes%2Fae
%2Ftmm_customproperties_ra.html
WebSphere Application Server (Distributed operating systems), Version 7.0 >
Administering applications and their environment > Messaging resources > Managing messaging with the WebSphere MQ messaging provider > Configuring JMS resources for the WebSphere MQ messaging provider
Configuring custom properties for the WebSphere MQ resource adapter

Additional information:

The above properties are documented in the MQ V7 Info Center:

http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp?
topic=/com.ibm.mq.csqzaw.doc/jm40190_.htm
WebSphere MQ > Using Java > WebSphere MQ classes for JMS > Installation and
configuration > The WebSphere MQ resource adapter > Configuration
Configuration of the ResourceAdapter object

Name of property	Type	Default value	Description
maxConnections	String	10	The maximum number of connections to a WebSphere MQ queue manager.
connectionConcurrency	String	5	The maximum number of MDBs that can be supplied by each connection.
reconnectionRetryCount	String	5	The maximum number of attempts made by the resource adapter to reconnect to a WebSphere MQ queue manager if a connection fails.
reconnectionRetryInterval	String	300 000	The time, in milliseconds, that the resource adapter waits before making another attempt to reconnect to a WebSphere MQ queue manager.

Notice that reconnectionRetryInterval is 5 minutes!

+++ end +++