Using custom property connectionNameList to connect to WebSphere MQ multi-instance queue managers from WebSphere Application Server V7 and V8

IBM Techdoc: 7020700

http://www.ibm.com/support/docview.wss?rs=171&uid=swg27020700

Date last updated: 07-Jul-2014

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

+++ Objective

The objective of this technical document is to describe in detail how to exploit from the WebSphere Application Server the connection to Multi-Instance Queue Managers in WebSphere MQ 7.0.1 using a custom property called "connectionNameList" for Activation Specifications or "XMSC_WMQ_CONNECTION_NAME_LIST" for Connection Factories.

There are 2 ways to configure the WebSphere Application Server to work with multi-instance queue managers.

a) By using a custom property called "connectionNameList" or "XMSC_WMQ_CONNECTION_NAME_LIST", in which the host names and ports of the different instances are specified. This is the easiest way.
NOTE: This is covered in this techdoc.

a.1) In WAS 8, the WAS Administrative Console allows you to specify the connection name list during the creation of a Connection Factory or an Activation Specification.

b) By using an MQ "Client Channel Definition Table" (CCDT).
NOTE: This is covered in another techdoc, which shows how to create a CCDT that has multiple queue managers (via a "connection list"). It also shows how a Connection Factory is created via the Administrative Console and specifying the CCDT.
> http://www.ibm.com/support/docview.wss?rs=171&uid=swg27020701
> Using a CCDT file to use WebSphere MQ multi-instance queue managers with WebSphere Application Server V7

Technically speaking, these 2 approaches can be used also for connecting to different queue managers, and not necessarily to multi-instance queue managers. That is, if queue-manager-1 is not available, then the connection name list can tell the MQ JMS client that the next queue manager to try to connect is queue-manager-2, and so on. However, the focus is on multi-instance queue managers.

This document concentrates on the WebSphere Application Server V7 using a Listener Port, although it is possible to use V6.1.

The following is the configuration used to test the material in this techdoc:
- Version of the MQ Queue Manager: 7.0.1.3.
  - Host 1 and Port 1: veracruz.x.ibm.com(1421)
  - Host 2 and Port 2: cbeech.x.ibm.com(1421)
  - Queue Manager Name: QMMI1
  - Local queue; Q_MDB
- Version of the WebSphere Application Server: 7.0.0.13

+++ Important technote

It is highly recommended that you review the details described in the following technote:

http://www.ibm.com/support/docview.wss?uid=swg21508357
Technote: 1508357
Using WebSphere MQ automatic client reconnection with the WebSphere MQ classes for JMS

+++ Restrictions

Note: XA is not supported during automatic client reconnect

The MQ 7.0.1 JMS client introduced also the feature "automatic client reconnection", which is used on JMS standalone applications, but it is not used when running a JMS application within an application server. Instead, the automatic retry mechanisms for Listener Ports and Activation Specifications are used.

The following section of the WAS V7 Information Center mentions that:
"Automatic client reconnection is not supported by WebSphere Application Server."
http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp?
topic=/com.ibm.mq.csqzao.doc/mi21350_.htm
Integration of WebSphere MQ classes for JMS with WebSphere Application Server

http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp?
topic=/com.ibm.mq.amqzag.doc/fa70190_.htm
WebSphere MQ V7 Information Center
Automatic client reconnection
"For a JMS client, set the reconnection options in the connection factory
configuration. JMS clients running in a managed Java EE container are always
reconnectable. The reconnection is managed by the Java EE resource adapter, rather
than by the WebSphere MQ client support."

http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/index.jsp?
topic=/com.ibm.websphere.nd.iseries.doc/info/iseriesnd/ae/tmj_wmq_miqm.html
WebSphere Application Server, Version 7.0
Configuring multi-instance queue manager connections and automatic client
reconnect with WebSphere MQ messaging provider custom properties
There are some restrictions regarding automatic client reconnect:
+ Begin quote
Notes:
 * Automatic client reconnect is not supported within application server processes as
it can cause issues with XA transactions
 * Automatic client reconnect cannot be configured on WebSphere MQ messaging
provider activation specifications.
+ End quote

+++ Table of Contents

The chapters of this document are:

Chapter 1: Prerequisites (MQ 7.0.1.3, Sample MDB, userid with proper authority)
Chapter 2: Creating JNDI objects and adding custom property
Chapter 3: Switchover testing of the Sample MDB - Listener Port
  Test 1: Messages are passed to the MDB, from host-1 "veracruz"
  Test 2: Doing a switchover of the Queue Manager from host-1 "veracruz" to host-2
"cbeech"
  Test 3: MDB gets message that is put into the other instance of the queue manager
(host-2 "cbeech")
Chapter 4: Switchover testing of the Sample MDB - Activation Specification
  Similar steps as in Chapter 3
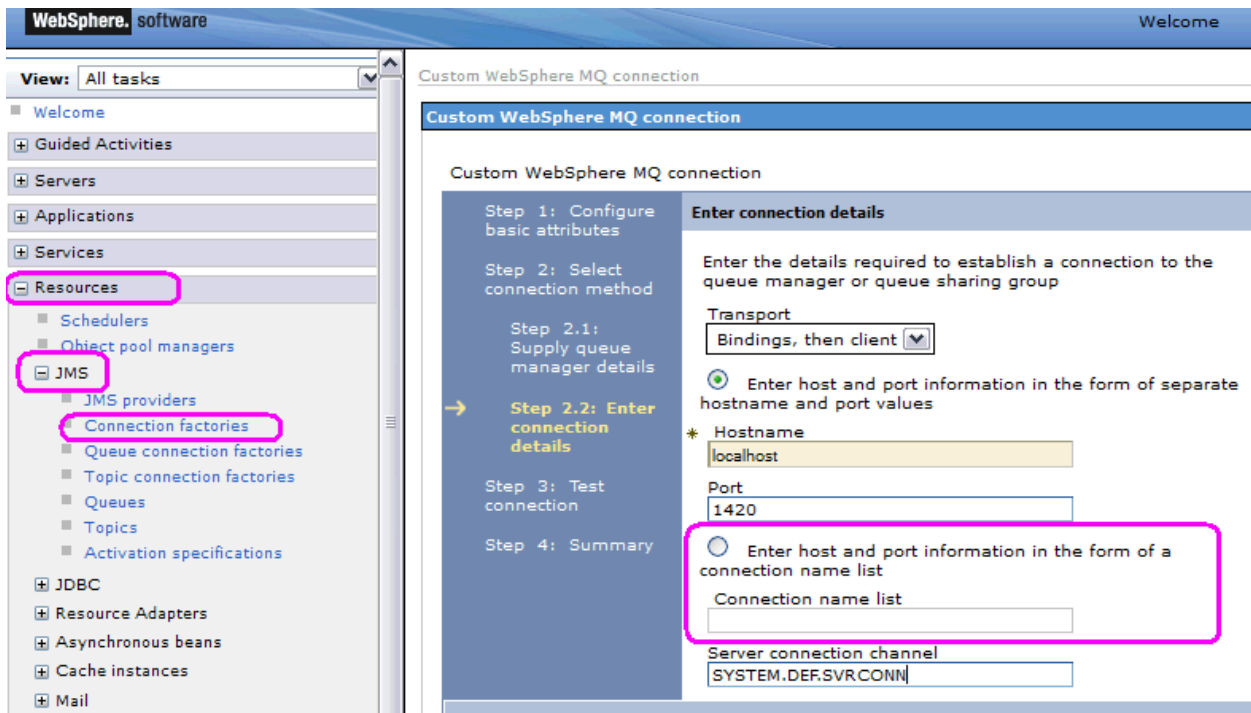Chapter 5: Troubleshooting

+++ Update 07-Jul-2014

Important change in WebSphere Application Server V8.

+ Support for WMQ connection name list in a Connection Factory (CF) or an Activation Specification (AS).
During the creation of any of these objects:
Notice the new radio button to choose between a single host/port or a connection name list

+++ Update on 24-May-2011 = Testing was done using the property connectionNameList for Connection Factories

The WAS V7 Information Center indicates that there are 2 properties to be used:
a) "connectionNameList" for Activation Specifications or
b) "XMSC_WMQ_CONNECTION_NAME_LIST" for Connection Factories.

The testing done for this techdoc was done using only the connectionNameList property for Connection Factories instead of XMSC_WMQ_CONNECTION_NAME_LIST. However, just in case that a future fix pack of either WAS or MQ enforces the rule and breaks the current functionality of using connectionNameList with Connection Factories, it is recommended to use the appropriate property indicated by the Information Center:

http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/index.jsp?
topic=/com.ibm.websphere.base.doc/info/aes/ae/tmj_wmq_miqm.html
Configuring multi-instance queue manager connections with WebSphere MQ messaging provider custom properties
Section: Multi-instance queue managers

Procedure
• To configure a multi-instance queue manager for a WebSphere MQ messaging provider **connection factory**, use the administrative console to complete the following steps:
1. In the navigation pane, select the type of connection factory you want to configure:
■ To configure a connection factory, click Resources > JMS ->Connection factories.
■ To configure a queue connection factory, click Resources > JMS ->Queue connection factories.
■ To configure a topic connection factory, click Resources > JMS ->Topic connection factories.
2. Select the Scope setting corresponding to the scope of the connection factory to be used with a multi-instance queue manager.
3. Follow the steps described in Configuring custom properties for WebSphere MQ messaging provider resources to set a custom property called **XMSC_WMQ_CONNECTION_NAME_LIST** to the list of host and port names that you want to connect with. Specify the list in the format host[(port),host[(port)]]. Specifying the port is optional. The default port is 1414. For example, XMSC_WMQ_CONNECTION_NAME_LIST=host1,host2(1415) first tries to connect to the queue manager on host 1, listening on port 1414. If it cannot connect to host 1 it attempts to connect to host 2, listening on port 1415.

Note: The host and port values specified in the WebSphere MQ multi-instance queue manager must match the host and port values specified in the first entry of XMSC_WMQ_CONNECTION_NAME_LIST.

• To configure a multi-instance queue manager for a WebSphere MQ messaging provider **activation specification**, use the administrative console to complete the following steps:
1. In the navigation pane, click Resources > JMS ->Activation specifications.
2.Select the Scope setting corresponding to the scope of the activation specification to be used with a multi-instance queue manager.
3.Follow the steps described in Configuring custom properties for WebSphere MQ messaging provider resources to set a custom property called **connectionNameList** to the list of host and port names that you want to connect with. Specify the list in the format host[(port),host[(port)]]. Specifying the port is optional. The default port is 1414. For example, connectionNameList=host1,host2(1415) first tries to connect to the queue manager on host 1, listening on port 1414. If it cannot connect to host 1 it attempts to connect to host 2, listening on port 1415.

A corroboration of the above can be found in the following presentation:
http://www.webspghereusergroup.org.uk/wug/files/presentations/31/Matthew_White_-_WMQ_and_Java_EE_-_WUGMarch2011.pdf
WMQ and JavaEE: Living and Working with WAS and other Application Servers
Matthew B White & Graham Hopkins- IBM WebSphere MQ Platform Integration

+ begin quote

Page 31:
Within an Application Server
- Connection to Multi-Instance Queue Managers is supported
- Minimum of WAS 7.0.0.13 required (contains WMQ 7.0.1.3)
- Can be configured using CCDT
- Can be configured using custom properties

- Connection Factories:
Set a custom property called XMSC_WMQ_CONNECTION_NAME_LIST to the list of host/port names that you wish to connect to. For example:
host1(port1),host2(port2)

- Activation Specs:
Set a custom property called connectionNameList on the activation spec with the same format as above: host1(port1),host2(port2)

- Automatic Client Reconnect is not supported (in EJB/WEB container)

- MDB based applications use ListenerPort/Activation Specification based reconnection
- Automatic Client Reconnect can be used in unmanaged/client environments

+ end quote

++ Related articles +++

a) Sample MDB

The testing done in the techdoc is based on the Sample MDB and configuration steps that are described in the following techdoc:

Using WebSphere MQ V7 as JMS Provider for WebSphere Application Server V7
http://www.ibm.com/support/docview.wss?rs=171&uid=swg27016505
Chapter 1: MQ V7 configuration: queue and topic
Chapter 2: WebSphere Application Server V7 configuration: JNDI objects
The following files are included as attachments to the mentioned techdoc
EAR file with MDB: SampleMDBEJB.ear

b) Multi-Instance Queue Managers in Linux

The following references have more information on how to create and do basic switchover and failover tests for Multi-Instance Queue Managers in Linux.

http://www.ibm.com/developerworks/websphere/library/techarticles/1006_sampige
/1006_sampige.html?ca=drs-
Creating a multi-instance queue manager for WebSphere MQ on Linux

http://www.ibm.com/support/docview.wss?uid=swg27017883
Creating and using a WebSphere MQ 7.0.1 multi-instance queue manager in Linux

http://www.ibm.com/support/docview.wss?uid=swg27018726
Using MQ Explorer 7.0.1 to create and to connect to a Multi-instance Queue Manager

c) For those users who are not too familiar with the Administrative Console from WAS, the following technote could be useful to find the items that are related to MQ.

http://www.ibm.com/support/docview.wss?uid=swg27018556
Quick Guide of the Administrative Console from WebSphere Application Server for WebSphere MQ users

++
+++ Chapter 1: Prerequisites (MQ 7.0.1.3, Sample MDB, userid with proper authority)
++

a) The version of the MQ JMS Client must be 7.0.1.3 or later.
http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/topic/com.ibm.mq.amqzag.
doc/fa70190_.htm
Automatic client reconnection

There is an important note in Table 1 of the above link:

Client interface: Messaging APIs
Client: JMS (JSE, and Java EE client container and managed containers)
Program access to reconnection options: 7.0.1.3
Reconnection support: 7.0.1.3

b) The requisite in (a) means that the minimum version of WAS 7 must be WAS
7.0.0.13. The MQ Resource Adapter (RA) shipped with this version of WAS has a
version of: 7.0.1.3.

The following technote provides the mapping of which is the version of the MQ JMS
client (or Resource Adapter) that is shipped with each version of WAS:

http://www.ibm.com/support/docview.wss?uid=swg21248089
Which version of MQ is shipped with WebSphere Application Server

c) The sample MDB called "SampleMDBEJB.ear" from the technical document
mentioned in the "Related article" section will be used. Thus, you will need to consult
that document to properly install the MDB.

d) In MQ JMS V7 when running from within WAS, the userid that is passed to the MQ
queue manager is the userid that starts the WAS processes, such as "root" or
"wsadmin".

It is CRITICAL that this userid must have the proper authorities defined in all the
hosts where the MQ queue managers are running, otherwise, the
security/authorization failure reason code 2035 (MQRC_NOT_AUTHORIZED) will be
issued.

d) You can use WebSphere Application Server V6.1 and install the MQ V7.0.1.3 Client
in the same machine and modify the MQ_INSTALL_ROOT environment variable in WAS
to point to the location of the MQ client, such as /opt/mqm in Linux.

To download the MQ V7.0 Client code, which is a free download and includes already both the base code plus the latest fix pack, and the MQ classes for JMS, visit:

http://www.ibm.com/support/docview.wss?uid=swg24019253
MQC7: WebSphere MQ V7.0 Clients

To customize the MQ_INSTALL_ROOT environment variable in WAS 6.1 to use the MQ V7 Client, consult Chapter 4 " Using Bindings transport type and local native MQ libraries" from the following document:

http://www.ibm.com/support/docview.wss?uid=swg27017881
WebSphere MQ V5.3, V6 and V7 as JMS Provider for WebSphere Application Server V5, V6.0, V6.1 and V7

e) <u>Starting the active and standby instances of the queue manager</u>

Window-1 (MQ): host-1 veracruz:

Login to host-1 ("veracruz" in this example) and start the queue manager to allow a standby instance and display the status:

```
$ strmqm -x QMMI1
WebSphere MQ queue manager 'QMMI1' starting.
5 log records accessed on queue manager 'QMMI1' during the log replay phase.
Log replay for queue manager 'QMMI1' complete.
Transaction manager state recovered for queue manager 'QMMI1'.
WebSphere MQ queue manager 'QMMI1' started.


$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                          STATUS(Running)
    INSTANCE(veracruz) MODE(Active)
```

Notice that only the active instance is running.

Ensure that the local queue (Q_MDB) is defined. This is the queue which is going to be monitored by the Listener Port in WAS.

```
$ runmqsc QMMI1
define qlocal(Q_MDB) replace
     1 : define qlocal(Q_MDB) replace
AMQ8006: WebSphere MQ queue created.
end
     2 : end
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

Window-2 (MQ): host-2 cbeech:

Login to host-2 ("cbeech" in this example) and start the queue manager to allow a standby instance and display the status:

```
$ strmqm -x QMMI1
WebSphere MQ queue manager 'QMMI1' starting.
A standby instance of queue manager 'QMMI1' has been started. The active
instance is running elsewhere.

$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                         STATUS(Running as
standby)
    INSTANCE(veracruz) MODE(Active)
    INSTANCE(cbeech) MODE(Standby)
```

At this point we have 2 instances of the queue manager:
- The active one is running in host-1 "veracruz"
- The standby is running in host-2 "cbeech".

+++
+++ Chapter 2: Creating JNDI objects and adding custom property +++++++++++++++++
+++

Follow the instructions to create JNDI objects in WAS from the mentioned techdoc:
"Using WebSphere MQ V7 as JMS Provider for WebSphere Application Server V7"

A summary of the steps is shown in this chapter.

++ Key item: custom property for "connectionNameList"

The key item for the MQ classes for JMS used by WAS to connect to multi-instance
queue managers is to have a custom property for the connection factory and the
activation specification that lists the queue managers.
This property is called:

connectionNameList

This property has the list of TCP/IP connection names.
The list is tried in order, once per each reconnection retry attempt.
The items in the list must be comma separated items of HOSTNAME(PORT).
The HOSTNAME can be either a DNS name or IP address.
The PORT defaults to 1414, if not specified.

For example, if the value is:
 veracruz.x.ibm.com(1421),cbeech.x.ibm.com(1421)

... then this means that the MQ JMS client will try to connect first to:

Hostname: veracruz.x.ibm.com
Port: 1421

If the client cannot connect or if it connects but later on the connection is lost, then
the client will try the next hostname in the list:

Hostname: cbeech.x.ibm.com
Port: 1421

CAVEAT:
 When using the Administrative Console to create / edit a connection factory and an
activation specification, the FIRST entry in the list, MUST be the hostname and port
number specified in the fields "Hostname" and "Port".

The following screen shots show how to add this custom property to a connection factory:

First, you need to ensure that the Hostname and Port are filled in with a SINGLE hostname and port. That is, these fields in the GUI do NOT support a list of multiple items.

In this example, the values are:

Hostname: veracruz.x.ibm.com
Port: 1421
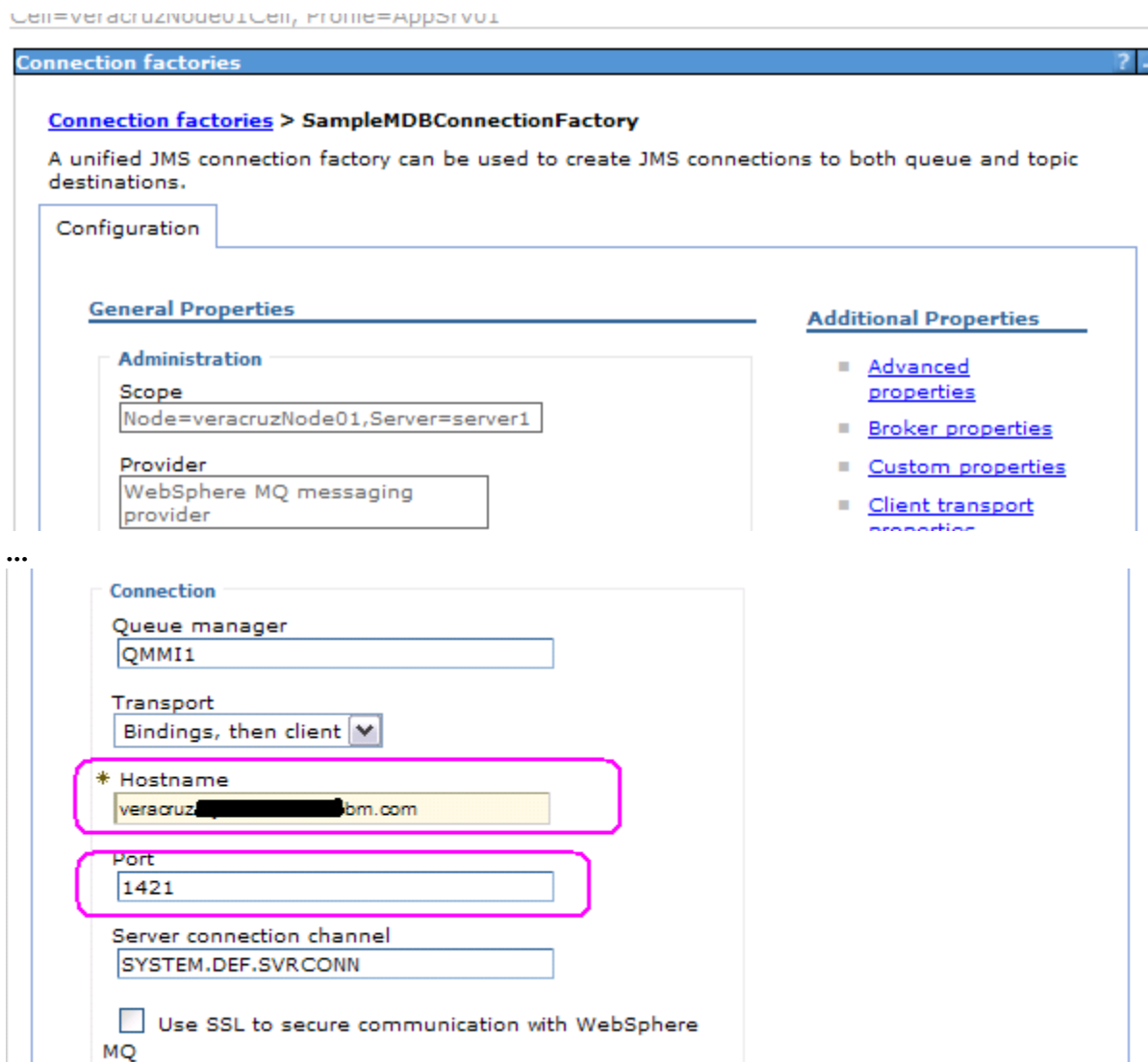
Second, click on "Custom Properties"

Cell=veracruzNode01Cell, Profile=AppSrv01

Connection factories

Connection factories > SampleMDBConnectionFactory

A unified JMS connection factory can be used to create JMS connections to both queue and topic destinations.

Configuration

General Properties

Administration

Scope
Node=veracruzNode01,Server=server1

Provider
WebSphere MQ messaging provider

* Name
SampleMDBConnectionFactory

* JNDI name
jms/SampleMDBConnectionFactory
The JNDI name for the resource.

Description

Additional Properties

■ Advanced properties

■ Broker properties

■ Custom properties

■ Client transport properties

■ Connection pool

■ Session pools

Related Items

■ JAAS - J2C authentication data

Click on "New"

Connection factories > **SampleMDBConnectionFactory** > **Custom properties**

Use this page to specify custom properties that your enterprise information system (EIS) requires for the resource providers and resource factories that you configure. For example, most database vendors require additional custom properties for data sources that access the database.

⊞ Preferences

New Delete

Select	Name ◇	Value ◇	Description ◇	Required ◇
None				
Total 0				

Enter:

Name: connectionNameList
Value: veracruz.x.ibm.com(1421),cbeech.x.ibm.com(1421)
Type: java.lang.String

Connection factories > SampleMDBConnectionFactory > Custom properties > New

Use this page to specify custom properties that your enterprise information system (EIS) requires for the resource providers and resource factories that you configure. For example, most database vendors require additional custom properties for data sources that access the database.

Configuration

General Properties

✱ Scope
cells:veracruzNode01Cell:nodes:veracruzNode01:servers:server1

✱ Name
connectionNameList

Value
beech.   om(1421)

Description

Type
java.lang.String

Apply   OK   Reset   Cancel

Click OK

You will see:

Connection factories

Connection factories > SampleMDBConnectionFactory > Custom properties

Use this page to specify custom properties that your enterprise information system (EIS) requires for the resource providers and resource factories that you configure. For example, most database vendors require additional custom properties for data sources that access the database.

⊞ Preferences

New Delete

Select	Name ◇	Value ◇	Description ◇	Required ◇
You can administer the following resources:				
☐	connectionNameList	veracruz.███████████com (1421),cbeech.████████om (1421)		false

Total 1

The rest of this chapter has the details on the JNDI administrative objects needed for the testing of the Sample MDB.

Note: It is highly recommended that you click on the button "Test Connection" when creating a Connection Factory or an Activation Specification. However, in order for this test to be successful, the queue manager specified in the fields Hostname and in Port must be running. In the case of a multi-instance queue manager, the active instance must be the one specified in the field Hostname for the Connection Factory. For example, if Hostname is host-1 but the active instance is running in host-2, then the Test Connection will fail.

+ Connection Factory:
Name: SampleMDBConnectionFactory
JNDI Name: jms/SampleMDBConnectionFactory
Transport: Bindings, then client
Hostname: veracruz.x.ibm.com
Port: 1421
Server connection channel: SYSTEM.DEF.SVRCONN
Custom property:
  Name: connectionNameList
  Value: veracruz.x.ibm.com(1421),cbeech.x.ibm.com(1421)
  Type: java.lang.String

Note:
Make sure that the Hostname and Port fields of the Connection Factory matches the first entry in the connection name list.

+ Destination: Queue
Name: SampleMDBQueue
JNDI Name: jms/SampleMDBQueue
Queue Name in MQ: Q_MDB
Queue Manager: QMMI1

+ Activation Specification for a Queue:
Name: SampleMDBQueueActivationSpec
JNDI Name: jms/SampleMDBQueueActivationSpec
Destination JNDI name: jms/SampleMDBQueue
Destination Type: Queue
Queue manager: QMMI1
Transport: Bindings, then client
Hostname: veracruz.x.ibm.com
Port: 1421
Server connection channel: SYSTEM.DEF.SVRCONN

Custom property:
 Name: connectionNameList
 Value: veracruz.x.ibm.com(1421),cbeech.x.ibm.com(1421)
 Type: java.lang.String

+ Listener Port for Queue:
Name: SampleMDBQueueLP
Initial State: Started
Connection factory JNDI name: jms/SampleMDBConnectionFactory
Destination JNDI name: jms/SampleMDBQueue

Note:
After you create or edit JNDI items, you must stop and restart the application server, in order to make the changes available to the applications.

++
+++ Chapter 3: Switchover testing of the Sample MDB - Listener Port
++

At this point, it is assumed that you have defined the proper JNDI objects for JMS Resources, installed the Sample MDB with its default mapping to a Listener and restarted the application server.

This chapter shows the steps needed to test the switchover of the multi-instance queue managers and the corresponding reconnection retry from the Listener Port associated with the MDB, using the custom property "connectionNameList" from the connection factory.

The last Test scenario describes the behavior when using an Activation Specification instead of a Listener Port.

Notes:
- Listener Ports for MQ are supported in both WAS V7 and V6.
- Activation Specifications for MQ are only supported with WAS V7.
- For WAS V7, you can use Listener Ports or Activation Specifications for MQ.
- For WAS V6, you can only use Listener Ports for MQ.

++ Test 1: Messages are passed to the MDB, from host-1 "veracruz"

1.1) In WAS, you need to verify that the application is running.
At this point, the Sample MDB is mapped to a Listener Port.

Cell=veracruzNode01Cell, Profile=AppSrv01

Enterprise Applications

Enterprise Applications

Use this page to manage installed applications. A single application can be deployed onto m

⊞ Preferences

| Start | Stop | Install | Uninstall | Update | Rollout Update | Remove File | Export |

Select	Name ⌄	Application Status ↻
	You can administer the following resources:	
☐	DefaultApplication	✖
☐	IBMUTC	✖
☐	SampleJMSSelectorMdbEjbEAR	✖
☐	SampleMDBEJBEAR	➡
☐	SamplePoisonMsgMdbEjbEAR	✖

1.2) In WAS, verify that the Listener Port is running:

Application servers

Application servers > server1 > Message listener service > Listener Ports

Use this page to configure listener ports upon which message-driven beans listen for messages. Each port specifies the JMS connection factory and JMS destination that a message-driven bean, deployed against that port, listens upon.

⊞ Preferences

| Convert to activation specification | New | Delete | Start | Stop |

Select	Name ⌄	Description ⌄	Connection factory JNDI name ⌄	Destination JNDI name ⌄	Status ↻
	You can administer the following resources:				
☐	SampleMDBQueueLP		jms/SampleMDBConnectionFactory	jms/SampleMDBQueue	➡
☐	SampleMDBTopicLP		jms/SampleMDBConnectionFactory	jms/SampleMDBTopic	✖

Window-3 (WAS): host-1 veracruz:

Login to the host where WAS is running and open a command prompt.

Change to the directory where SystemOut.log is stored, such as:

```
$ cd /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/server1
```

Issue the following command to continuously show the most recent entries appended to the end of the SystemOut.log:

```
$ tail  -f  SystemOut.log
```

Window-1 (MQ): host-1 veracruz:

Login to the host where the Queue Manager is running and open a command prompt.

Issue the MQ command to put a message into the queue Q_MDB. Enter the text:
      TEST-MDB <ENTER>
Then press <ENTER> again to terminate the amsput sample.

```
$ amqsput Q_MDB QMMI1
Sample AMQSPUT0 start
target queue is Q_MDB
TEST-MDB

Sample AMQSPUT0 end
```

Window-3 (WAS): host-1 veracruz:

You should notice a new line at the bottom of the SystemOut.log:

```
[1/12/11 13:46:17:437 EST] 00000022 SystemOut     O +++ SAMPLE MDB: Text
Message => TEST-MDB
```

This line is generated by the Sample MDB, which got a message from the WAS Listener Port. Then read the message, extracted the text and printed an entry into the SystemOut.log with the text.

At this point, the scenario has shown that the Sample MDB is working fine to get a message from the active instance running in host-1 "veracruz" of the MQ queue manager.

++ Test 2: Doing a switchover of the Queue Manager from host-1 "veracruz" to host-2 "cbeech"

Window-1 (MQ): host-1 veracruz:

Issue the following command to switchover the running instance of the queue manager from host-1 to host-2.
It is critical to use the following flags
-s   = to switch to another instance
-i   = to stop immediately, forcing an interruption with existing client connections.
          This is critical, because the Listener Port which uses the MQ JMS client in WAS
is connected to the queue manager, waiting for new messages into the queue Q_MDB.
If this flag is not specified, then the endmqm will end only when the Listener Port in
WAS is stopped.

```
$ endmqm -i -s QMMI1
WebSphere MQ queue manager 'QMMI1' ending.
```

Note: the switchover may take more than 30 seconds, depending on the NFS server.
Thus, it is important to allow for sufficient time for the Listener Port to retry the
connection. The default is 60 seconds.

Eventually you should see the following message:

```
WebSphere MQ queue manager 'QMMI1' ended, permitting switchover to a standby
instance.
```

Show the status of the queue manager. It shows that the old standby is now the
active (and only instance).

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                              STATUS(Running
elsewhere)
    INSTANCE(cbeech) MODE(Active)
```

Window-2 (MQ): host-2 cbeech:

From host-2, verify again that the queue manager is running.

```
$ dspmq -x -m QMMI1
QMNAME(QMMI1)                                              STATUS(Running)
    INSTANCE(cbeech) MODE(Active)
```

Window-3 (WAS): host-1 veracruz:

The switchover of the queue manager from host-1 to host-2 will cause a break with the connection of the Listener Port which uses the MQ JMS client.

This will cause a generation of several informational, warning and error messages in the SystemOut.log. This is normal behavior and these messages can be ignored. The following is a sample of such messages:

```
[1/12/11 13:58:01:751 EST] 0000002f 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
…
[1/12/11 13:58:01:765 EST] 00000031 JMSExceptionL E   WMSG0018E: Error on
JMSConnection for MDB SampleMDB , JMSDestination jms/SampleMDBQueue  :
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.
        at
com.ibm.msg.client.wmq.common.internal.Reason.reasonToException(Reason.java:6
08)
…
Caused by: com.ibm.mq.MQException: JMSCMQ0001: WebSphere MQ call failed with
compcode '2' ('MQCC_FAILED') reason '2009' ('MQRC_CONNECTION_BROKEN').
        at
com.ibm.msg.client.wmq.common.internal.Reason.createException(Reason.java:223)
        ... 9 more

[1/12/11 13:58:01:781 EST] 00000031 MCWrapper     E   J2CA0081E: Method
cleanup failed while trying to execute method cleanup on ManagedConnection
com.ibm.ejs.jms.JMSManagedConnection@27012701
        managed connection factory =
com.ibm.ejs.jms.WMQJMSRAManagedConnectionFactory@773a773a
        physical connection = com.ibm.mq.jms.MQXAConnection@27652765
        connectorName = JMS$SampleMDBConnectionFactory$JMSManagedConnection@5
        credential = null
        open connection handles = [] from resource
jms/SampleMDBConnectionFactory. Caught exception:
javax.resource.spi.ResourceAdapterInternalException: Failed to stop
connection on cleanup
…
```

```
Caused by: com.ibm.msg.client.jms.DetailedJMSException: JMSCMQ0002: The
method 'MQCTL' failed. A WebSphere MQ call failed. Please see the linked
exception for more information.
…
com.ibm.ejs.jms.JMSManagedConnection.cleanup(JMSManagedConnection.java:1111)
        ... 16 more
Caused by: com.ibm.mq.MQException: JMSCMQ0001: WebSphere MQ call failed with
compcode '2' ('MQCC_FAILED') reason '2009' ('MQRC_CONNECTION_BROKEN').
        at
com.ibm.msg.client.wmq.common.internal.Reason.createException(Reason.java:223)
        ... 23 more
Caused by: com.ibm.mq.jmqi.JmqiException: CC=2;RC=2009

[1/12/11 13:58:01:800 EST] 00000031 SibMessage     W   [:] CWSJY0003W:
JMSCC3034: The exception is ignored as no exception listener is registered: '
                    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.
…
      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

[1/12/11 13:58:01:874 EST] 00000031 MCWrapper      E   J2CA0081E: Method
destroy failed while trying to execute method destroy on ManagedConnection
com.ibm.ejs.jms.JMSManagedConnection@27012701
        managed connection factory =
com.ibm.ejs.jms.WMQJMSRAManagedConnectionFactory@773a773a
        physical connection = com.ibm.mq.jms.MQXAConnection@27652765
        connectorName = JMS$SampleMDBConnectionFactory$JMSManagedConnection@5
        credential = null
        open connection handles = [] from resource No longer available.
Caught exception: javax.resource.spi.ResourceAdapterInternalException: Failed
to close connection
        at
com.ibm.ejs.jms.JMSCMUtils.mapToResourceException(JMSCMUtils.java:176)
…
Caused by: com.ibm.msg.client.jms.DetailedJMSException: JMSWMQ0019: Failed to
disconnect from queue manager 'QMMI1' using connection mode '1' and host name
'veracruz.x.ibm.com(1421),cbeech.x.ibm.com(1421)'. Please see the linked
exception for more information.
…
Caused by: com.ibm.mq.MQException: JMSCMQ0001: WebSphere MQ call failed with
compcode '2' ('MQCC_FAILED') reason '2009' ('MQRC_CONNECTION_BROKEN').

[1/12/11 13:58:01:886 EST] 00000031 MDBListenerIm I   WMSG0043I: MDB Listener
SampleMDBQueueLP stopped for JMSDestination jms/SampleMDBQueue
```

NOTE: The following message indicates that the Listener Port will try the reconnection

```
[1/12/11 13:58:01:889 EST] 00000031 MDBListenerIm I   WMSG0058I: Listener
Port SampleMDBQueueLP will attempt to restart in 60 seconds
```

NOTE: The following message indicates that the Listener Port has reconnected

```
[1/12/11 13:59:02:704 EST] 00000014 MDBListenerIm I   WMSG0042I: MDB Listener
SampleMDBQueueLP started successfully for JMSDestination jms/SampleMDBQueue
```

++ Test 3: MDB gets message that is put into the other instance of the queue manager (host-2 "cbeech")

Window-2 (MQ): host-2 cbeech:

Put another message into the queue:

```
$ amqsput Q_MDB QMMI1
Sample AMQSPUT0 start
target queue is Q_MDB
TEST-CBEECH

Sample AMQSPUT0 end
```

Window-3 (WAS): host-1 veracruz:

You should notice a new line at the bottom of the SystemOut.log:

```
[1/12/11 14:13:03:683 EST] 00000022 SystemOut     O +++ SAMPLE MDB: Text
Message => TEST-CBEECH
```

This proves that the Listener Port that uses the MQ JMS client is able to reconnect to another instance of the queue manager.

++
+++ Chapter 4: Switchover testing of the Sample MDB - Activation Specification
++

This chapter is practically the same as Chapter 3, but instead of using a Listener Port, an Activation Specification is used.
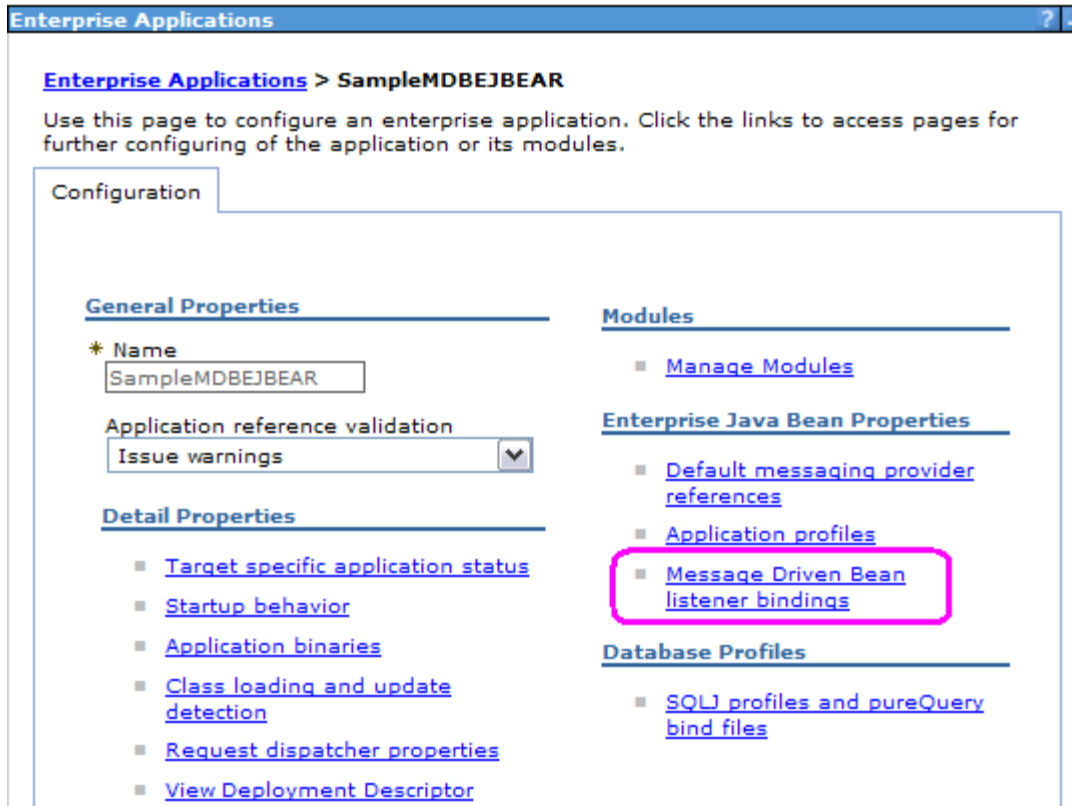
We need to modify the mapping of the MDB to use an Activation Specification instead of a Listener Port.

You need to stop the application and modify it.

Click on the Sample MDB: SampleMDBEJBEAR

Click on "Message Driven Bean listener bindings"

Unselect "Listener Port"

Check "Activation Specification" and specify the Activation Spec for Queues.
Targer Resource JNDI Name: jms/SampleMDBQueueActivationSpec

Click OK and then Save the configuration.

Restart the application

++ Test 1: Messages are passed to the MDB, from host-1 "veracruz"

Follow the same steps as the corresponding Test 1 in Chapter 3.

++ Test 2: Doing a switchover of the Queue Manager from host-1 "veracruz" to host-2 "cbeech"

Follow the same steps as the corresponding Test 2 in Chapter 3.

However, the set of warning and error messages that are displayed in the SystemOut.log when the Activation Specification detects a disconnection with the MQ queue manager from the instance running in host-1 and then the reconnection with the instance at host-2 after the switchover, is different:

```
[1/13/11 10:24:38:656 EST] 00000030 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.

    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

[1/13/11 10:24:38:678 EST] 00000032 SibMessage    W    [:] CWSJY0003W:
MQJCA4013: A connection to a queue manager failed for activationSpec
'javax.jms.Queue:jms/SampleMDBQueue@QMMI1 <1993111244>'. Check the queue
manager error logs for details.
[1/13/11 10:24:43:692 EST] 00000032 SibMessage    W    [:] CWSJY0003W:
MQJCA4003: A recoverable exception occurred in the JMS layer: 'JMSCMQ0002:
The method 'MQCTL' failed.'
[1/13/11 10:24:43:697 EST] 00000032 SibMessage    W    [:] CWSJY0003W:
MQJCA4003: A recoverable exception occurred in the JMS layer: 'JMSWMQ0019:
Failed to disconnect from queue manager 'QMMI1' using connection mode '1' and
host name 'veracruz.x.ibm.com(1421),cbeech.x.ibm.com(1421)'.'
```

NOTE: The following message indicates that the Activation Specification has reconnected

```
[1/13/11 10:29:45:129 EST] 00000033 SibMessage    W    [:] CWSJY0003W:
MQJCA4019: ActivationSpec 'javax.jms.Queue:jms/SampleMDBQueue@QMMI1
<1993111244>' reconnected successfully.
```

++ Test 3: MDB gets message that is put into the other instance of the queue manager (host-2 "cbeech")

Window-2 (MQ): host-2 cbeech:

Put another message into the queue:

```
$ amqsput Q_MDB QMMI1
Sample AMQSPUT0 start
target queue is Q_MDB
TEST-ACT-SPEC-CBEECH
```

Sample AMQSPUT0 end

Window-3 (WAS): host-1 veracruz:

You should notice a new line at the bottom of the SystemOut.log:

```
[1/13/11 10:31:01:544 EST] 0000002f SystemOut     O +++ SAMPLE MDB: Text
Message => TEST-ACT-SPEC-CBEECH
```

This proves that the Activation Specification that uses the MQ JMS client is able to reconnect to another instance of the queue manager.

++
+++ Chapter 5: Troubleshooting
++

This chapter shows the error messages in case that for scenarios.

+ Error that you may get when you do not use the custom property
"connectionNameList" for a Connection Factory or an Activation Specification in WAS,
but rather, you try to use the Administrative Console to enter multiple hostnames and
multiple ports in the fields for "hostname" and "port"

When specifying "Client" mode, this window, does NOT recognize multiple host
names and port numbers.
At least for the Port field, if you attempt to enter 2 numbers, you will get an error
message indicate that you need to enter a single integer.
For the Hostname however, there is no checking. Thus, if you enter 2 hostnames, the
whole string is considered to be the single hostname to use at runtime, which causes
problems:
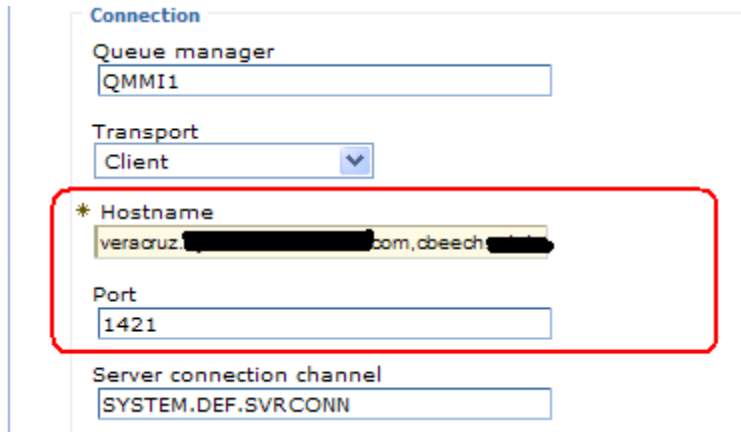
The error message that you will see is:

[1/13/11 11:02:25:855 EST] 00000007 MDBListenerIm W   WMSG0019E: Unable to start
MDB Listener SampleMDB, JMSDestination jms/SampleMDBQueue :
com.ibm.msg.client.jms.DetailedIllegalStateException: JMSWMQ0018: Failed to
connect to queue manager 'QMMI1' with connection mode 'Client' and host name
'veracruz.x.ibm.com,cbeech.x.ibm.com'. Check the queue manager is started and if
running in client mode, check there is a listener running. Please see the linked
exception for more information.

+ Custom properties for the Activation Specification regarding reconnection retry

http://publib.boulder.ibm.com/infocenter/wasinfo/v7r0/index.jsp?
topic=/com.ibm.websphere.base.doc/info/aes/ae/tmj_adm20.html
WAS V7 Information Center
Configuring an activation specification for the WebSphere MQ messaging provider

Note: There are four other 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.

The reconnectionRetryCount and reconnectionRetryInterval 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
Configuration of the ResourceAdapter object

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!

+ Custom properties for the Listener Port regarding reconnection retry

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp?
topic=/com.ibm.websphere.nd.doc/info/ae/ae/umb_prolscp.html
Message listener service custom properties

You can use the Custom properties page to define the following properties for use by the message listener service.

  * MAX.RECOVERY.RETRIES
  * NON.ASF.RECEIVE.TIMEOUT
  * RECOVERY.RETRY.INTERVAL

+++ end +++