Generic JMS Resource Adapter Test Specification

Sonia Liu

Version 1.0
Date last updated 11/02/2006

1.0 Introduction

1.1 Overview

The Generic JMS Resource Adapter version 1.7 (GRA 1.7) helps JMS providers to integrate with J2EE application servers by wrapping their JMS client library in a J2EE Connector Architecture 1.5 resource adapter. The GRA 1.7 will be delievered with Sun Java System Application Server 9.1 (SJSAS9.1).

This test spec is a test document that designed to test the GRA 1.7 functionality. It describes features need to be tested, test scenarios that will be performed by SQE. It also includes test schedule, risks for this project.

1.2 Major feature enhancements in GRA 1.7

- Reliable message redelivery and recovery.
- Load balancing for topic destinations in a cluster configuration.
- Monitoring inbound resource parameters in GRA using JMX
  Test for the Load balancing for topic destination in a cluster configuration will be covered by the HA testing team. This test spec will not cover this feature.

2.0 Technical Description of the new features

- Redelivery is a feature in GRA 1.7 that can be used to resend the message to an MDB (or any listener implementation) if there was an exception during the first send attemp. If the first invocation of the MDB's listener method throws an exception then the RA tries to redeliver it N (number of redelivery attemps configured using the activation config property RedeliveryAttemps in inbound endpoint) times at an interval (configured using activation property RedeliveryInterval). If redelivery is unsucessful even after the attempts are enhausted the RA tries to send the message to a Dead Message Destination (a JMS destination configured in the activation config). In the earlier version of GRA, redelivery was un-reliable because the redelivered message could not be recovered during transaction recovery.

- Monitoring JMS resources is an integral part of an application's life cycle. Statistics collected during runtime aid in creating an optimimum configuration which improves the run-time performance. The application server provides ways to monitor connection-pools that are managed by it. But it would be a good value-add in a resource adapter to provide such a feature so that all the resources created by the ra are available to be monitored during execution. Generic JMS resource adapter exposes such a functionality using JMX MBeans.

3.0 Test scenarios for the Reliable message redelivery and recovery feature

General Setup : MDB listening to a queue "inqueue" and sends message to queue "outqueue". Both "inqueue" and "outqueue" present in one broker.

1.No redelivery test:

Redelivery enabled : attempts is 3 and interval is 1 second. The first delivery succeeds , so redelivery need not happen.

a. Transaction logs in broker should be clean.
b. There is 1 message in outqueue and 0 messages in inqueue (none of them held in transaction).
c. No messages in DMD (does not matter if DMD is configured or not for this test).

Test Strateges:
a. Write a bean class to send a message to a queue "inqueue". MDB listens to the message, then sends a reply to queue "outqueue".
b. Enable Redelivery and DMD in sun-ejb-jar.xml
c. Write a servlet to call ejb method to check the messge received in queue "outqueue".

2. Successful redelivery test:

Redelivery enabled : attempts is 3 and interval is 60 seconds. The first delivery fails, and the redelivery succeeds on the first redelivery attempt.

a. Transaction logs in broker should be clean,
b. There is 1 message in outqueue and 0 messages in inqueue
c. No messages in DMD (does not matter if DMD is configured or not for this test).

Test Strateges:
a. Write a bean class to send a message to a queue "inqueue". MDB listens to the message, then sends a reply to queue "outqueue". Write code to make the first delivery fail, and RA then delivers the message at second time, it successful.
b. Enable Redelivery and DMD in sun-ejb-jar.xml.
c. Write a servlet to call ejb method to check the message received in queue "outqueue".
d. Make sure only one message in the "outqueue". Once the first redelivery is successful, no more redelivery happens.

3. Failed Redelivery

Redelivery enabled : attempts is 3 and interval is 1 second. DMD is not configured , (SendDeadMessageToDMD is false). In this test all the delivery attempts to MDB should fail, make the MDB throw runtime exception for all requests. Since all attempts for delivery fails, the RA tries to send message to DMD, and since DMD is not configured the message should remain in the inqueue.

a. In broker TX logs shows one transaction in STARTED state
b. There is 1 message in the inqueue and should not be held in transaction, no messages in outqueue.
c. No messages in DMD.

Test Strateges:
a. Write a bean class to send a message to a queue "inqueue". MDB listens to the message, then send a reply to queue "outqueue". Make the MDB throw a runtime exception for all requests. The RA then tries to delivery the message to DMD.
b. Enable Redelivery in sun-ejb-jar.xml. Disable DMD in sun-ejb-jar.xml.
c. Write a servlet to call ejb method to check the message is NOT received in queue "outqueue" and DMD. The message should be retained in "inqueue".

4. Failed redelivery - Wrong DMD destination information.

Redelivery enabled : attempts is 3 and interval is 1 second. DMD is configured , (SendDeadMessageToDMD is true) and DMD connection factory information is invalid.

In this scenario all the delivery attempts to MDB should fail, make sure the MDB throw runtime exception for all requests. Since all attempts for delivery fails, the RA tries to send message to DMD, and since DMD is not configured the message should remain in the inqueue.

a. In broker TX logs shows one transaction in STARTED
b. There is 1 message in the inqueue and should not be held in transaction, no messages in outqueue.
c. No messages in DMD.

Test Strateges:
a. Write a bean class to send a message to a queue "inqueue". MDB listens to the message, then sends a reply to queue "outqueue". Make the MDB throw a runtime exception for all requests. The RA then tries to delivery the message to DMD.
b. Enable Redelivery in sun-ejb-jar.xml. Enable DMD in sun-ejb-jar.xml, but setup an incorrect value for the DMD ConnectionFactoryProperties .
c. Write a servlet to call ejb method to check the message is NOT received in queue "outqueue" and DMD. The message should be retained in "inqueue".

5. Failed redelivery - Delivered successfully to DMD.

Redelivery enabled : attempts is 3 and interval is 1 second. DMD is configured , (SendDeadMessageToDMD is true) and DMD connection factory information is valid and DMD destination is valid

In this test all the delivery attempts to MDB should fail, make the MDB throw runtime exception for all requests.
Since all attempts for delivery fails, the RA tries to send message to DMD, and since DMD is not configured the message should be retained in the inqueue.

a. No entries in broker TX logs.

b. There is 0 message in the inqueue and should not be held in transaction, no messages in outqueue.
c. 1 messages in DMD.

Test Strateges:
a. Write a bean class to send a message to a queue "inqueue". MDB listens to the message, then sends a reply to queue "outqueue". Make the MDB throw a runtime exception for all requests. The RA then tries to delivery the message to DMD.
b. Enable Redelivery in sun-ejb-jar.xml. Enable DMD in sun-ejb-jar.xml.
c. Write a servlet to call ejb method to check the message is received in DMD.

6. Redelivery not configured, but DMD is configured

Redelivery disabled, DMD is configured properly with correct CF and destination.

The message delivery to MDB should fail , throw a runtime exception, no redelivery so the message should be retained in broker if its transacted (supportsXA=true)

a. Message is delivered to DMD , 1 message in DMD destination.
b. 0 messages in inqueue with no transactions.
c. 0 messages in outqueue, no transactions pending.

Test Strateges:
a. Write a bean class to send a message to a queue "inqueue". MDB listens to the message, then sends a reply to queue "outqueue". Make the MDB throw a runtime exception. No redelivery happens since the Redelivery is disabled. The message will be delivered to DMD.
b. Disable Redelivery in sun-ejb-jar.xml. Enable DMD in sun-ejb-jar.xml.
c. Write a servlet to call ejb method to check the message is received in DMD.

4.0 Test scenarios for "Monitoring inbound resource parameters in GRA using JMX" feature

Test Strateges:
a. Setup Monitoring=true when create resource adapter
asadmin create-resource-adapter-config --property <other properties>:Monitoring=true
Or setup the following properties in sun-ejb-jar.xml

```
<activation-config-property>
  <activation-config-property-name>Monitoring</activation-config-property-name>
  <activation-config-property-value>true</activation-config-property-value>
</activation-config-property>
<activation-config-property>
  <activation-config-property-name>ApplicationName</activation-config-property-name>
  <activation-config-property-value>PM</activation-config-property-value>
</activation-config-property>
```

b. Implement an inbound messaging endpoint. Then deploy it to AS.

c. Launch Jconsole (java/bin/jconsole). Should see output of the following MBean operations:
String getPoolStatistics(String name) : The argument for this (and all the operations) is the application name that is configured in the deployment descriptor (PM in the above e.g). It returns all the pool parameters for the endpoint in a formatted fashion.

int getCurrentSize(String name) : Returns the current size of the inbound jms resource pool.

int getBusyResources(String name): Returns the number of resources that are marked as busy.

int getFreeResources(String name) : Returns number of resources marked as free.

int getWaiting(String name): Returns number of resources in waiting state.

int getConnections(String name): Returns the number of connections in the pool

int getMaxSize(String name) : Maximum size of the pool.

long getMaxWaitTime(String name): Maximum wait time for the resource to get released.

5.0 Risks

The SQE test only covers Sun MQ and IBM MQ, and all other MQs are not tested.

6.0 Reference

1. https://genericjmsra.dev.java.net/docs/redelivery/redelivery.html
2. http://www.glassfishwiki.org/gfwiki/attach/OnePagersOrFunctionalSpecs/genericra-onepager.txt
3. https://genericjmsra.dev.java.net/
4. https://genericjmsra.dev.java.net/docs/userguide/userguide.html
5. http://blrshare.india.sun.com/~rp138409/jmxmonitoring/jmxmonitoring.html