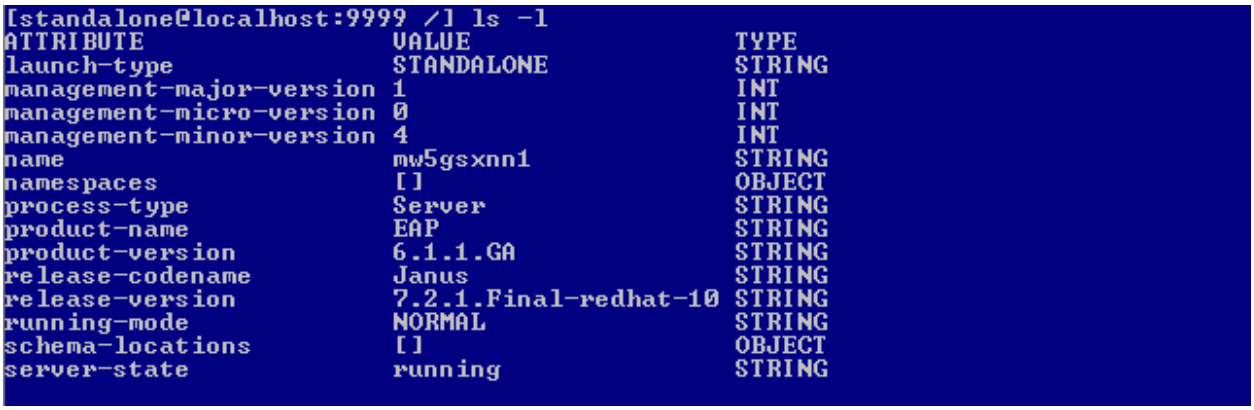
Table of Contents

Software version

1. JBOSS EAP 6.1.1 GA (AS 7.2.1.Final-redhar-10)
2. HornetQ Server 2.3.5.Final-redhat-2
3. JVM 1.6
4. SoftwareAG Integration Server 8.2 running on its own provided JVM 1.6

Screenshot showing JBoss version



HornetQ

HornetQ is a multi-protocol, asynchronous messaging system developed by Red Hat. HornetQ provides high availability (HA) with automatic client failover to guarantee message reliability in the event of a server failure. HornetQ also supports flexible clustering solutions with load-balanced messages.

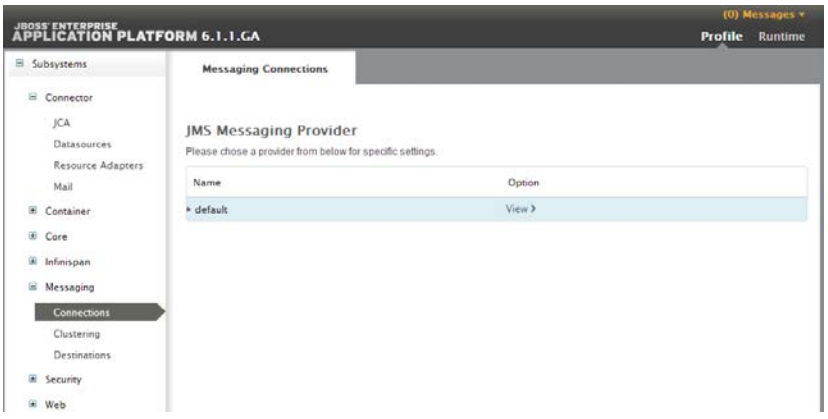
HornetQ In JBOSS EAP

HornetQ is the default (built-in) JMS provider in JBoss AS.

Screenshot showing default HornetQ server running inside JBOSS EAP.

```
[standalone@localhost:9999 hornetq-server=default] ls
acceptor                                        journal-buffer-timeout=undefined
address-setting                                 journal-compact-min-files=10
bridge                                          journal-compact-percentage=30
broadcast-group                                 journal-file-size=10485760
cluster-connection                              journal-max-io=undefined
connection-factory                              journal-min-files=2
connector                                       journal-sync-non-transactional=true
connector-service                               journal-sync-transactional=true
core-address                                    journal-type=NIO
discovery-group                                 log-journal-write-rate=false
divert                                          management-address=jms.queue.hornetq.management
grouping-handler                                management-notification-address=hornetq.notifications
in-vm-acceptor                                  memory-measure-interval=-1
in-vm-connector                                 memory-warning-threshold=25
jms-queue                                       message-counter-enabled=false
jms-topic                                       message-counter-max-day-history=10
path                                            message-counter-sample-period=10000
pooled-connection-factory                       message-expiry-scan-period=30000
queue                                           message-expiry-thread-priority=3
remote-acceptor                                 page-max-concurrent-io=5
remote-connector                                perf-blast-pages=-1
runtime-queue                                   persist-delivery-count-before-delivery=false
security-setting                                persist-id-cache=true
active=true                                     persistence-enabled=true
allow-failback=true                             remoting-incoming-interceptors=undefined
async-connection-execution-enabled=true         remoting-interceptors=undefined
backup=false                                    remoting-outgoing-interceptors=undefined
backup-group-name=undefined                     replication-clustername=undefined
check-for-live-server=false                     run-sync-speed-test=false
cluster-password=CHANGE ME!!                    scheduled-thread-pool-max-size=5
cluster-user=HORNETQ.CLUSTER.ADMIN.USER         security-domain=other
clustered=false                                 security-enabled=true
connection-ttl-override=-1                      security-invalidation-interval=10000
create-bindings-dir=true                        server-dump-interval=-1
create-journal-dir=true                         shared-store=true
failback-delay=5000                             started=true
failover-on-shutdown=false                      thread-pool-max-size=30
id-cache-size=20000                             transaction-timeout=300000
jmx-domain=org.hornetq                          transaction-timeout-scan-period=1000
jmx-management-enabled=false                    version=2.3.5.Final-redhat-2 (Monster Bee, 123)
journal-buffer-size=undefined                   wild-card-routing-enabled=true
```

Screenshot showing HornetQ as default JMS Messaging Provider inside JBOSS EAP



Messaging Configuration

The JMS server configuration is done through the messaging subsystem

Screenshot showing messaging subsystem.

```
[standalone@localhost:9999 subsystem=messaging] ls -l
CHILD          MIN-OCCURS MAX-OCCURS
hornetq-server n/a        n/a
jms-bridge     n/a        n/a
```

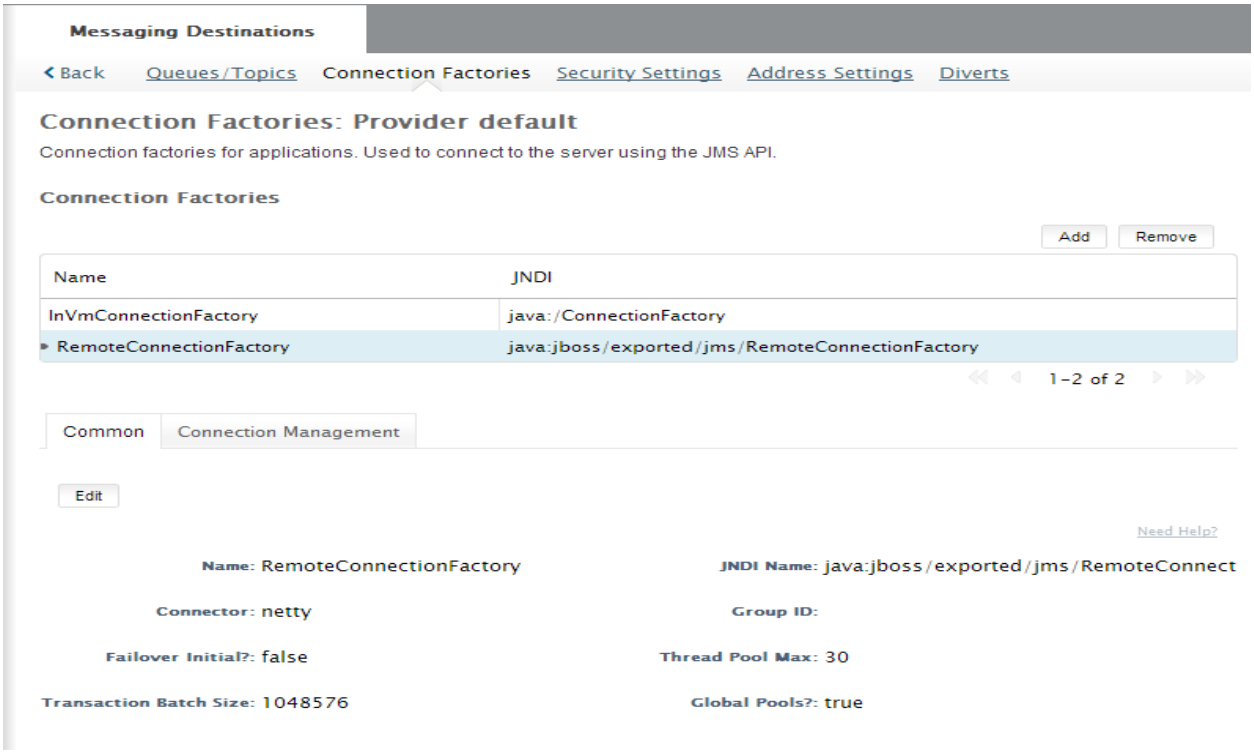
Connection Factories

There are two kinds of basic JMS connection-factory:

1. In-VM connection factories can be used by a local client (i.e. one running in the same JVM as the server)
2. Netty connections factories can be used by a remote client.

Screenshot showing JMS connection factory

```
[standalone@localhost:9999 hornetq-server=default] cd connection-factory=
[standalone@localhost:9999 connection-factory] ls -l
InVmConnectionFactory
RemoteConnectionFactory
```



Acceptors and Connectors

HornetQ uses the concept of connectors and acceptors as a key part of the messaging system.

1. An acceptor defines which types of connections are accepted by the HornetQ server.
2. A connector defines how to connect to a HornetQ server, and is used by the HornetQ client.

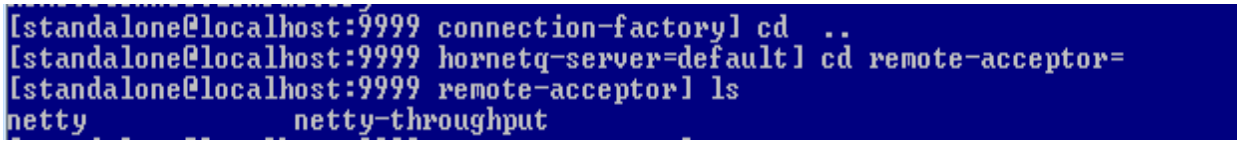
Types of connectors and acceptors

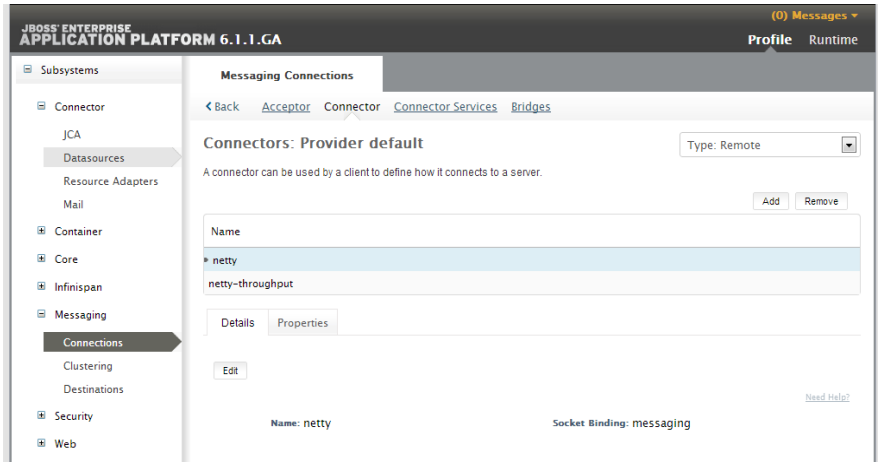
There are two types of connectors and acceptors, relating to the whether the matched connector and acceptor pair occurs within same JVM or not.

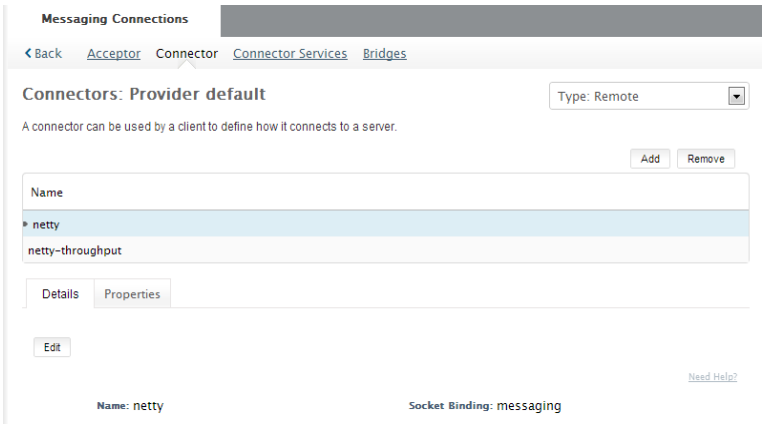
1. Invm -Invm is short for Intra Virtual Machine. It can be used when both the client and the server are running in the same JVM.
2. Netty -The name of a JBoss project. It must be used when the client and server are running in different JVMs.

A HornetQ client must use a connector that is compatible with one of the server's acceptors. Only an Invm connector can connect to an Invm acceptor, and only a netty connector can connect to netty acceptor. The connectors and acceptors are both configured on the server in a standalone.xml and domain.xml. You can use either the Management Console or the Management CLI to define them.

Screenshot showing remote netty acceptor and connector







Start JBOSS and create users

Start JBOSS EAP as standalone-full configuration

1. On command prompt go to JBoss AS home directory then to bin directory
2. Execute command

   standalone.bat -c standalone-full.xml

Screenshot showing JBOSS EAP start from command prompt

```
C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\bin>
C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\bin>
C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\bin>standalone.bat -c standalone-full.xml
Calling "C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\bin\standalone.conf.bat"
Setting JAVA property to 'C:\Program Files\Java\jdk1.6.0_33\bin\java'
=========================================================================

  JBoss Bootstrap Environment

  JBOSS_HOME: C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1
```

Create management user.

Management user is required to access JBOSS EAP admin console.

1. On command prompt go to JBoss AS home directory then to bin directory
2. Run add-user.bat

Refer below screenshot to create management user.

```
C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\bin>add-user.bat

What type of user do you wish to add?
 a) Management User (mgmt-users.properties)
 b) Application User (application-users.properties)
(a): a

Enter the details of the new user to add.
Realm (ManagementRealm) :
Username : JMSAdminUser
Password :
Re-enter Password :
About to add user 'JMSAdminUser' for realm 'ManagementRealm'
Is this correct yes/no? yes
Added user 'JMSAdminUser' to file 'C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\standalone\configuration\mgmt-users.properties'
Added user 'JMSAdminUser' to file 'C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\domain\configuration\mgmt-users.properties'
Is this new user going to be used for one AS process to connect to another AS process?
e.g. for a slave host controller connecting to the master or for a Remoting connection for server to server EJB calls.
yes/no? yes
To represent the user add the following to the server-identities definition <secret value="Sk1TQWRtaW5Vc2VyMSE=" />
Press any key to continue . . .
```

Create application user.

Application user is required to do JNDI lookup from client.

1. On command prompt go to JBoss AS home directory then to bin directory
2. Run add-user.bat

Refer below screenshot to create application user. I picked application role as guest, which default available role.

```
C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\bin>add-user.bat

What type of user do you wish to add?
 a) Management User (mgmt-users.properties)
 b) Application User (application-users.properties)
(a): b

Enter the details of the new user to add.
Realm (ApplicationRealm) :
Username : JMSTestUser
Password :
Re-enter Password :
What roles do you want this user to belong to? (Please enter a comma separated list, or leave blank for none)[  ]: guest
About to add user 'JMSTestUser' for realm 'ApplicationRealm'
Is this correct yes/no? yes
Added user 'JMSTestUser' to file 'C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\standalone\configuration\application-users.properties'
Added user 'JMSTestUser' to file 'C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\domain\configuration\application-users.properties'
Added user 'JMSTestUser' with roles guest to file 'C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\standalone\configuration\application-roles.properties
Added user 'JMSTestUser' with roles guest to file 'C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\domain\configuration\application-roles.properties'
Is this new user going to be used for one AS process to connect to another AS process?
e.g. for a slave host controller connecting to the master or for a Remoting connection for server to server EJB calls.
yes/no? yes
To represent the user add the following to the server-identities definition <secret value="Sk1TVGVzdFVzZXIxIQ==" />
Press any key to continue . . .

C:\as2576\hornetq\jboss-eap-6.1.1\jboss-eap-6.1\bin>
```

JMS Queues and Topics

JMS queues and topics are sub resources of the messaging subsystem. They are defined in the jms-destinations section. One can define either a jms-queue or jms-topic. Each destination *must* be given a name and contain at least one entry element.

Each entry refers to a JNDI name of the queue or topic. Keep in mind that any jms-queue or jms-topic which needs to be accessed by a remote client needs to have an entry in the "java:jboss/exported" namespace. As with connection factories, if a jms-queue or jms-topic has an entry bound in the "java:jboss/exported" namespace a remote client would look it up using the text *after* "java:jboss/exported". For example, the following jms-queue"testQueue" is bound to "java:jboss/exported/jms/queue/test" which means a remote client would look-up this jms-

queue using "jms/queue/test". A local client could look it up using "java:jboss/exported/jms/queue/test", "java:jms/queue/test", or more simply "jms/queue/test"

Create queue in JBOSS HornetQ (using JBOSS EAP Management console)

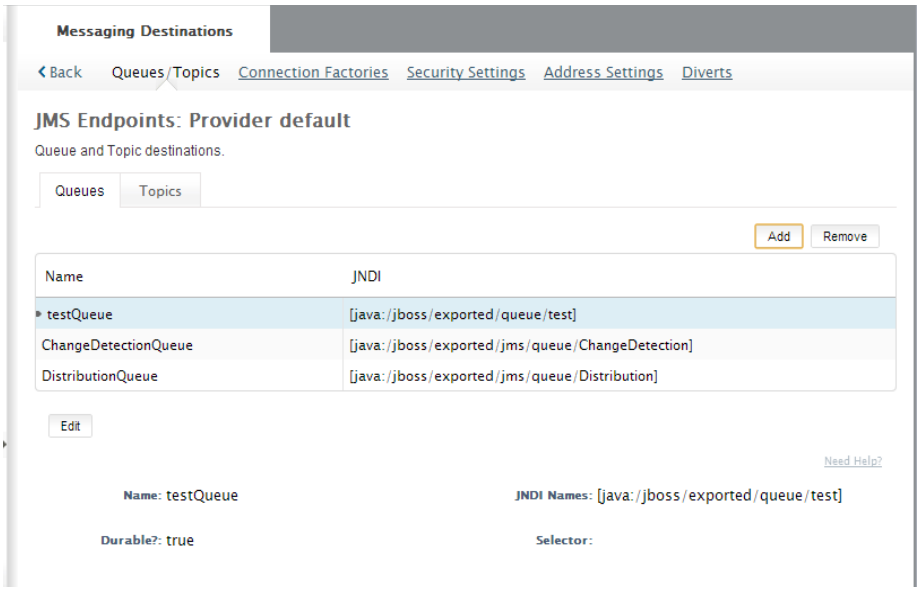
Note: This assumes JBOSS EAP is up and running, user such as management user and application user has been created.

Hit http://localhost:9990/console/App.html in web browser. By default it shows server-overview and runtime view.

1. Change to Profile view
2. Under Subsystem, hit Messaging then Destinations
3. Under Queues, hit button Add.
4. Refer below screenshot to add queue.

5. Save.

Screenshot showing Queues.



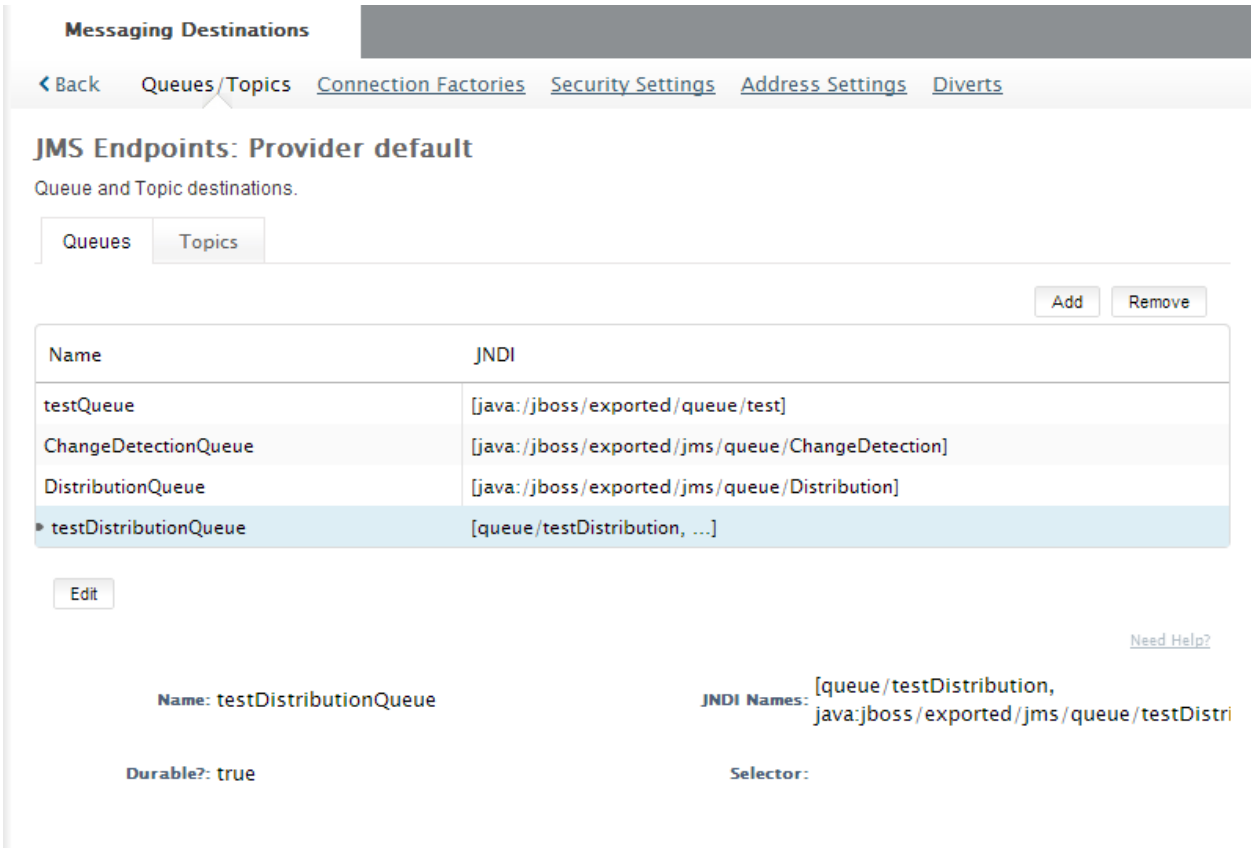
Create queue in JBOSS HornetQ (using Management CLI)

1. On command prompt go to JBoss AS home directory then to bin directory
2. Run jboss-cli.bat
3. connect

Example

jms-queue add --profile=full --queue-address=testDistributionQueue --entries=queue/testDistribution,java:jboss/exported/jms/queue/testDistribution

Screenshot showing queue created by using management CLI



HornetQ integration with webmethods

Add JBOSS related JARs

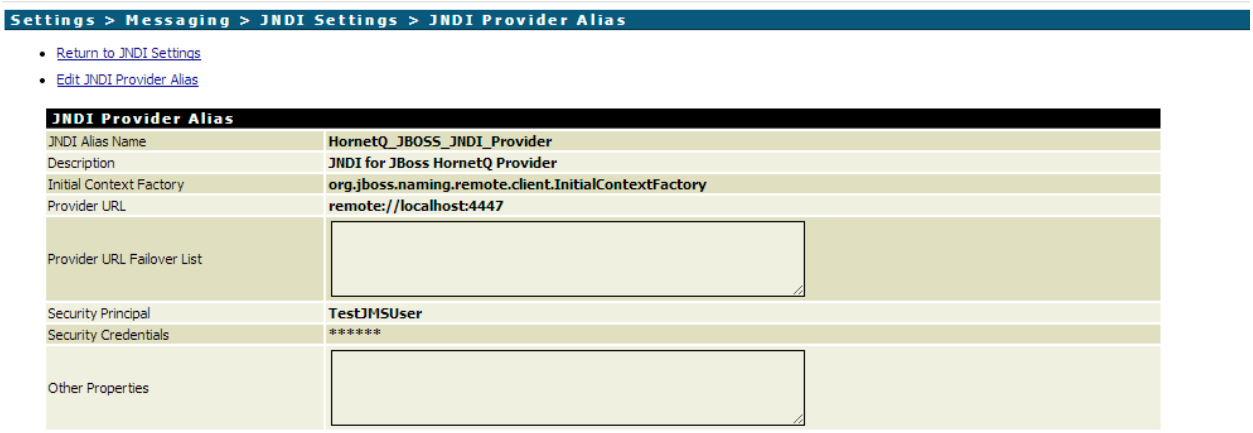
1. Add jboss-client.jar and jboss-remote-naming-1.0.6.Final.jar to /SoftwareAG/IntegrationServer/lib/jars
2. Start Integration Server

Start Integration Server and make sure it gets started without any exception.

3. Go to Integration Servers' Administrator console

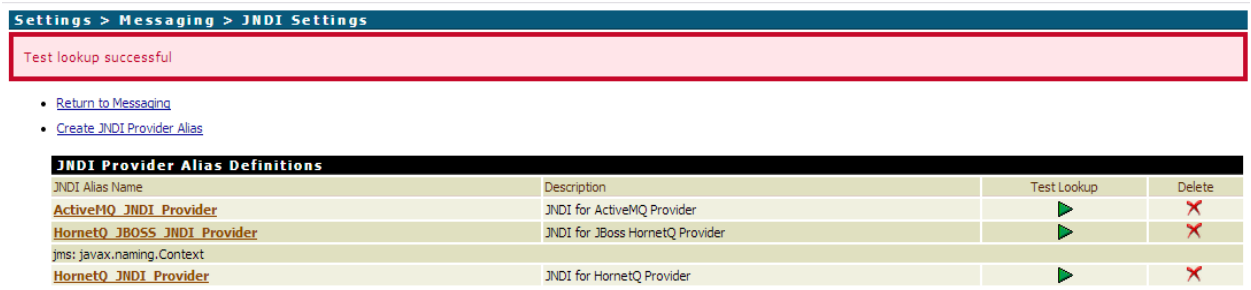
Create JNDI Provider Alias

4. Navigate to Messaging and hit **Create JNDI Provider Alias**, refer below screenshot to fill in the details.
5. Save Changes.



6. Test JNDI lookup –Hit green arrow across the defined JNDI Provider Alias.

Make sure JNDI lookup succeeds without exception.



7. Navigate to Messaging and hit JMS Settings.

Create JMS Connection Alias

8. Hit **Create JMS Connection Alias**. Refer below screenshot to fill in the details.
9. Save Changes.

Settings > Messaging > JMS Settings > JMS Connection Alias

- Return to JMS Settings
- Edit JMS Connection Alias

General Settings	
Connection Alias Name	**HornetQ_JBOSS_JMS_Connection**
Enabled	✔ **Yes**
Description	**JMS connection alias for JBOSS HornetQ**
Transaction Type	**NO_TRANSACTION**
Connection Client ID	**HornetQ_JBOSS_Client**
User	**TestJMSUser**
Password	*****
Connection Protocol Settings	
Create Connection Using	**JNDI LOOKUP**
JNDI Provider Alias	**HornetQ_JBOSS_JNDI_Provider**
Connection Factory Lookup Name	**jms/RemoteConnectionFactory**
Monitor webMethods Connection Factory	**No**
Advanced Settings	
Class Loader	**INTEGRATION_SERVER**
Maximum CSQ Size	**UNLIMITED**
Drain CSQ In Order	**Yes**
Create Temporary Queue	**Yes**
Manage Destinations (webMethods JMS provider only)	**Yes**
Create New Connection per Trigger	**No**
Producer Caching	
Caching Mode	**DISABLED**
* Enabled Connection Aliases can not be edited.	

10. Enable defined JMS Connection. Hit hyperlinked No under Enabled column.

Make sure JMS Connection gets enabled without any exceptions.

- Return to Messaging
- Create JMS Connection Alias
- Refresh Page

JMS Connection Alias Definitions

Connection Alias Name	Description	Transaction Type	CSQ Count	Enabled	Delete
ActiveMQ_JMS_Connection	JMS connection alias for Active MQ	NO TRANSACTION	0	No	✘
DEFAULT_IS_JMS_CONNECTION	system generated JMS connection alias	NO TRANSACTION	0	No	✘
HornetQ_JBOSS_JMS_Connection	JMS connection alias for JBOSS HornetQ	NO TRANSACTION	0	No	✘
HornetQ_JMS_Connection	JMS connection alias for HornetQ	NO TRANSACTION	0	No	✘
PE_NONTRANSACTIONAL_ALIAS	system generated JMS connection alias	NO TRANSACTION	0	No	✘

Settings > Messaging > JMS Settings

JMS Connection Alias "HornetQ_JBOSS_JMS_Connection" set to enabled.

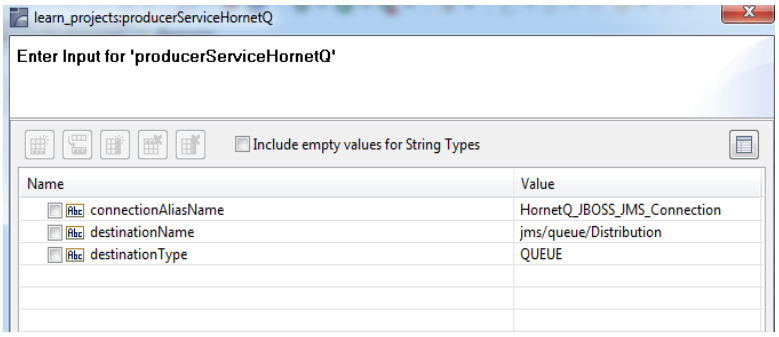
- Return to Messaging
- Create JMS Connection Alias
- Refresh Page

JMS Connection Alias Definitions

Connection Alias Name	Description	Transaction Type	CSQ Count	Enabled	Delete
ActiveMQ_JMS_Connection	JMS connection alias for Active MQ	NO TRANSACTION	0	No	✘
DEFAULT_IS_JMS_CONNECTION	system generated JMS connection alias	NO TRANSACTION	0	No	✘
HornetQ_JBOSS_JMS_Connection	JMS connection alias for JBOSS HornetQ	NO TRANSACTION	0	✔ Yes	✘
HornetQ_JMS_Connection	JMS connection alias for HornetQ	NO TRANSACTION	0	No	✘
PE_NONTRANSACTIONAL_ALIAS	system generated JMS connection alias	NO TRANSACTION	0	No	✘

Producer Flow Service (test)

Producer Flow service to test connectivity and JNDI lookup from webmethods.

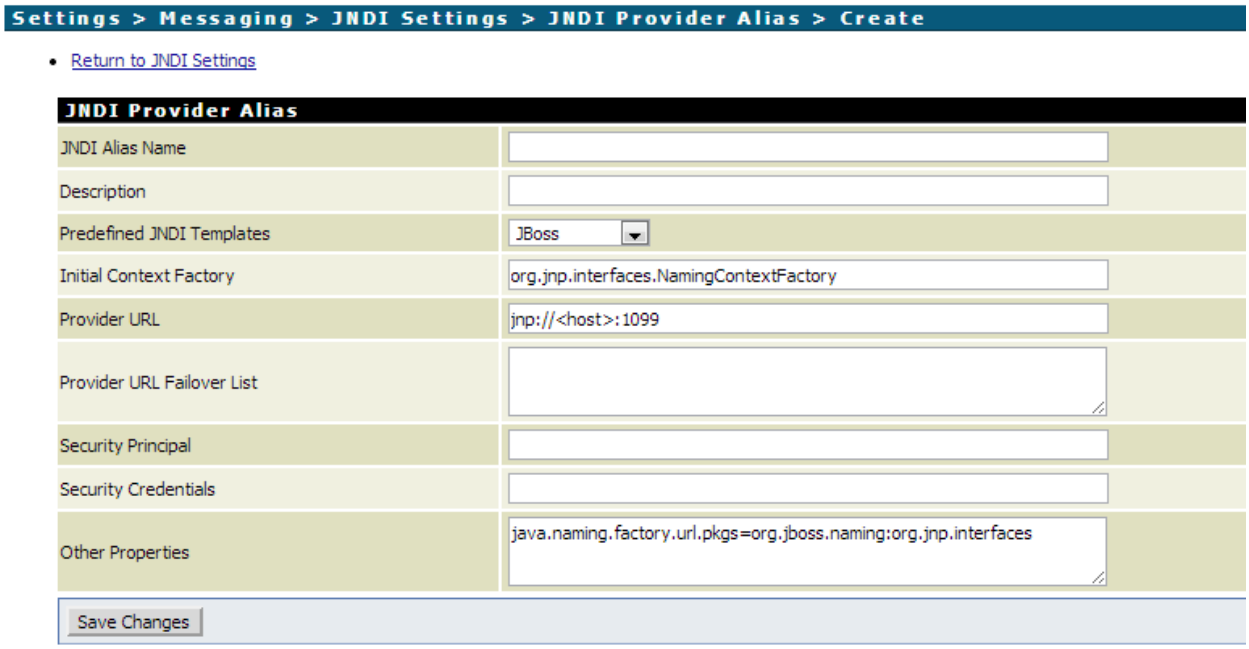


HornetQ standalone (not with JBOSS EAP) integration with webmethods

Software version

1. hornetq-2.3.0.Final
2. JVM 1.6

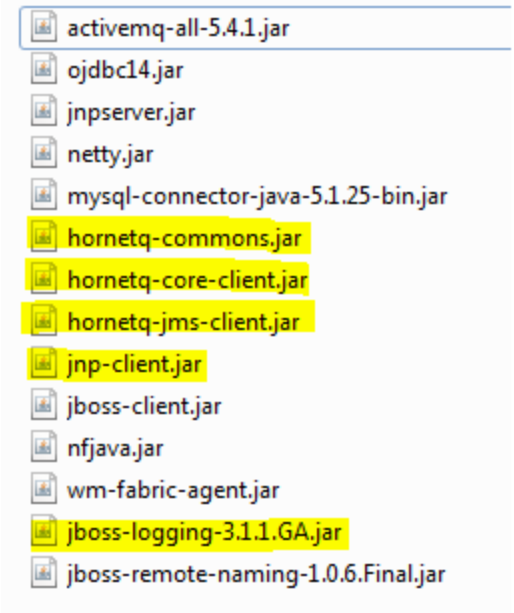
Webmethods supports JNP lookup and JNP connection protocol.



JNP: Java Naming Provider

Add HornetQ and JBOSS logging related JARs

1. Add highlighted JARs to /SoftwareAG/IntegrationServer/lib/jars



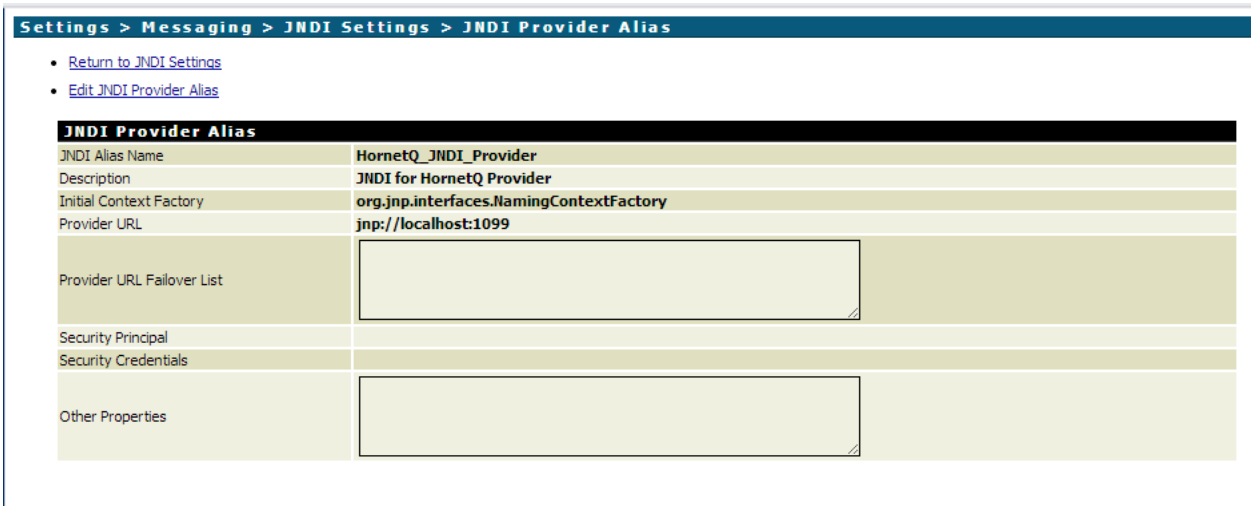
2. Start Integration Server

Start Integration Server and make sure it gets started without any exception.

3. Go to Integration Server's Administrator console.

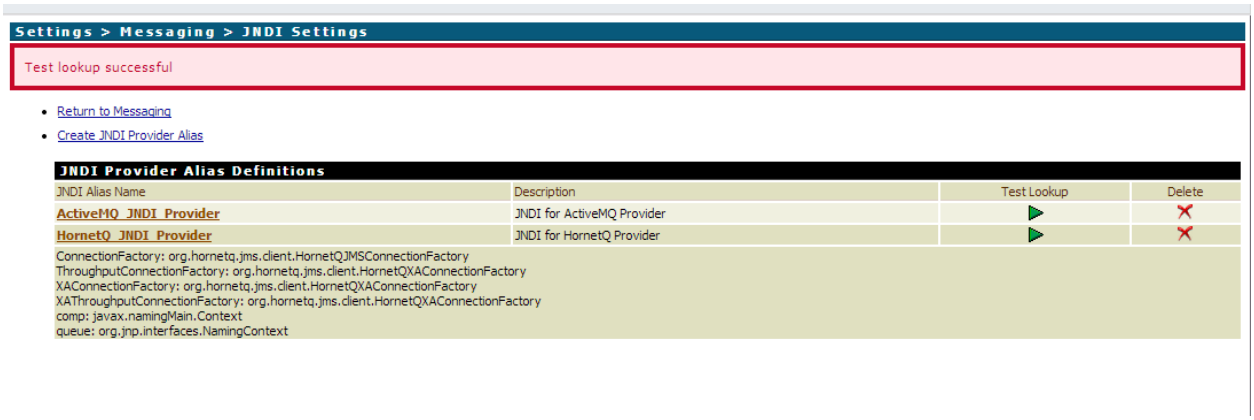
Create JNDI Provider Alias

4. Navigate to Messaging and hit **Create JNDI Provider Alias**, refer below screenshot to fill in the details.
5. Save Changes.



6. Test JNDI lookup –Hit green arrow across the defined JNDI Provider Alias.

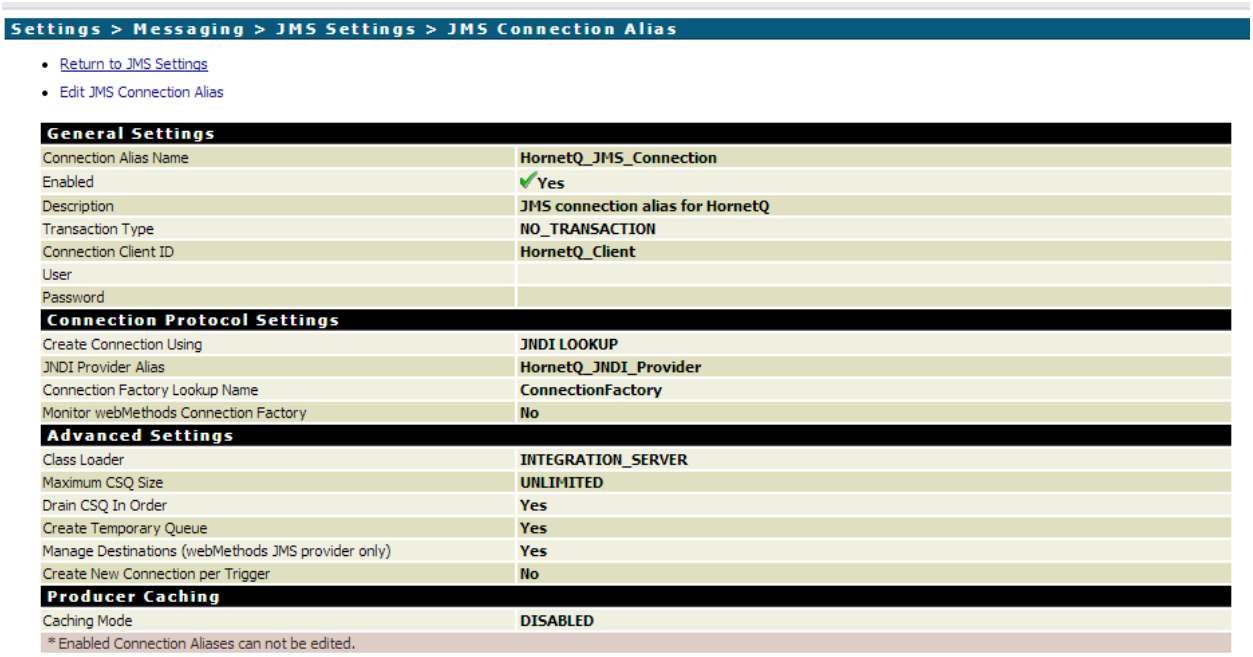
Make sure JNDI lookup succeeds without exception.



7. Navigate to Messaging and hit JMS Settings.

Create JMS Connection Alias

8. Hit **Create JMS Connection Alias**. Refer below screenshot to fill in the details.
9. Save Changes.



10. Enable defined JMS Connection. Hit hyperlinked No under Enabled column.

Make sure JMS Connection gets enabled without any exceptions.

- Return to Messaging
- Create JMS Connection Alias
- Refresh Page

JMS Connection Alias Definitions

Connection Alias Name	Description	Transaction Type	CSQ Count	Enabled	Delete
ActiveMQ_JMS_Connection	JMS connection alias for Active MQ	NO TRANSACTION	0	No	
DEFAULT_IS_JMS_CONNECTION	system generated JMS connection alias	NO TRANSACTION	0	No	
HornetQ_JBOSS_JMS_Connection	JMS connection alias for JBOSS HornetQ	NO TRANSACTION	0	No	
HornetQ_JMS_Connection	JMS connection alias for HornetQ	NO TRANSACTION	0	No	
PE_NONTRANSACTIONAL_ALIAS	system generated JMS connection alias	NO TRANSACTION	0	No	

Settings > Messaging > JMS Settings

JMS Connection Alias "HornetQ_JMS_Connection" set to enabled.

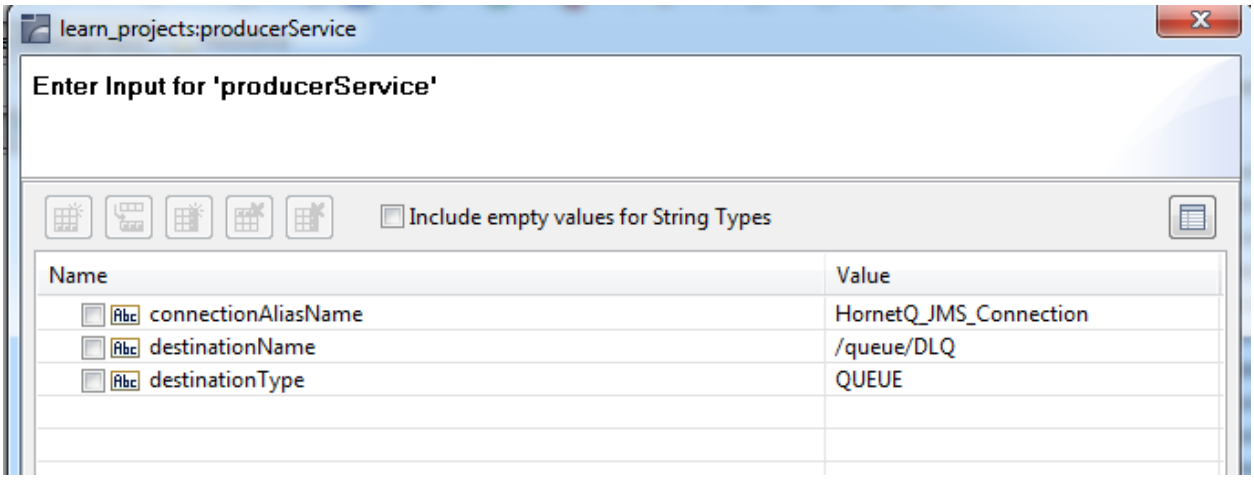
- Return to Messaging
- Create JMS Connection Alias
- Refresh Page

JMS Connection Alias Definitions

Connection Alias Name	Description	Transaction Type	CSQ Count	Enabled	Delete
ActiveMQ_JMS_Connection	JMS connection alias for Active MQ	NO TRANSACTION	0	Yes	
DEFAULT_IS_JMS_CONNECTION	system generated JMS connection alias	NO TRANSACTION	0	No	
HornetQ_JMS_Connection	JMS connection alias for HornetQ	NO TRANSACTION	0	Yes	
PE_NONTRANSACTIONAL_ALIAS	system generated JMS connection alias	NO TRANSACTION	0	No	

Producer Flow Service (test)

Producer Flow service to test connectivity and JNDI lookup from webmethods.



Using default DLQ queue. One can find queue definition and add new queue definition to hornetq-jms.xml which can located at %HORNETQ_HOME%/config/stand-alone/non-clustered