



**Using the JNBridge JMS Adapter for BizTalk Server
with IBM WebSphere
Version 4.0**

www.jnbridge.com

Using the JMS Adapter with WebSphere

JNBridge, LLC
www.jnbridge.com

COPYRIGHT © 2008-2016 JNBridge, LLC. All rights reserved.

JNBridge is a registered trademark and JNBridgePro and the JNBridge logo are trademarks of JNBridge, LLC.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

Microsoft, Windows, Windows Server, BizTalk and the Windows logo are trademarks, or registered trademarks of Microsoft Corporation in the United States and/or other countries.

All other marks are the property of their respective owners.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

Contents

Quick Config for WebSphere	4
Adapter Transport Handler Properties.....	4
Adapter Send or Receive Port Properties	5
Using the JMS Adapter with WebSphere	6
Resources.....	6
BizTalk Server Machine Prerequisites	6
Configuring the BizTalk Server Machine.....	7
Installing the IBM WebSphere 6 JMS client	7
Obtaining the IBM WebSphere 7/8 JMS client.....	7
Configuring the Sun ORB.....	7
Deploy JMS Header Schema to BizTalk Application	8
Configuring the Adapter Send and Receive Transport Handlers.....	8
JMS Properties Category	8
JNBridge Properties Category.....	10
Security Properties Category	11
Configuring Send Ports and Receive Locations	12
Configuring Send Ports.....	12
Connection Properties Category	12
JMS Operation Properties Category	12
Configuring Receive Ports and Locations.....	13
Connection Properties Category	13
JMS Operation Properties Category	13
JNDI Names	15
Using the RefFSContextFactory JNDI with WebSphere MQ.....	16

Using the JMS Adapter with WebSphere

Quick Config for WebSphere

The proper configuration of the adapter for WebSphere JMS is dependent on the presence of the WebSphere Application Server (WAS). The application server provides JMS as a generic service, abstracting the underlying messaging provider. This means that regardless of the underlying messaging provider, WAS provides a consistent configuration for the JMS service. This is true even if the underlying messaging provider in WAS is WebSphere MQ (WMQ).

If WAS is not being used, then WMQ is being used as a *stand-alone* broker—there is no application server. In that case, the configuration will be different. Notably, the WMQ configuration will be dependent on the Java Naming and Directory Interface (JNDI) used by WMQ. JNDI is required by the adapter. A JNDI-like scaffolding is used by WMQ. The JMS adapter supports the file-based JNDI based on the Sun *RefFSContext* implementation. This support requires the use of a wrapper that extends the *RefFSContextFactory* class. For more information, see the section *Using the RefFSContextFactory JNDI with WebSphere MQ* at the end of this document.

Adapter Transport Handler Properties

■ Initial Context Factory

Using WAS: `com.ibm.websphere.naming.WsnInitialContextFactory`

Using stand-alone WMQ: `com.sun.jndi.fscontext.JnbBtsRefFSContextFactory`

■ JMS Scheme

Using WAS: `iiop`

Using stand-alone WMQ: `fscontext`

■ Queue Connection Factory:

No connection factories are pre-configured in WAS or WMQ. Please configure connection factories and use the defined JNDI path as the value for this property.

■ Topic Connection Factory:

No connection factories are pre-configured in WAS or WMQ. Please configure connection factories and use the defined JNDI path as the value for this property.

■ Class Path

Using WAS

For WebSphere 6

IBM JRE: `sibc.jndi.jar`, `sibc.jms.jar`

Using the JMS Adapter with WebSphere

Oracle/Sun JRE: `sibc.jndi.jar`, `sibc.jms.jar`, `sibc.orb.jar`

For WebSphere 7 and 8 (for version 8, use JAR files with ‘_8.x.x’):

IBM JRE:

`com.ibm.ws.sib.thin.jms_7.0.0.jar`
`com.ibm.ws.ejb.thinclient_7.0.0.jar`

Oracle/Sun JRE:

`com.ibm.ws.sib.thin.jms_7.0.0.jar`
`com.ibm.ws.ejb.thinclient_7.0.0.jar`
`com.ibm.ws.orb_7.0.0.jar`

Using WMQ:

`com.ibm.mqjms.jar`
`com.ibm.mq.jar`
`jnbjms-fscontext.jar`

■ JVM Path (examples)

Using IBM JRE: `C:\Program Files\IBM\WebSphere\AppServer\java\jre\bin\j9vm\jvm.dll`

Using Oracle/Sun JRE: `C:\Program Files\Java\jre7\bin\client\jvm.dll`

■ JVM Argument

For WMQ (only):

`-Djava.naming.provider.url=file:/C:/[PathToRefFSContextBindingFile]`

Adapter Send or Receive Port Properties

■ Port Number:

Using WAS: **2809**

Using WMQ: **1414**

■ JMS Object Name

No destinations are pre-configured in in WAS or WMQ. Please configure destinations and use the defined JNDI path as the value for this property.

Using the JMS Adapter with WebSphere

Using the JMS Adapter with WebSphere

This document demonstrates connecting the JNBridge JMS Adapter for BizTalk Server to the IBM WebSphere Application Server, (WAS) or a stand-alone WebSphere MQ Queue Manager (WMQ). The configuration discussed here is applicable to WAS releases 6/7/8. The document is applicable to WMQ version 7/8. Within WAS, the messaging provider that supplies the JMS service will always require the WAS connection configuration regardless of the underlying messaging provider, including WMQ. The configuration for stand-alone WMQ means that WAS is not involved.

WebSphere Application Server does not come with a pre-configured JMS provider. It is necessary to configure the service using the default messaging provider or WMQ. This document assumes some passing familiarity with enterprise Java such as JNDI contexts, factories and general connection requirements and parameters. This document also assumes that the BizTalk developer has access to particular information peculiar to the target WebSphere JMS implementation. Finally, this document assumes that the reader is knowledgeable and experienced with BizTalk Server.

Resources

- The user guide, *Using the JNBridge JMS Adapter for BizTalk Server*.
- This link is IBM documentation providing an overview and download of the stand-alone [WebSphere 6 JMS client](#) required by the JMS adapter. This link provides information about [installing and configuring](#) the stand-alone client. This link provides information on the [WebSphere 7/8 JMS thin client](#).

BizTalk Server Machine Prerequisites

The following prerequisites are needed for the adapter.

- The stand-alone WebSphere client can use either the IBM Java Run-time Environment (JRE) or the Oracle/Sun JRE. The IBM JRE ships with WebSphere. It can also be obtained by downloading the [IBM Eclipse Development](#) environment. Please contact IBM for additional licensing information regarding the IBM JRE. The Oracle/Sun JRE can be downloaded from the Sun website. Regardless of which JRE is used, it must be present on the BizTalk machine. Supported JRE versions are Java 5, 6 and 7. Java 4 is not supported.

Using the JMS Adapter with WebSphere

Configuring the BizTalk Server Machine

Installing the IBM WebSphere 6 JMS client

The JNBridge JMS Adapter for BizTalk uses the WebSphere 6 stand-alone JMS environment supplied by IBM (see *BizTalk Server Machine Prerequisites*). This JMS client can be downloaded as a single JAR file, `sibc_install-<version>.jar`, from the IBM website.

If the IBM JRE is being used, you must install the environment by executing this command:

```
java -jar sibc_install-<version>.jar jms_jndi_ibm [InstallDir]
```

If the Sun JRE is being used, use this command line:

```
java -jar sibc_install-<version>.jar jms_jndi_sun [InstallDir]
```

The installation supporting the IBM JRE will result in two JAR files, `sibc.jndi.jar` and `sibc.jms.jar`, being installed in `[InstallDir]\lib`. The installation supporting the Sun JRE will add a third JAR file, `sibc.orb.jar`.

Obtaining the IBM WebSphere 7/8 JMS client

Starting with WebSphere 7, IBM no longer supplies a download of the stand-alone JMS client. The client JAR files must be obtained from the WebSphere 7 or 8 installation, e.g., `C:\Program Files\IBM\WebSphere\AppServer\runtimes` (see *BizTalk Server Machine Prerequisites*).

Configuring the Sun ORB

If the Sun JRE is being used, then a required J2EE system property configuring the Sun Object Request Broker (ORB) must be supplied to the Sun JVM. If the IBM JRE is being used, skip this configuration step. Arguments are passed to the Sun JVM by the adapter using the Transport Handler Property JVM Arguments. Put this argument in the property JVM Arguments (see Figure 1).

```
-Dcom.ibm.CORBA.ORBInit=com.ibm.ws.sib.client.ORB
```

Using the JMS Adapter with WebSphere

Deploy JMS Header Schema to BizTalk Application

In order to properly handle JMS header properties within BizTalk, you must deploy the assembly, `JNBridgeBTS2006JMSProperties.dll`, to your BizTalk application. This assembly contains the XSD namespaces and schemas used by the JNBridge JMS Adapter to promote JMS header properties within messages stored in the BizTalk Message Box.

! *Deploying this assembly is mandatory.*

■ To deploy the schema assembly

- 1 Open up the BizTalk Administrator and open your BizTalk application in the left-side tree view.
- 2 Right click on the application's root node and choose **Add ► Resources**. This opens the **Add Resources** dialog.
- 3 In the dialog, click on the **Add** button and navigate to the schema DLL in the adapter install directory, e.g. `C:\Program Files\JNBridge\JMSAdapters\BTS2006\bin\JNBridgeBTS2006JMSProperties.dll`.
- 4 Click on **OK** to close the **Add Resources** dialog.
- 5 Open the **Schemas** folder in your application. You should see the three schemas:
`JMSSendProperty.SendPropertySchema`,
`JMSRecvProperty.RecvPropertySchema`
and `JMSConfProperty.ConfPropertySchema`.
- 6 Restart the host instance and application.

Configuring the Adapter Send and Receive Transport Handlers

The transport handler property grids for the Send and Receive sides contain properties global to all send or receive ports configured to use the JNBridge JMS Adapter and that reside in the same BTS host instance. In other words, all JMS Adapter send or receive ports in the BTS host instance will inherit these transport properties. You must configure send handler transport properties in order to produce messages to queues and topics. Likewise, you must configure receive handler transport properties in order to consume messages from queues and topics. In most cases, the values of the properties will be identical between the send and receive handlers; however, depending on the JMS server implementation, they may be different.

JMS Properties Category

The JMS Properties category are properties used to properly connect to a JMS server.

■ **JMS Acknowledge Mode**

The Acknowledge Mode is a drop-down list containing the JMS specification that determines how a JMS client and server institute a reliable messaging protocol. The choices are `AUTO_ACKNOWLEDGE`, `CLIENT_ACKNOWLEDGE` and `DUPS_OK_ACKNOWLEDGE`. Regardless of the choice, the JNBridge JMS Adapter will correctly implement the protocol. For the WebSphere JMS service, [AUTO_ACKNOWLEDGE](#) is the default configuration.

Using the JMS Adapter with WebSphere

■ Initial Context Factory

This is a text-editable field containing the name of the JNDI initial context factory. The initial context factory is a JNDI class used to locate and instance factories and JMS destinations. The default initial context factory for WebSphere Application Server is:

```
com.ibm.websphere.naming.WsnInitialContextFactory
```

If WebSphere MQ is being used stand-alone, the default initial context factory is:

```
com.sun.jndi.fscontext.JnbBtsRefFSContextFactory
```

! *Factory names are case sensitive—be sure the name (including the complete namespace, if necessary) is typed correctly.*

■ JMS Scheme

This is a text-editable field. The JMS Scheme or *Protocol* is particular to each vendor's RMI implementation. The protocol is part of the URI used to connect to the JMS service.

For WebSphere Application Server, the scheme is:

```
iiop.
```

If WebSphere MQ is being used stand-alone, the scheme is:

```
fscontext
```

■ JMS Version

This property tells the adapter which JMS implementation to expect when it loads the vendor's client stack—the JAR files in the Class Path property. The drop-down list contains two choices, 'JMS 1.1' and 'JMS 2.0'.

■ JMS Security Mode

The JMS Security Mode is a drop-down list that specifies the type of security required by the JMS server implementation. The choices are none, simple and strong. If the choice is simple, then the server expects a user name and password. If the choice is strong, then the server expects a digital certificate security strategy, usually based on the X.509 standard.

! *If the JMS server implements simple security, it is not necessary to configure this property. Enter a user name and password—the JNBridge JMS Adapter will automatically switch to the simple security mode.*

■ Queue Connection Factory

This is a text-editable field. There are no pre-configured, default connection factories in WebSphere.

Using the JMS Adapter with WebSphere

! *This value includes the JNDI directory path to the connection factory. The complete JNDI path must be provided.*

■ Topic Connection Factory

This is a text-editable field. There are no pre-configured, default connection factories in WebSphere.

! *This value includes the JNDI directory path to the connection factory. The complete JNDI path must be provided.*

JNBridge Properties Category

The JNBridge Properties Category correctly configure the .NET-to-Java interoperability core components.

■ Class Path

The Java class path is a set of semicolon-separated paths to the JAR or class files required for a JMS client installation. The class path is used by the JNBridge Java and .NET interoperability components to locate the JMS and JNDI client Java classes so they can be instantiated in the Java Virtual Machine.

To edit the class path, click in the field to enable the browse button. Click on the button to launch the Edit Class Path dialog. Note that only checked elements will be added to the class path when the dialog is dismissed.

For WebSphere Application Server 6

IBM JRE: `sibc.jndi.jar, sibc.jms.jar`

Oracle/Sun JRE: `sibc.jndi.jar, sibc.jms.jar, sibc.orb.jar`

For WebSphere Application Server 7 and 8 (for version 8, use JAR files with ‘_8.x.x’):

IBM JRE:

`com.ibm.ws.sib.thin.jms_7.0.0.jar`

`com.ibm.ws.ejb.thinclient_7.0.0.jar`

Oracle/Sun JRE:

`com.ibm.ws.sib.thin.jms_7.0.0.jar`

`com.ibm.ws.ejb.thinclient_7.0.0.jar`

`com.ibm.ws.orb_7.0.0.jar`

For WebSphere MQ stand-alone 7 and 8:

`com.ibm.mqjms.jar`

Using the JMS Adapter with WebSphere

`com.ibm.mq.jar`

`jnbjms-fscontext.jar`

Note that the WMQ JAR file, `com.ibm.mqjms.jar`, contains an embedded classpath that points to several other JAR files in the complete WMQ installation. This classpath changes with each release of WMQ. A JAR file is just a zip archive of a directory structure, so the manifest (META-INF/MANIFEST.MF) can be opened with a good zip utility in order to look at the classpath, or change qualifier ‘.jar’ to ‘.zip’ and use Windows Explorer. By doing this, you can scrape the necessary JAR files from the WMQ install (`[WMQ InstallDir]\java\lib`) and place them, along with the three above, on the client machine.

The JAR file shipped with the adapter, `jnbjms-fscontext.jar`, can be found in the **support** directory in the zip archive `RefFSContextFC.zip`.

■ JVM Path

The JVM Path property is the absolute path to the Java Virtual Machine implementation, `jvm.dll`. To edit the JVM Path property, click in the field to enable the browse button. Click on the button to launch the standard File Open dialog. Navigate to `jvm.dll` and click OK. Please see the section *Quick Config for WebSphere* for information regarding using the Oracle/Sun or IBM JRE and which JVM to use.

■ JVM Argument

For WMQ (only), please note the use of forward strokes:

```
-Djava.naming.provider.url=file:/C:/[PathToRefFSContextBindingFile]
```

Security Properties Category

This category provides security credentials necessary to connect to a JMS server, if the JMS implementation supports security mode simple.

■ Password

Click in this field to drop-down the password edit field. Type in the password.

■ User Name

This is a text editable field. Enter the user name necessary to connect to the JMS server.

Configuring Send Ports and Receive Locations

Configuring Send Ports

Connection Properties Category

These properties determine where the JMS server resides and the port number where it is listening for connections.

- Host Name

This is a text-editable field. Enter the name or IP address of the machine hosting the JMS server.

- Port Number

This is a text-editable field. Enter the port number on which the JMS server is accepting connections. For WebSphere, this port is usually, by default, **2809**. For stand-alone WebSphere MQ, the default port number is **1414**.

- Proprietary Connection String

This is a text-editable field. This property is only used if the JMS implementation uses complex URLs containing query expressions, or some proprietary connection string, that can not be constructed from the Host Name and Port Number properties. If this property contains a value, then the Host Name and Port Number properties will be ignored.

JMS Operation Properties Category

These properties determine what operation the send port will enable.

- JMS Object Name

This is a text-editable field. Enter the JNDI name bound to the JMS queue or topic. For example, a complete JNDI name for a queue or topic could be:

```
myQ  
west_coast/jms/partnerTopic
```

- JMS Object Type

This is a drop-down list containing two values: Queue or Topic.

- Message Type

This is a drop-down list containing the values: Text, Text UTF, Text ISO-8859-15 or Bytes.

If `Text` is chosen, then the JNBridge JMS Adapter will send a JMS Text Message. Because a JMS Text Message is by definition UTF-16 BE, the types `Text UTF` and `Text ISO-8859-15` ensure that the string which is the payload of the JMS message is correctly encoded from the binary representation in the BizTalk Message Box. If the type `Text` is chosen, then the binary representation is considered UTF-8. If `Bytes` is chosen, then the JNBridge JMS Adapter will send a JMS Bytes Message.

Configuring Receive Ports and Locations

Connection Properties Category

These properties determine where the JMS server resides and the port number where it is listening for connections.

- **Host Name**

This a text-editable field. Enter the name or IP address of the machine hosting the JMS server.

- **Port Number**

This is a text-editable field. Enter the port number on which the JMS server is accepting connections. For WebSphere, this port is usually, by default, **2809**. For stand-alone WebSphere MQ, the default port number is **1414**.

- **Proprietary Connection String**

This is a text-editable field. This property is only used if the JMS implementation uses complex URLs containing query expressions, or some proprietary connection string. that can not be constructed from the Host Name and Port Number properties. If this property contains a value, then the Host Name and Port Number properties will be ignored.

JMS Operation Properties Category

These properties determine what operation type of operation the send port will enable.

- **JMS Object Name**

This is a text-editable field. Enter the JNDI name bound to the JMS queue or topic. For WebSphere 6.1, the example topic is at:

```
myQ  
west_coast/jms/partnerTopic
```

Using the JMS Adapter with WebSphere

■ JMS Object Type

This is a drop-down list containing two values: Queue, Topic or SharedTopic.

■ Message Type

This is a drop-down list containing the values: Text, Text UTF-8, Text UTF-16, Text ISO-8859-15, Bytes or Map. If Text is chosen, then the JNBridge JMS Adapter expects to receive a JMS Text Message. If Bytes is chosen, then the JNBridge JMS Adapter expects to receive a JMS Bytes Message. If Map is chosen, then the JNBridge JMS Adapter expects to receive a JMS Map Message. Because a JMS Text Message by definition contains UTF-16 BE text, the types Text UTF-8, Text UTF-16 and Text ISO-8859-15 ensure that the text in the body of the message is encoded correctly to binary for submittal to the BizTalk Message Box. The type Text without a qualifier means UTF-8.

■ Client ID

This is a unique string that identifies the receive port connection to WebSphere. It is only used if durable subscriptions are enabled.

■ Durable Subscription Name

Durable subscriptions are particular to topics only. A durable subscription for a topic allows consumers to register a name with the JMS server such that whenever a receive port is active, all messages in the topic will be received. In this way, a receive port does not have to be continually connected to receive messages from a topic. A receive port that does not use durable subscriptions must be active and connected in order to subscribe to a topic—any messages published by the topic while a nondurable receive port is not active will not be available to that receive port when it becomes active. This is a text-editable field. Enter the durable subscription name.

■ Message Selector Filter

Message selectors are used by receive ports to filter or select messages from topics and queues based on JMS and custom message header properties.

This is a text-editable field. Enter in a selector expression. The expression is derived from a subset of the SQL92 standard.

JNDI Names

Figure 2 shows the WebSphere 6.1 Integrated Solutions Console. The console is displaying the configured JMS queues. Note that there are two configured queues, one each for two different JMS providers. The queue, `test_Q`, is created under the default messaging provider (the JMS implementation that comes with WebSphere) and has the JNDI name, `myQ`. The queue, `MQ_Q`, is provided by MQSeries and has the JNDI name, `myMQ_Q`.

It is important to use complete JNDI paths, either forward slash ('/') or dot ('.') delineated, for connection factories and JMS destinations.

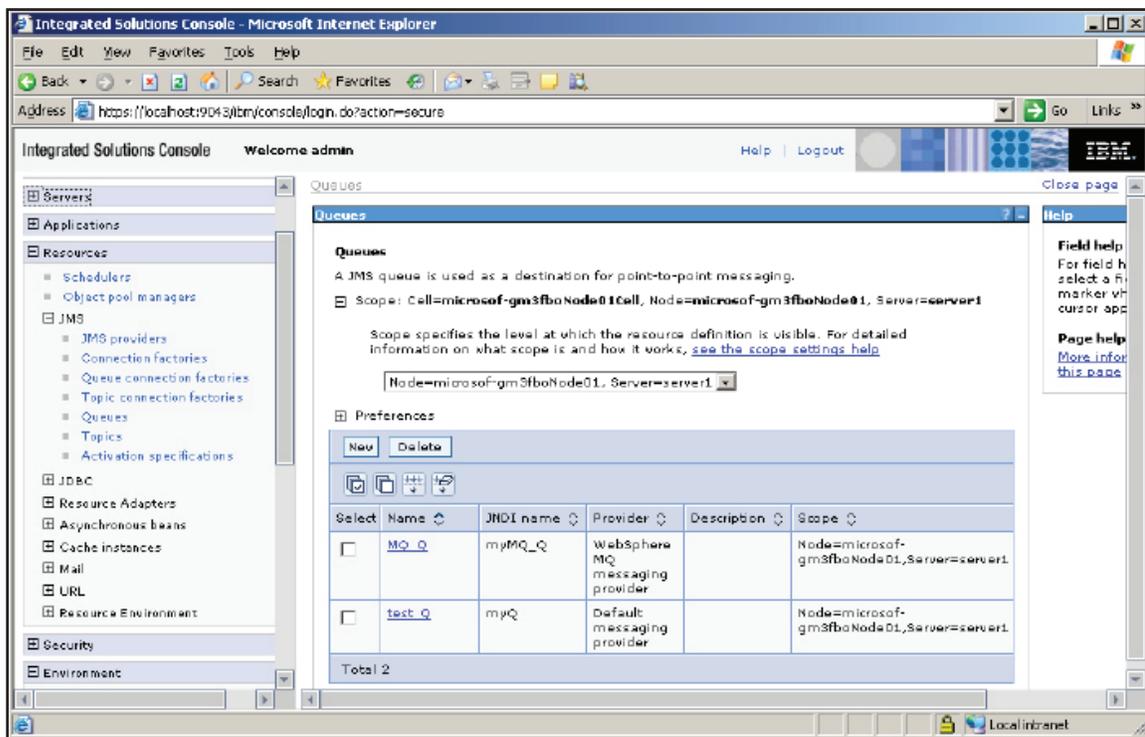


Figure 2. The JNDI names

Using the RefFSContextFactory JNDI with WebSphere MQ

Stand-alone WebSphere MQ uses a JNDI scaffolding based on the Sun *RefFSContextFactory* implementation. This implements a local JNDI store on each machine where the adapter is installed. Using the WMQ Explorer, the JNDI store can be created under the node, JMS Administered Objects, by using the menu choice, Add Initial Context. The JNDI store is a file, *.bindings*, that can be placed on each BizTalk machine where the adapter is installed.

To correctly use the RefFSContextFactory, please follow these directions.

1. Copy the **.bindings** file from the WMQ machine to the BizTalk machine.
2. Add the JAR file, **jnbjms-fscontext.jar**, to the BizTalk transport handler property, Class Path. This JAR file can be found in the adapter installation support directory in the zip archive, RefFSContextFC.zip.
3. For the BTS transport handler property, Initial Context Factory, use this ICF, **com.sun.jndi.fscontext.JnbBtsRefFSContextFactory**.
4. In the BTS transport handler property, JVM Arguments, point to the .bindings file directory (please note the forward strokes):
-Djava.naming.provider.url=file:/C:/[PathToRefFSContextBindingFile]