

Configuring JMS endpoints

- Configuration properties
 - Common adapter parameters
 - `jms.adaptor.ClientNames`
 - `jms.adaptor.ConnectionNames`
 - Connections parameters
 - `jms.adaptor.Connection.[ConnectionName].InitialContextFactory`
 - `jms.adaptor.Connection.[ConnectionName].ProviderURI`
 - `jms.adaptor.Connection.[ConnectionName].User`
 - `jms.adaptor.Connection.[ConnectionName].Password`
 - `jms.adaptor.Connection.[ConnectionName].ConnectionFactory`
 - `jms.adaptor.Connection.[ConnectionName].Reconnect`
 - `jms.adaptor.Connection.[ConnectionName].ReconnectTries`
 - `jms.adaptor.Connection.[ConnectionName].ReconnectInterval`
 - Clients parameters
 - `jms.adaptor.Client.[ClientName].ConnectionName`
 - `jms.adaptor.Client.[ClientName].StorageDir`
 - `jms.adaptor.Client.[ClientName].SessionType`
 - `jms.adaptor.Client.[ClientName].MessagingMode`
 - `jms.adaptor.Client.[ClientName].DestinationURI`
 - `jms.adaptor.Client.[ClientName].DeliveryMode`
 - `jms.adaptor.JMSTA.Client.[ClientName].TimeToLive`
 - `jms.adaptor.JMSTA.Client.[ClientName].SessionAckMode`
 - `jms.adaptor.JMSTA.Client.[ClientName].MessageType`
 - `jms.adaptor.Client.[ClientName].CustomMessageType`
 - `jms.adaptor.Client.[ClientName].Transacted`
 - `jms.adaptor.Client.[ClientName].TransactionBatchSize`
 - `jms.adaptor.Client.[ClientName].QueueSize`
 - `jms.adaptor.Client.[ClientName].QueuePersistent`
 - `jms.adaptor.Client.[ClientName].QueueNormalFileSize`
 - `jms.adaptor.Client.[ClientName].QueueMaxFileSize`
 - `jms.adaptor.Client.[ClientName].DurableSubscription`
 - `jms.adaptor.Client.[ClientName].DurableSubscriptionName`
 - `jms.adaptor.Client.[ClientName].ConnectionsCount`
 - `jms.adaptor.Client.[ClientName].ThreadsPerConnection`
 - `jms.adaptor.Client.[ClientName].maskPasswordsInStorages`
 - `jms.adaptor.Client.[ClientName].startOnload`
 - `jms.adaptor.Client.[ClientName].startTime`
 - `jms.adaptor.Client.[ClientName].stopTime`
 - `jms.adaptor.Client.[ClientName].scheduleTimeZone`
- Sample Configuration
 - Sample configuration for ActiveMQ vendor
 - Configuration for TIBCO EMS vendor
 - Configuration for RedHat AMQ vendor

The Java Message Service (JMS) is a messaging standard that allows application components based on the Java Platform Enterprise Edition (Java EE) to create, send, receive, and read messages. It enables distributed communication that is loosely coupled, reliable, and asynchronous.

As per FIXEdge Java, JMS endpoints allow you to establish a connection with JMS-message providers and receive/send data from/to them. JMS endpoints can work with queues and topics. Each endpoint has a unique name. A JMS endpoint is routing messages only in one way: it can be configured either producer or consumer.

During FIXEdge Java server start-up, it initiates endpoints that are defined in **jms-adaptor.properties**. The list of active endpoints is defined by the `jms.adaptor.ClientNames` property. If `jms.adaptor.ClientNames` is empty, none of the clients are active.

To add a JMS endpoint to **jms-adaptor.properties** file, you should perform the following actions:

1. Define the JMS connection configuration. JMS connection properties have the `jms.adaptor.Connection.[ConnectionName]` prefix, where `[ConnectionName]` is a unique name of this connection and it is used to link it with JMS endpoint configuration. The JMS connection has a set of predefined properties (specified in the table below) but can be extended with custom ones (see the samples below).
2. Define the JMS endpoint configuration and add its name to the `jms.adaptor.ClientNames` list.

Configuration properties

The JMS adaptor is configured by means of the following properties:

Property Name	Description	Required	Default Value
Common adapter parameters			

jms.adaptor. ClientNames	A comma-delimited list of JMS endpoints. A separate configuration section for each listed client should be specified.	Y	
jms.adaptor. ConnectionNames	A comma-delimited list of TA connections. A separate configuration section for each listed connection should be specified.	Y	
Connections parameters			
jms.adaptor. Connection. [ConnectionName]. InitialContextFactory	JNDI option. The fully qualified class name of the factory class that will create the initial context. An initial context is the starting point for naming operations.	C, required if JNDI mechanism is used	
jms.adaptor. Connection.[ConnectionName]. ProviderURI	JMS provider URI that defines where the created connection is to connect to as well as the protocol that should be used, for example, TCP/IP. Additionally, configuration information can be encoded in the URI.	Y	
jms.adaptor. Connection.[ConnectionName]. User	User name	Y	
jms.adaptor. Connection.[ConnectionName]. Password	User password	Y	
jms.adaptor. Connection. [ConnectionName]. ConnectionFactory	JNDI option. The connection factory object name in the JNDI objects store.	C, required if JNDI mechanism is used	
jms.adaptor. Connection. [ConnectionName]. Reconnect	Enables or disables the reconnect procedure for connection restore. This parameter is used for JMS brokers that don't support the failover mechanism.	N	false
jms.adaptor. Connection. [ConnectionName]. ReconnectTries	Number of reconnect tries or -1 for an infinite number of attempts	N	3
jms.adaptor. Connection. [ConnectionName]. ReconnectInterval	Fixed interval in milliseconds between reconnection attempts	N	2000
Clients parameters			
jms.adaptor. Client.[ClientName]. ConnectionName	Name of the primary connection used by the client. The connection should be registered in the <code>ConnectionNames</code> enumeration and has all required parameters.	Y	
jms.adaptor. Client.[ClientName]. StorageDir	Directory where the persistence file is stored in case of any communication problem.	Y, if the persistent mode is enabled	

jms.adaptor. Client.[ClientName]. SessionType	Session type: <ul style="list-style-type: none"> Producer - the session is a message producer Consumer - the session is a message consumer 	Y	
jms.adaptor. Client.[ClientName]. MessagingMode	Session messaging mode: Queue - Point-To-Point. Each message is addressed to a specific queue, and receiving clients extract messages from the queue(s) established to hold their messages. Queues retain all messages sent to them until the messages are consumed or until the messages expire. Topic - Publish/Subscribe. In a pub/sub product or application, clients address messages to a topic. Publishers and subscribers are generally anonymous and may dynamically publish or subscribe to the content hierarchy. The system takes care of the distributing messages arriving from a topic's multiple publishers to its multiple subscribers. Topics retain messages only as long as it takes to distribute them to current subscribers.	Y	
jms.adaptor. Client.[ClientName]. DestinationURI	URI of session destination (queue/topic name)	Y	
jms.adaptor. Client.[ClientName]. DeliveryMode	Specifies whether the sent messages are lost if the JMS provider fails. Persist - instructs the JMS provider to take extra care to ensure that a message is not lost in transit in case of JMS provider failure. A message sent with this delivery mode is logged to a stable storage when it is sent. NoPersist - does not require the JMS provider to store the message or otherwise guarantee that it is not lost if the provider fails. NOTE: used only for Producer sessions.	Y	
jms.adaptor. JMSTA.Client. [ClientName]. TimeToLive	This value defines a message expiration time (in milliseconds) that is the sum of the message's time-to-live and the GMT when it is sent (for transacted sends, this is the time at which the client sends the message, not the time the transaction is committed). Setting the parameter value to "0" will lead to infinite TTL.	Y	3000
jms.adaptor. JMSTA.Client.[ClientName]. SessionAckMode	Defines a mode in which the JMS session will acknowledge the messages that it receives and dispatches. Auto - With this acknowledgment mode, the session automatically acknowledges a client's receipt of a message either when the session has successfully returned from a call to receive or when the message listener the session has called to process the message successfully returns. Client - With this acknowledgment mode, the client acknowledges a consumed message by calling the message's acknowledge method. Acknowledging a consumed message acknowledges all messages that the session has consumed. When the client acknowledgment mode is used, a client may build up a large number of unacknowledged messages while attempting to process them. DupsOk - This acknowledgment mode instructs the session to lazily acknowledge the delivery of messages. This is likely to result in the delivery of some duplicate messages if the JMS provider fails, so it should only be used by consumers that can tolerate duplicate messages. The use of this mode can reduce session overhead by minimizing the work the session does to prevent duplicates.	Y	
jms.adaptor. JMSTA.Client.[ClientName]. MessageType	JMS Message type used for the session. Bytes - A stream of interpreted bytes. This message type is for literally encoding a body to match the existing message format. Text - Data is stored as a string. This message type is useful for exchanging simple text messages and for more complex character data, such as XML documents. Custom - The ability to use your own message type. This message type is useful for adaptation to already existing systems. For this message type, the <i>jms.adaptor.Client.[ClientName].CustomMessageType</i> property is required.	Y	
jms.adaptor. Client. [ClientName]. CustomMessageType	Class of a custom message type implementation. This should be the implementation of one of the interfaces: - <i>com.epam.fixengine.jms.client.consumer.IFromMessageConverter</i> - <i>com.epam.fixengine.jms.client.consumer.IToMessageConverter</i> NOTE: used only for Custom message type	Y, if MessageTy pe is Custom	
jms.adaptor. Client.[ClientName]. Transacted	Sending/consuming of messages in the session is transacted. true - transacted. false - not transacted. The value is used by default.	N	false
jms.adaptor. Client. [ClientName]. TransactionBatchSize	Max number of messages for one transaction. Takes the available number of messages from the queue but not more than that specified in this property. NOTE: used only if transacted is enabled.	N	20

jms.adaptor. Client. [ClientName]. QueueSize	Max number of messages in the outgoing endpoint queue. Messages are queued until successfully sent. Waits for space to become available if the queue is full.	N	100
jms.adaptor. Client. [ClientName]. QueuePersistent	Store messages in the file until it is committed. It allows restoring and sending messages after a sudden cardiac application. true – persistent queue. false – in-memory queue. Faster but less safe; some messages may be lost after restart.	N	true
jms.adaptor. Client. [ClientName]. QueueNormalFile Size	After reaching this size, the endpoint waits when the queue will be empty to truncate the file. File size in Mbytes. NOTE: used only for the persistent queue.	N	10
jms.adaptor. Client. [ClientName]. QueueMaxFileSi ze	After reaching this size, the endpoint truncates the file and overwrites the messages that are queued. File size in Mbytes. NOTE: used only for the persistent queue.	N	50
jms.adaptor. Client.[ClientNa me]. DurableSubscrip tion	Durable topic subscriptions allow receiving messages published while the subscriber is not active. Durable subscriptions offer the reliability of queues to the publish/subscribe message domain. See more here . true - Subscription is durable. false - Subscription is not durable. NOTE: <ul style="list-style-type: none">• used only for Consumer sessions.• checks that Producer sends messages using the Persistent delivery mode.	N	false
jms.adaptor. Client.[ClientNa me]. DurableSubscrip tionName	Durable subscription name.	Y, required if DurableSub scription set to <i>true</i> .	
jms.adaptor. Client. [ClientName]. ConnectionsCou nt	A number of JMS connections. The endpoint will open a given amount of connections to the JMS server for the given session to send simultaneously. NOTE: If a number of connections is greater than 1, there is no guarantee that messages will be placed into JMS in the same order they were received by the endpoint.	N	1
jms.adaptor. Client. [ClientName]. ThreadsPerCon nection	A number of threads per one JMS connection. The endpoint will open a given amount of JMS sessions for each JMS connection to send simultaneously. NOTE: If the number of threads per connection is greater than 1, there is no guaranty that messages will be placed into JMS in the same order they were received by the adapter.	N	1
jms.adaptor. Client. [ClientName]. maskPasswords InStorages	Enable/disable passwords obfuscating. When enabled, the Password (tag 554) value in the Logon(35=A) message will be substituted with a mask "*****".	N	true
jms.adaptor. Client. [ClientName]. startOnload	Whether a JMS endpoint should be started during FIXEdge Java server initialization.	N	true
jms.adaptor. Client. [ClientName]. startTime	A cron expression that defines a JMS endpoint start time.	N	

jms.adaptor. Client. [ClientName]. stopTime	A cron expression that defines a JMS endpoint stop time.	N	
jms.adaptor. Client. [ClientName]. scheduleTimeZo ne	A time zone for the start and stop times	N	

NOTE: All changes in the properties file are applied only after FixEdge Java server restart.

NOTE: All defined JMS endpoints are started automatically during the FIXEdge Java server initialization by default if there is a defined schedule for them (starting from FEJ 1.8.0). Schedule may be defined by `startTime/stopTime` properties or within a separate schedule configuration (Refer to the [Scheduler](#) section for more details).

NOTE: It is possible to use **environment variables** in configuration. Example: `jms.adaptor.Connection.[ConnectionName].Password = ${CONNECTION_PASSWORD}`, where `CONNECTION_PASSWORD` is the name of the environment variable.

Sample Configuration

Sample configuration for ActiveMQ vendor

Make sure that `activemq-client.jar` is present in the `/lib` directory.

Typical JMS endpoint parameters in `jms-adaptor.properties`:

jms-adaptor.properties

```
jms.adaptor.ConnectionNames =LocalActiveMQConnection
jms.adaptor.ClientNames = ActiveMQProducer, ActiveMQConsumer

# ActiveMQ Connection definition
jms.adaptor.Connection.LocalActiveMQConnection.InitialContextFactory = org.apache.activemq.jndi.
ActiveMQInitialContextFactory
jms.adaptor.Connection.LocalActiveMQConnection.ProviderURI = tcp://localhost:61616
jms.adaptor.Connection.LocalActiveMQConnection.User =
jms.adaptor.Connection.LocalActiveMQConnection.Password =
jms.adaptor.Connection.LocalActiveMQConnection.ConnectionFactory = queueConnectionFactory
jms.adaptor.Connection.LocalActiveMQConnection.Reconnect = true
jms.adaptor.Connection.LocalActiveMQConnection.ReconnectTries = 3
jms.adaptor.Connection.LocalActiveMQConnection.ReconnectInterval = 500

# ActiveMQ Producer definition
jms.adaptor.Client.ActiveMQProducer.ConnectionName = LocalActiveMQConnection
jms.adaptor.Client.ActiveMQProducer.StorageDir = ./logs
jms.adaptor.Client.ActiveMQProducer.SessionType = Producer
jms.adaptor.Client.ActiveMQProducer.MessagingMode = Queue
jms.adaptor.Client.ActiveMQProducer.DestinationURI = MyQueue
jms.adaptor.Client.ActiveMQProducer.DeliveryMode = NoPersist
jms.adaptor.Client.ActiveMQProducer.TimeToLive = 100000
jms.adaptor.Client.ActiveMQProducer.SessionAckMode = Auto
jms.adaptor.Client.ActiveMQProducer.MessageType = Text
jms.adaptor.Client.ActiveMQProducer.Transacted = true
jms.adaptor.Client.ActiveMQProducer.DurableSubscription = false
jms.adaptor.Client.ActiveMQProducer.DurableSubscriptionName = DurableTest

# ActiveMQ Consumer definition
jms.adaptor.Client.ActiveMQConsumer.ConnectionName = LocalActiveMQConnection
jms.adaptor.Client.ActiveMQConsumer.StorageDir = ./logs
jms.adaptor.Client.ActiveMQConsumer.SessionType = Consumer
jms.adaptor.Client.ActiveMQConsumer.MessagingMode = Queue
jms.adaptor.Client.ActiveMQConsumer.DestinationURI = MyQueue
jms.adaptor.Client.ActiveMQConsumer.DeliveryMode = NoPersist
jms.adaptor.Client.ActiveMQConsumer.TimeToLive = 100000
jms.adaptor.Client.ActiveMQConsumer.SessionAckMode = Auto
jms.adaptor.Client.ActiveMQConsumer.MessageType = Text
jms.adaptor.Client.ActiveMQConsumer.Transacted = false
jms.adaptor.Client.ActiveMQConsumer.DurableSubscription = false
jms.adaptor.Client.ActiveMQConsumer.DurableSubscriptionName = DurableTest
```

NOTE: `jms.adaptor.Connection.LocalActiveMQConnection.ProviderURI = tcp://localhost:61616` - *host:port* of Apache ActiveMQ Message Broker.

Configuration for TIBCO EMS vendor

Copy the appropriate TIBCO JMS client library (**tibjms.jar**) from the TIBCO EMS installation **/lib** directory.

Typical JMS consumer endpoint parameters in **jms-adaptor.properties**:

jms-adaptor.properties

```
jms.adaptor.ConnectionNames =TibcoJMSConnection
jms.adaptor.ClientNames = TibcoConsumer

# JMS Connection definition
jms.adaptor.Connection.TibcoJMSConnection.InitialContextFactory = com.tibco.tibjms.naming.
TibjmsInitialContextFactory
jms.adaptor.Connection.TibcoJMSConnection.ProviderURI = tcp://localhost:7222
jms.adaptor.Connection.TibcoJMSConnection.User =dev
jms.adaptor.Connection.TibcoJMSConnection.Password =dev
jms.adaptor.Connection.TibcoJMSConnection.ConnectionFactory = TopicConnectionFactory
jms.adaptor.Connection.TibcoJMSConnection.ReconnectTries = 2000000
jms.adaptor.Connection.TibcoJMSConnection.ReconnectInterval = 2000
jms.adaptor.Connection.TibcoJMSConnection.Reconnect = true

# Topic Consumer client
jms.adaptor.Client.TibcoConsumer.ConnectionName = TibcoJMSConnection
jms.adaptor.Client.TibcoConsumer.SessionType = Consumer
jms.adaptor.Client.TibcoConsumer.MessagingMode = Topic
jms.adaptor.Client.TibcoConsumer.DestinationURI = fixededge.deals
jms.adaptor.Client.TibcoConsumer.DeliveryMode = NoPersist
jms.adaptor.Client.TibcoConsumer.TimeToLive = 100000
jms.adaptor.Client.TibcoConsumer.SessionAckMode = Auto
jms.adaptor.Client.TibcoConsumer.MessageType = Text
jms.adaptor.Client.TibcoConsumer.StorageDir = ./logs
jms.adaptor.Client.TibcoConsumer.DurableSubscription = true
jms.adaptor.Client.TibcoConsumer.DurableSubscriptionName = fixededge.deals
```

Configuration for RedHat AMQ vendor

Install appropriate RHEL [JMS Client](#) (copy client libraries to the `/lib` directory).

Typical JMS consumer endpoint parameters in `jms-adaptor.properties`:

jms-adaptor.properties

```
#JMS Connection definition
jms.adaptor.ConnectionNames = Connection1

jms.adaptor.Connection.Connection1.InitialContextFactory = org.apache.qpid.jms.jndi.JmsInitialContextFactory
jms.adaptor.Connection.Connection1.ConnectionFactory = myFactoryLookup
jms.adaptor.Connection.Connection1.ConnectionFactory.myFactoryLookup = amqp://localhost:5672
jms.adaptor.Connection.Connection1.User = centos
jms.adaptor.Connection.Connection1.Password = centos

jms.adaptor.ClientNames = ProducerSession, ConsumerSession

# Poducer client
jms.adaptor.Client.ProducerSession.ConnectionName = Connection1
jms.adaptor.Client.ProducerSession.StorageDir = ./logs
jms.adaptor.Client.ProducerSession.SessionType = Producer
jms.adaptor.Client.ProducerSession.MessagingMode = Queue
jms.adaptor.Client.ProducerSession.DestinationURI = queue1
jms.adaptor.Client.ProducerSession.DeliveryMode = NoPersist
jms.adaptor.Client.ProducerSession.TimeToLive = 100000
jms.adaptor.Client.ProducerSession.SessionAckMode = Auto
jms.adaptor.Client.ProducerSession.MessageType = Text

# Consumer client
jms.adaptor.Client.ConsumerSession.ConnectionName = Connection1
jms.adaptor.Client.ConsumerSession.StorageDir = ./logs
jms.adaptor.Client.ConsumerSession.SessionType = Consumer
jms.adaptor.Client.ConsumerSession.MessagingMode = Queue
jms.adaptor.Client.ConsumerSession.DestinationURI = queue2
jms.adaptor.Client.ConsumerSession.DeliveryMode = NoPersist
jms.adaptor.Client.ConsumerSession.TimeToLive = 100000
jms.adaptor.Client.ConsumerSession.SessionAckMode = Auto
jms.adaptor.Client.ConsumerSession.MessageType = Text
```