



## Installing and Configuring Apache ActiveMQ With iWay Service Manager

This use case describes how to install and configure Apache ActiveMQ with iWay Service Manager (iSM).

---

### Prerequisites

Before continuing, ensure that you have the following components and settings available on your system:

- Apache ActiveMQ Version 5.1.x  
  
You can download Apache ActiveMQ from <http://activemq.apache.org/>.
- iWay Service Manager Version 7.0.x
- Java JDK Version 1.7 or higher
- Set the `JAVA_HOME` and `ACTIVEMQ_HOME` environment variables as follows:
  - `JAVA_HOME = C:\Java\jdk1.7.0_75`
  - `ACTIVEMQ_HOME = C:\apache-activemq-5.1x`

### Installing iWay Service Manager

For more information on installing iWay Service Manager (iSM), see the *iWay Installation and Configuration Guide*. You can access and download this documentation from the iWay Technical Library:

<http://ecl.informationbuilders.com/iway/index.jsp>

### Installing Apache ActiveMQ

To install Apache ActiveMQ:

1. Extract the Apache ActiveMQ files to a directory on your system. For example:

`C:\apache_activemq-5.1.x`

2. Copy the `activemq-all-5.13-SNAPSHOT.jar` file from `C:\apache_activemq-5.1.x` and paste it into the iWay classpath (`iway_home\lib`) directory. For example:

C:\iway7\lib

3. Open a command line prompt, navigate to the `ACTIVEMQ_HOME\bin` directory and type `activemq start`. For example:

```
C:\apache-activemq-5.1.x\bin>activemq start
```

You should see the following startup sequence:

```
INFO | Apache ActiveMQ 5.13-SNAPSHOT (localhost, ID:INFORMA-PNUJOK8-57364-14399  
17421865-0:1) started  
INFO | For help or more information please see: http://activemq.apache.org  
INFO | ActiveMQ WebConsole available at http://0.0.0.0:8161/
```

**Note:** You may see other log information surrounding these lines, which is normal.

4. To check and verify your Apache ActiveMQ installation, enter the following URL in your browser:

<http://localhost:8161>

5. Use the following user credentials to log on:

- User name: `admin`
- Password: `admin`

### Configuring Apache ActiveMQ Queues

To configure Apache ActiveMQ queues:

1. Navigate to the Apache ActiveMQ web console by entering the following URL in your browser:

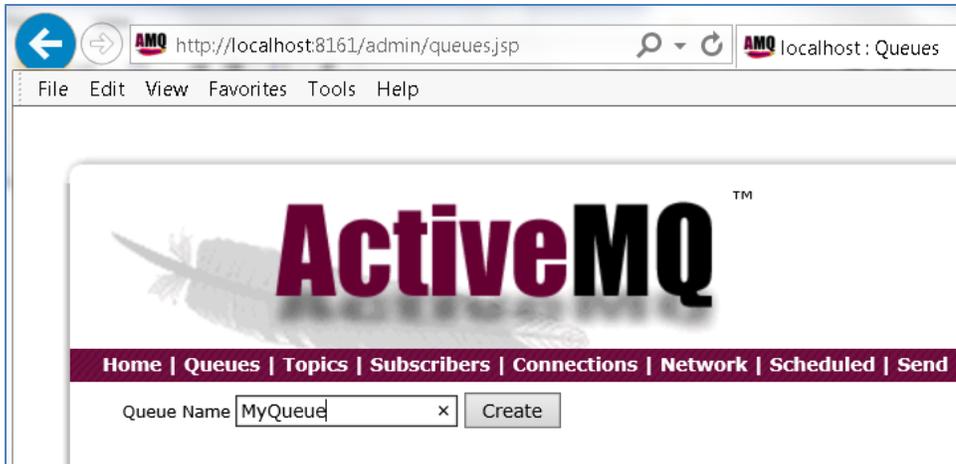
<http://localhost:8161>

2. Select *Manage ActiveMQ broker*.

If prompted, use the following user credentials:

- User name: `admin`
- Password: `admin`

3. Click *Queues* in the menu bar, as shown in the following image.



4. In the Queue Name field, type *MyQueue* and then click *Create*.
5. Repeat this procedure to create the following additional queues:
  - replyto
  - pend
  - out

### Configuring iWay Service Manager

To configure iWay Service Manager (ISM):

1. Start iSM as a service or use `iwsrv64.exe` from command line.
2. Navigate to the iSM Administration Console (<http://localhost:9999>) or use iWay Integration Tools (iIT).
3. Create the following components:
  - Channel (*ActiveMQ\_channel*)
  - Inlet (*ActiveMQ\_inlet*)
  - Route (*ActiveMQ\_route*), or just use the Move route
  - Outlet (*ActiveMQ\_outlet*)
4. Create a new *Java Message Service (jmsq)* listener.

**Listeners**  
Listeners are protocol handlers, that receive input for a channel from a configured endpoint. Listed below are references to the listeners that are defined in the registry.

**Select listener type**

Type \*      Type of the new listener

Select a type

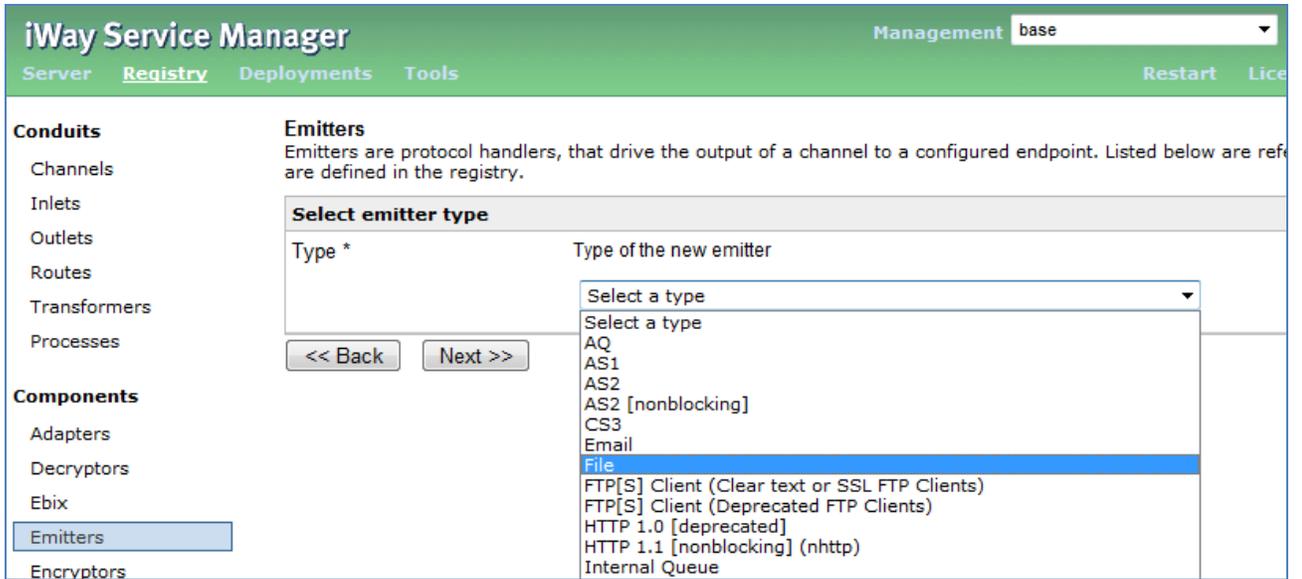
- FTP[S] Client (Deprecated FTP Clients)
- FTP[S] Server (Clear text or SSL FTP Server)
- HL7-MLLP-Listener
- HTTP 1.0 [deprecated]
- HTTP 1.1 [nonblocking] (nhttp)
- iEI
- Internal Queue
- Java Message Service (jmsq)**
- LDAP High Watermark/File
- LDAP Listener
- LogListener
- MLLP

Specify the following values for the *Java Message Service (jmsq)* listener parameters:

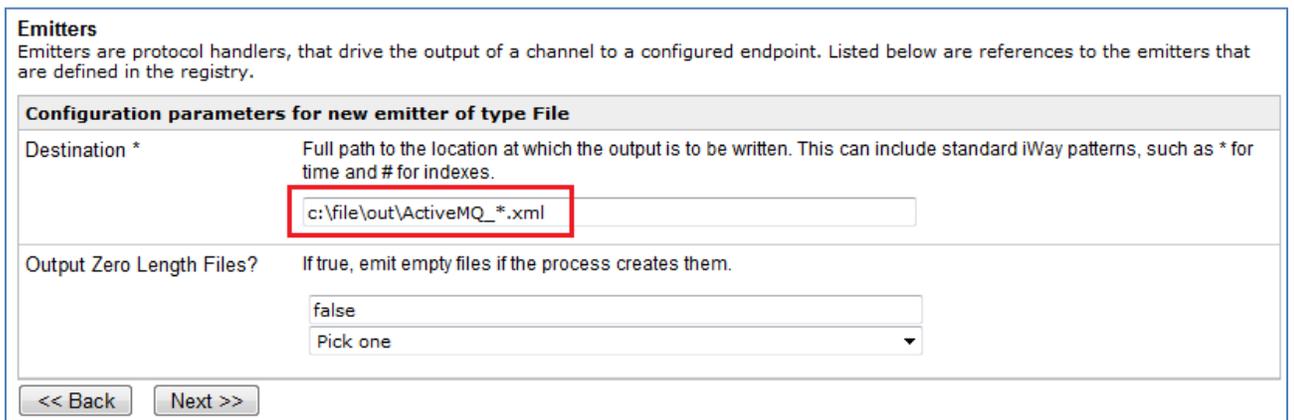
- Name = **ActiveMQ\_listener**
- Connection = **connectionFactory**
- JNDI Factory = **org.apache.activemq.jndi.ActiveMQInitialContextFactory**
- JNDI URL = **tcp://localhost:61616**
- Form of Input = **queue**
- Receiver Name = **MyQueue**
- Default Reply = **replyto**
- Userid = **admin**
- Password = **admin**
- JMSReply To = **out**
- Pending Queue = **pend**

**Note:** Leave the default values for all of the remaining parameters.

5. Create a new File emitter.



- Set the Destination field to `c:\file\out\ActiveMQ_*.xml`, as shown in the following image.



- Click *Next*.
- Enter `ActiveMQ_emitter` in the Name field, as shown in the following image.

**Emitters**  
Emitters are protocol handlers, that drive the output of a channel to a configured endpoint. Listed below are references to the emitters that are defined in the registry.

**Provide the name for the new emitter**

Name *	Name of the new emitter
	<input type="text" value="ActiveMQ_emitter"/>
Description	Description for the new emitter
	<input type="text"/>

<< Back    Finish

9. Click *Finish*.
10. Create a new channel (ActiveMQ\_channel) combining the following components (artifacts):
  - a. Add *ActiveMQ\_listener* to *ActiveMQ\_inlet*.
  - b. Add *Move* route to *ActiveMQ\_route* (ignore if you are already using the *Move* route).
  - c. Add *ActiveMQ\_emitter* to *ActiveMQ\_outlet*.
  - d. Build and deploy the channel (ActiveMQ\_channel).
11. Create a new *jni.properties* file in the iWay classpath (*iway\_home\lib*) directory.
12. Add the following lines to the *jni.properties* file:

```
java.naming.factory.initial = org.apache.activemq.jndi.ActiveMQInitialContextFactory
```

```
# use the following property to configure the default connector
```

```
java.naming.provider.url = vm://localhost
```

```
# use the following property to specify the JNDI name the connection factory
```

```
# should appear as.
```

```
#connectionFactoryNames = connectionFactory, queueConnectionFactory, topicConnectionFactry
```

```
connectionFactoryNames = QueueCF, connectionFactory, queueConnectionFactory,
```

```
topicConnectionFactory
```

```
# register some queues in JNDI using the form
```

```
# queue.[jndiName] = [physicalName]
```

```
queue.MyQueue = MyQueue
```

```
queue.replyto = replyto
```

```
queue.out = out
```

```
queue.pend = pend
```

```
# register some topics in JNDI using the form
# topic.[jndiName] = [physicalName]
topic.MyTopic = example.MyTopic
```

```
# END SNIPPET: jndi
```

**Note:** A sample *jndi.properties* file is packaged with this use case.

13. Restart iSM.

### Placing (PUT) a Message on an Apache ActiveMQ Queue

To test the iSM Apache ActiveMQ channel that you configured, you must PUT a message on the Receiver queue.

1. Navigate to the Apache ActiveMQ web console and click *Send* on the title bar.

The Send a JMS Message page opens, as shown in the following image.

The screenshot shows the Apache ActiveMQ web console interface. The browser address bar displays `http://localhost:8161/admin/send.jsp`. The page title is "Send a JMS Message". The "Message Header" section contains the following fields:

Destination	MyQueue	Queue or Topic	Queue
Correlation ID		Persistent Delivery	<input type="checkbox"/>
Reply To		Priority	
Type		Time to live	
Message Group		Message Group Sequence Number	
delay(ms)		Time(ms) to wait before scheduling again	
Number of repeats		Use a CRON string for scheduling	
Number of messages to send	1	Header to store the counter	JMSXMessageCounter

Buttons for "Send" and "Reset" are located below the form. The "Message body" section contains the text: `<test>MessageSuccessfullySent</test>`.

2. In the Destination field, type *MyQueue*.
3. In the Message Body area, enter the following:

<test>MessageSuccessfullySent</test>

4. Click *Send*.

You will now have one message in the *MyQueue* queue.

5. Start the iSM channel *ActiveMQ\_channel* that you previously configured.

The message you PUT on the Receiver queue is consumed and placed in the following directory:

`c:\file\out\`