



Fuse Message Broker Migration Guide

Version 5.5
February 2012

Migration Guide

Version 5.5

Updated: 27 Mar 2014

Copyright © 2012-2013 Red Hat, Inc. and/or its affiliates.

Trademark Disclaimer

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at <http://creativecommons.org/licenses/by-sa/3.0/>. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Apache, ServiceMix, Camel, CXF, and ActiveMQ are trademarks of Apache Software Foundation. Any other names contained herein may be trademarks of their respective owners.

Third Party Acknowledgements

One or more products in the Red Hat JBoss Fuse release includes third party components covered by licenses that require that the following documentation notices be provided:

- JLine (<http://jline.sourceforge.net>) jline:jline:jar:1.0

License: BSD (LICENSE.txt) - Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR

SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- Stax2 API (<http://woodstox.codehaus.org/StAX2>) org.codehaus.woodstox:stax2-api:jar:3.1.1

License: The BSD License (<http://www.opensource.org/licenses/bsd-license.php>)

Copyright (c) <YEAR>, <OWNER> All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- jibx-run - JiBX runtime (<http://www.jibx.org/main-reactor/jibx-run>) org.jibx:jibx-run:bundle:1.2.3

License: BSD (<http://jibx.sourceforge.net/jibx-license.html>) Copyright (c) 2003-2010, Dennis M. Sosnoski.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of JiBX nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON

ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- JavaAssist (<http://www.jboss.org/javassist>) org.jboss.javassist:com.springsource.javassist:jar:3.9.0.GA:compile
License: MPL (<http://www.mozilla.org/MPL/MPL-1.1.html>)
- HAPI-OSGI-Base Module (<http://hl7api.sourceforge.net/hapi-osgi-base/>) ca.uhn.hapi:hapi-osgi-base:bundle:1.2
License: Mozilla Public License 1.1 (<http://www.mozilla.org/MPL/MPL-1.1.txt>)

Table of Contents

- 1. Summary of Fuse Message Broker 5.4 to Fuse Message Broker 5.5 Migration 9
- 2. Unix Shell Scripts 11
- 3. Persistence Layer 13
- 4. Configuration File 15
- 5. Known Issues 19

List of Examples

4.1. Fuse Message Broker Blueprint configuraiton	15
--	----

Chapter 1. Summary of Fuse Message Broker 5.4 to Fuse Message Broker 5.5 Migration

Overview

This section summarizes the issues you need to consider when migrating from Fuse Message Broker 5.4 to Fuse Message Broker 5.5.

Product dependencies

Fuse Message Broker 5.5 has the following notable product dependencies:

- JDK 1.6.0_18 (or later) is now required.
- Spring 3.0.x is now required (if using Spring).
- Both Maven 2.2.1 and Maven 3.0 are now supported (for building the source).
- Camel is upgraded to version 2.7.1.

Migration procedure

When migrating to Fuse Message Broker 5.5, it is recommended that you follow these steps:

1. Update your environment variables to point at the new installation of Fuse Message Broker, as follows:

Windows O/S

Update your environment variables as described in ["Setting up the Windows Environment"](#) in *Managing and Monitoring a Broker*.

*NIX O/S

Run the `activemq setup` administration tool to generate a new startup script, as follows:

```
./activemq setup StartupScript
```

Where *StartupScript* is the path to either of the standard startup script locations, `/etc/default/activemq` and `/home/User/.activemqrc`. After generating the startup script, edit and

customize it as necessary. For more details, see ["Setting up the Unix/Linux/OS X Environment"](#) in *Managing and Monitoring a Broker*.

2. You can re-use your existing Fuse Message Broker XML configuration files. If you are migrating from a Fuse Message Broker version 5.3 or earlier, however, you might need to change the order of elements appearing inside the broker element, which are now required to appear in alphabetical order (as defined in the XML schema). The alphabetical ordering requirement applies to elements *at all nesting depths*.

For more details, see ["Configuration File"](#) on page 15.

3. To migrate KahaDB data files from earlier Fuse Message Broker versions, please observe the following points:
 - Always make backup copies of your existing KahaDB data files, before upgrading the broker.
 - Upgrading from 5.4.x to 5.5.0—when you start the 5.5 broker instance for the first time, the KahaDB auto-upgrade feature automatically upgrades the KahaDB data files.
 - Upgrading from 5.3.x to 5.5.0—due to a recent change in the format of the KahaDB index file, db.data, the KahaDB auto-upgrade feature cannot upgrade 5.3 index files. To upgrade from 5.3, therefore, delete the index and redo files, db.data and db.redo, before starting the 5.5 broker. Now, when you start the 5.5 broker, it recreates the index file as it starts up (if your database is large, this could take a long time, for example 20 to 30 minutes).
4. Study the rest of this *Migration Guide*, to check whether there are any other issues that could affect your system.

Chapter 2. Unix Shell Scripts

Single Unix shell

From version Fuse Message Broker 5.4.0 onwards, the **activemq** script incorporates the functionality of the **activemq-admin** script.

New options

The new **activemq** script adds the following commands:

- **start**—starts the broker in background. It saves process id in `${ACTIVEMQ_DATA_DIR}/activemq.pid` for future reference.
- **console**—starts the broker in foreground. It is the replacement for the old **activemq** script.
- **stop**—stops the broker that is running in the background. It tries first to stop the broker using shutdown task. If that task doesn't return in 30 seconds, it kills the process.
- **restart**—stops the running instance and starts a new one.
- **status**—checks to see if a broker is running.
- **setup**—create the specified configuration file for this script. The configuration of the script can be placed at `/etc/default/activemq` or `$HOME/.activemqrc`.

Chapter 3. Persistence Layer

Default persistence layer

Since Fuse Message Broker 5.4.0, the KahaDB message store is the default persistence layer used by Fuse Message Broker. The KahaDB message store is the latest evolution of Fuse Message Broker's native message store and it supersedes both the AMQ message store and the (original) Kaha message store.

If no configuration is specified for the KahaDB message store, an instance with default settings is automatically created for you.

Example KahaDB message store configuration

The KahaDB message store can be configured explicitly using the `kahaDB` element inside the `persistenceAdapter` element, as follows:

```
<broker brokerName="broker" persistent="true" useShutdownHook="false">
  ...
  <persistenceAdapter>
    <kahaDB directory="activemq-data"/>
  </persistenceAdapter>
</broker>
```

Reference

For more details about the KahaDB message store, see ["Using the KahaDB Message Store"](#) in *Configuring Broker Persistence*.

Chapter 4. Configuration File

Order of elements inside broker

In versions prior to 5.4.0, the element order inside the broker element was unimportant, because the XML configuration file was not validated. In version 5.5, however, the configuration file is validated against the Fuse Message Broker XML schema as it is loaded. This means that the element's inside the broker element *must* follow the order defined in the XML schema.



Tip

The element order is always alphabetical. The XML schema is generated by the XBeans utility, so alphabetical ordering is guaranteed.

Disabling schema validation

If you have some legacy configuration files that are not alphabetically ordered, you might prefer to skip the schema validation step. You can disable schema validation by setting the `validate=false` option on the xbean URL. For example, to launch a broker on a UNIX O/S using the configuration file, `conf/activemq.xml`, *without* performing schema validation, enter the following command:

```
./bin/activemq xbean:conf/activemq.xml?validate=false
```

Blueprint configuration file

In Fuse Message Broker 5.5, when deploying into the OSGi container, you have the option of embedding the broker XML inside a Blueprint configuration file, instead of using a Spring configuration file. The Fuse Message Broker Blueprint schema belongs to the following namespace:

```
http://activemq.apache.org/schema/blueprint
```

[Example 4.1 on page 15](#) shows an example of a Fuse Message Broker Blueprint configuration.

Example 4.1. Fuse Message Broker Blueprint configuration

```
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0"
  xmlns:cm="http://aries.apache.org/blueprint/xmlns/blueprint-cm/v1.0.0"
  xmlns:ext="http://aries.apache.org/blueprint/xmlns/blueprint-ext/v1.0.0"
  xmlns:amq="http://activemq.apache.org/schema/core">
```

```

<ext:property-placeholder />

<broker xmlns="http://activemq.apache.org/schema/core" brokerName="localhost" dataDir
ectory="${karaf.data}/activemq/localhost" useShutdownHook="false">

  <destinationPolicy>
    <policyMap>
      <policyEntries>
        <policyEntry topic="" producerFlowControl="true" memoryLimit="1mb">
          <pendingSubscriberPolicy>
            <vmCursor />
          </pendingSubscriberPolicy>
        </policyEntry>
        <policyEntry queue="" producerFlowControl="true" memoryLimit="1mb">
          </policyEntry>
        </policyEntries>
      </policyMap>
    </destinationPolicy>

    <managementContext>
      <managementContext createConnector="false"/>
    </managementContext>

    <persistenceAdapter>
      <kahaDB directory="${karaf.data}/activemq/localhost/kahadb"/>
    </persistenceAdapter>

    <transportConnectors>
      <transportConnector name="openwire" uri="tcp://localhost:61616"/>
      <transportConnector name="stomp" uri="stomp://localhost:61613"/>
    </transportConnectors>
  </broker>

  <bean id="activemqConnectionFactory" class="org.apache.activemq.ActiveMQConnectionFact
ory">
    <property name="brokerURL" value="tcp://localhost:61616" />
  </bean>

  <bean id="pooledConnectionFactory" class="org.apache.activemq.pool.PooledConnectionFact
ory">
    <property name="maxConnections" value="8" />
    <property name="connectionFactory" ref="activemqConnectionFactory" />
  </bean>

  <bean id="resourceManager" class="org.apache.activemq.pool.ActiveMQResourceManager"
init-method="recoverResource">
    <property name="transactionManager" ref="transactionManager" />
    <property name="connectionFactory" ref="activemqConnectionFactory" />
    <property name="resourceName" value="activemq.localhost" />

```



```
</bean>

<reference id="transactionManager" interface="javax.transaction.TransactionManager" />

<service ref="pooledConnectionFactory" interface="javax.jms.ConnectionFactory">
  <service-properties>
    <entry key="name" value="localhost"/>
  </service-properties>
</service>
</blueprint>
```


Chapter 5. Known Issues

Overview

Fuse Message Broker 5.5.1-fuse-00-xx has the following known issues:

- The Fuse Message Broker Web console does not work when the log4j rootLogger level is set to DEBUG (see [MB-805](http://fusesource.com/issues/browse/MB-805)¹).

A workaround is to leave the rootLogger level at INFO and set the activemq logger level to DEBUG instead. For example:

```
log4j.logger.org.apache.activemq=DEBUG
```

¹ <http://fusesource.com/issues/browse/MB-805>

