

# SOA-P 4.3.0.CP02

---

## HP SOA Registry Foundation Integration

JBESB-GS-8/11/09



## Legal Notices

The information contained in this documentation is subject to change without notice.

JBoss Inc. makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. JBoss Inc. shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Java™ and J2EE is a U.S. trademark of Sun Microsystems, Inc. Microsoft® and Windows NT® are registered trademarks of Microsoft Corporation. Oracle® is a registered U.S. trademark and Oracle9™, Oracle9 Server™ Oracle9 Enterprise Edition™ are trademarks of Oracle Corporation. Unix is used here as a generic term covering all versions of the UNIX® operating system. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

## Copyright

JBoss, Home of Professional Open Source Copyright 2006, JBoss Inc., and individual contributors as indicated by the @authors tag. All rights reserved.

See the copyright.txt in the distribution for a full listing of individual contributors. This copyrighted material is made available to anyone wishing to use, modify, copy, or redistribute it subject to the terms and conditions of the GNU General Public License, v. 2.0. This program is distributed in the hope that it will be useful, but WITHOUT A WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

See the GNU General Public License for more details. You should have received a copy of the GNU General Public License, v. 2.0 along with this distribution; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

## Software Version

SOA-P 4.3.0.CP02

## Restricted Rights Legend

Use, duplication, or disclosure is subject to restrictions as set forth in contract subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause 52.227-FAR14.

© Copyright 2009 JBoss Inc.

# Configuration

---

In order to communicate with the HP SOA Registry Foundation, you will have to configure JBoss ESB to communicate with the HP SOA Registry Foundation. In order to do this, you need to set up HP SOA Registry Foundation and then use HP SOA Registry Foundation as the registry for your JBoss ESB installation.

## Setup of HP SOA Registry Foundation

---

During installation of HP SOA Registry Foundation, the installation log will provide you with a set of instructions for installing HP SOA Registry inside SOA-P 4.3.0.CP02.

The following are instructions that are included within HP SOA Registry Foundation. If you have already applied these, then move on to “Further Setup”.

1. Unpack the created `/opt/hp/registry/conf/porting/jboss/build/registry.war` file into the `[context_name].war` subdirectory of the JBoss deployment directory, which is usually `JBOSS_HOME/server/[jboss_configuration]/deploy`

2. Modify the JBoss launch script (usually in `JBOSS_HOME/bin/run.sh`) as follows:

- \* Add the following jars to the beginning of the JBoss classpath:

  - `/opt/hp/registry/lib/security-ng.jar`

  - `/opt/hp/registry/conf/porting/dist/security3-ng.jar`

  - `JBOSS_HOME/server/[jboss_configuration]/lib/log4j.jar`

- \* Remove all JAXP variable settings from the script and remove `crimson.jar` from the JBoss classpath if these are present.

3. Enable security; edit the file:

  - `/opt/hp/registry/conf/porting/jboss/jaas.xml`

You may of course define your own JAAS configuration for the required module names.

4. Configure log4j for HP SOA Registry; add lines to the file `JBOSS_HOME/server/[jboss_configuration]/conf/jboss-log4j.xml` from this template:

  - `/opt/hp/registry/conf/porting/jboss/log4j.xml`

You need to follow the structure of the configuration file – place `<appender>` elements after all existing appenders, `<category>` elements after existing categories etc.

5. If you do not have SSL keys and certificate, generate them using the `keytool` from the JDK distribution as follows:

- \* Change the directory to the bin subdirectory of `JBOSS_HOME` and enter the following command:

  - `keytool -keystore JBOSS_HOME/server/[jboss_configuration]`

  - `/conf/server.keystore -genkey -alias jboss -keyalg RSA`

  - `-storepass changeit`

- \* Change your directory to the bin subdirectory of `/opt/hp/registry`.

- \* Export the certificate to a file using the following command:

  - `keytool -keystore JBOSS_HOME/server/[jboss_configuration]`

  - `/conf/server.keystore -export -file jboss.crt`

  - `-alias jboss -storepass changeit`

- \* Import the certificate to `clientconf.xml` in the HP SOA Registry distribution using this command:

  - `PStoreTool.sh (bat) add -certFile jboss.crt -alias jboss -config`

/opt/hp/registry/conf/clientconf.xml

6. Enable SSL in JBoss.

\* Uncomment following lines in the file  
JBOSS\_HOME/server/[jboss\_configuration]/deploy/jboss-web.deployer/server.xml

```
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
    maxThreads="150"
    address="{jboss.bind.address}"
    emptySessionPath="true"
    scheme="https" secure="true" clientAuth="false"
    keystoreFile="{jboss.server.home.dir}/conf/server.keystore"
    keystorePass="changeit"
    sslProtocol="TLS" />
```

Change the values of keystoreFile to {jboss.server.home.dir}/conf/server.keystore a keystorePass to changeit (use the actual values, which you have used when invoking keytool utility, if they differed from the examples shown here).

7. (Re)start the JBoss server

The HP SOA Registry Console URL is

<http://localhost.localdomain:8080/registry/uddi/web>

### **Further setup of HP SOA Registry Foundation**

---

If you have generated a test certificate, you may need to add your certificate to your cacerts file after following the instructions for setting up 4.3.0.CP02 SOA-P with HP SOA Registry.

```
keytool -alias jboss -import -file jboss.crt -keypass changeit -keystore <path-to-cacerts>/cacerts
```

Please replace all instances of uddiclient\_core.jar within the HP SOA Registry Foundation *registry.war* with the patched version of uddiclient\_core.jar. The uddiclient\_core.jar provided patches a limitation in the number of characters allowed in the ACCESS\_POINT URL, which is necessary to store EPRs.

Start up SOA-P 4.3.0.CP02 server (execute bin/run.sh) and then browse to:

<http://localhost:8080/registry/uddi/web>

## Uploading the Taxonomy

---

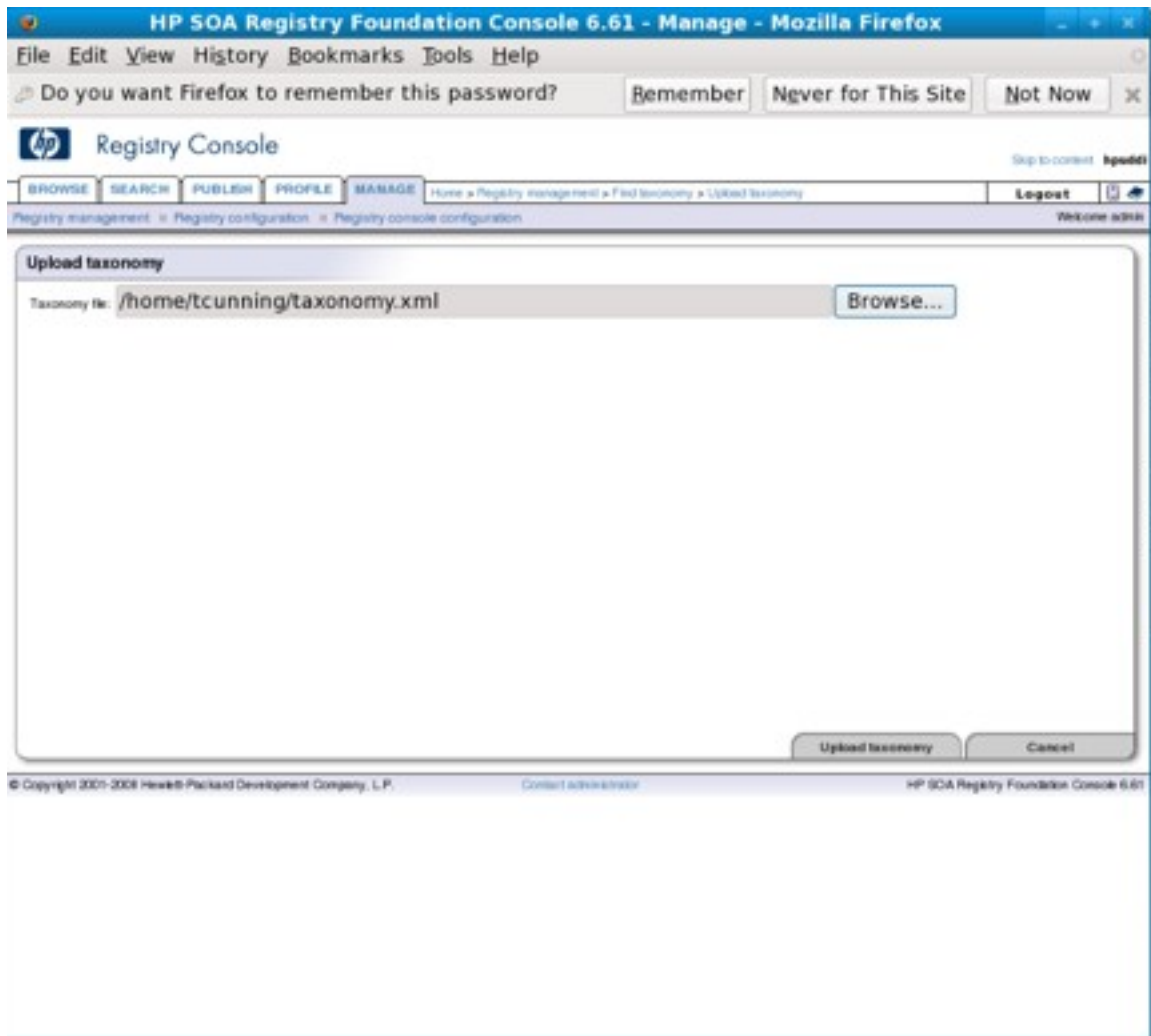
On the next page there is a XML snippet that you need to save to a file (taxonomy.xml). This file will create a non-enterprise taxonomy for JBossESB as well as an associated TModel.

Use the taxonomy.xml included in the transport package (SOA-P\_HOME/jboss-as/tools/systinet/taxonomy.xml), or paste the following into a taxonomy.xml.

```
<?xml version="1.0" encoding="UTF-8"?>
<taxonomy check="true" brief="false" xmlns="http://systinet.com/uddi/taxonomy/v3/5.5">
  <tModel tModelKey="uddi:org.jboss-org.jbossesb:category" deleted="false" tModelKeyV2="uuid:2a37a8d0-61b1-11de-99ab-d9552d7099aa"
  xmlns="http://systinet.com/uddi/taxonomy/v3/5.0">
    <name xmlns="urn:uddi-org:api_v3">org.jboss.soa.esb.:category</name>
    <description xmlns="urn:uddi-org:api_v3">org.jboss.soa.esb.:category</description>
    <overviewDoc xmlns="urn:uddi-org:api_v3">
      <overviewURL>http://uddi.org/pubs/uddi_v3.htm#GenKW</overviewURL>
    </overviewDoc>
    <categoryBag xmlns="urn:uddi-org:api_v3">
      <keyedReference tModelKey="uddi:uddi.org:categorization:types" keyName="Categorization system" keyValue="categorization"/>
      <keyedReference tModelKey="uddi:systinet.com:taxonomy:compatibility" keyName="Compatibility" keyValue="tModel"/>
      <keyedReference tModelKey="uddi:systinet.com:taxonomy:compatibility" keyName="Compatibility" keyValue="businessEntity"/>
      <keyedReference tModelKey="uddi:systinet.com:taxonomy:compatibility" keyName="Compatibility" keyValue="businessService"/>
      <keyedReference tModelKey="uddi:systinet.com:taxonomy:compatibility" keyName="Compatibility" keyValue="bindingTemplate"/>
      <keyedReference tModelKey="uddi:uddi.org:categorization:types" keyName="Checked value set" keyValue="checked"/>
      <keyedReference tModelKey="uddi:uddi.org:categorization:validatedby" keyName="" keyValue="uddi:86aca040-c380-11d8-b05c-7c163510b05c"/>
    </categoryBag>
  </tModel>
  <compatibilityBag xmlns="http://systinet.com/uddi/taxonomy/v3/5.0">
    <compatibility>tModel</compatibility>
    <compatibility>businessEntity</compatibility>
    <compatibility>businessService</compatibility>
    <compatibility>bindingTemplate</compatibility>
  </compatibilityBag>
  <categorizationBag xmlns="http://systinet.com/uddi/taxonomy/v3/5.0">
    <categorization>categorization</categorization>
  </categorizationBag>
  <validation xmlns="http://systinet.com/uddi/taxonomy/v3/5.0">
    <bindingTemplate bindingKey="uddi:86ae9c10-c380-11d8-b05c-7c163510b05c" serviceKey="uddi:86abb5e0-c380-11d8-b05c-7c163510b05c"
    xmlns="urn:uddi-org:api_v3">
      <accessPoint useType="endPoint">class:com.systinet.uddi.publishing.v1.validation.service.GeneralKeywordsValidator</accessPoint>
      <tModelInstanceDetails>
        <tModelInstanceInfo tModelKey="uddi:systinet.com:v1_validateValues"/>
        <tModelInstanceInfo tModelKey="uddi:uddi.org:categorization:general_keywords"/>
      </tModelInstanceDetails>
    </bindingTemplate>
    <bindingTemplate bindingKey="uddi:86adffd0-c380-11d8-b05c-7c163510b05c" serviceKey="uddi:86abb5e0-c380-11d8-b05c-7c163510b05c"
    xmlns="urn:uddi-org:api_v3">
      <accessPoint useType="endPoint">class:com.systinet.uddi.publishing.v2.validation.service.GeneralKeywordsValidator</accessPoint>
      <tModelInstanceDetails>
        <tModelInstanceInfo tModelKey="uddi:systinet.com:v2_validateValues"/>
        <tModelInstanceInfo tModelKey="uddi:uddi.org:categorization:general_keywords"/>
      </tModelInstanceDetails>
    </bindingTemplate>
    <bindingTemplate bindingKey="uddi:86aca040-c380-11d8-b05c-7c163510b05c" serviceKey="uddi:86abb5e0-c380-11d8-b05c-7c163510b05c"
    xmlns="urn:uddi-org:api_v3">
      <accessPoint useType="endPoint">class:com.systinet.uddi.publishing.v3.validation.service.GeneralKeywordsValidator</accessPoint>
      <tModelInstanceDetails>
        <tModelInstanceInfo tModelKey="uddi:uddi.org:v3_valueSetValidation"/>
        <tModelInstanceInfo tModelKey="uddi:uddi.org:categorization:general_keywords"/>
      </tModelInstanceDetails>
    </bindingTemplate>
  </validation>
</taxonomy>
```

“Publish” and login. Click “Manage”, and then click on “Taxonomy Management”.

Upload taxonomy.xml as a new taxonomy :



## Set up of JBoss ESB

---

Your server for HP Registry Foundation is now setup, and you will now need to set up the 4.3.0.CP02 SOA-P Server so that it uses HP Registry Foundation as the registry for the ESB.

1. Uncompress the registry.war file you copied into *server/[jboss\_configuration]/deploy* into a directory "registry.war".

```
cd server/[jboss_configuration]/deploy
mkdir registry.tmp
cd registry.tmp
jar xvf ../registry.war
cd ..
rm registry.war
mv registry.tmp registry.war
```

2. Move xercesImpl.jar out of *server/[jboss\_configuration]/deploy/registry.war*
3. Copy jbossesb-transport-1.0.jar to *server/[jboss\_configuration]/deploy/registry.war/WEB-INF/lib*
4. Copy jbossesb-transport-client-1.0.jar to *server/[jboss\_configuration]/deploy/jbossesb.sar/lib*
5. Add the following servlet definition to *server/[jboss\_configuration]/deploy/registry.war/WEB-INF/web.xml*:

```
<servlet>
    <servlet-name>RegisterServicesWithJNDI</servlet-name>
    <servlet-class>org.jboss.soa.esb.registry.rmi.RegistrationService</servlet-class>
    <load-on-startup>1</load-on-startup>
</servlet>
```

## Configuring the ESB for a co-located registry

---

Edit server/<configuration>/deploy/jbossesb.sar/jbossesb-properties.xml and change the registry section to the following:

```
<properties name="registry">
  <property name="org.jboss.soa.esb.registry.queryManagerURI"
    value="jnp://localhost:1099/InquiryService?org.jboss.soa.esb.registry.rmi.Inquiry#inquire"/>
  <property name="org.jboss.soa.esb.registry.lifeCycleManagerURI"
    value="jnp://localhost:1099/PublishService?org.jboss.soa.esb.registry.rmi.Publish#publish"/>
  <property name="org.jboss.soa.esb.registry.implementationClass"
    value="org.jboss.internal.soa.esb.services.registry.JAXRRegistryImpl"/>
  <property name="org.jboss.soa.esb.registry.factoryClass"
    value="org.apache.ws.scout.registry.ConnectionFactoryImpl"/>
  <property name="org.jboss.soa.esb.registry.user" value="admin"/>
  <property name="org.jboss.soa.esb.registry.password" value="changeit"/>
  <!-- the following parameter is scout specific to set the type of communication between scout
    and the UDDI (embedded, rmi, soap) -->
  <property name="org.jboss.soa.esb.scout.proxy.transportClass"
    value="org.apache.ws.scout.transport.RMITransport"/>

  <!-- specify the interceptors, in order -->
  <property name="org.jboss.soa.esb.registry.interceptors"
    value="org.jboss.internal.soa.esb.services.registry.InVMRegistryInterceptor"/>
    <!-- Example of specifying multiple registry interceptors
  <property name="org.jboss.soa.esb.registry.interceptors"
    value="org.jboss.internal.soa.esb.services.registry.InVMRegistryInterceptor,
    org.jboss.internal.soa.esb.services.registry.CachingRegistryInterceptor"/>
    -->

  <!-- The following properties modify the cache interceptor behaviour -->
  <property name="org.jboss.soa.esb.registry.cache.maxSize" value="100"/>
  <property name="org.jboss.soa.esb.registry.cache.validityPeriod" value="600000"/>
</properties>
```



## Configuring the ESB for a remote registry

### Add Axis and commons-discovery JARs

---

Copy a recent version of Apache Axis and Apache Commons Discovery into server/[jboss\_configuration]/lib. This integration was tested using axis-1.4.jar and commons-discovery-0.2.jar, so we recommend those JAR versions for use. Download axis-bin-1\_4.tar.gz from the Apache Axis site (<http://ws.apache.org/axis/java/releases.html>) to find these JARs.

### Configuring the ESB for a remote registry

---

Edit server/<configuration>/deploy/jbossesb.sar/jbossesb-properties.xml and change the registry section to the following:

```
<properties name="registry">
  <property name="org.jboss.soa.esb.registry.queryManagerURI"
    value="http://testmachine.redhat.com:8080/registry/uddi/inquiry"/>
  <property name="org.jboss.soa.esb.registry.lifeCycleManagerURI"
    value="http://testmachine.redhat.com:8080/registry/uddi/publishing"/>
  <property name="org.jboss.soa.esb.registry.implementationClass"
    value="org.jboss.internal.soa.esb.services.registry.JAXRRegistryImpl"/>
  <property name="org.jboss.soa.esb.registry.factoryClass"
    value="org.apache.ws.scout.registry.ConnectionFactoryImpl"/>
  <property name="org.jboss.soa.esb.registry.user" value="admin"/>
  <property name="org.jboss.soa.esb.registry.password" value="changeit"/>
  <!-- the following parameter is scout specific to set the type of communication between scout
and the UDDI (embedded, rmi, soap) -->
  <property name="org.jboss.soa.esb.scout.proxy.transportClass"
    value="org.apache.ws.scout.transport.AxisTransport"/>
</properties>
```

## Running

---

Start JBossESB :

```
cd bin  
run.bat (if on Windows) or run.sh (on Unix)
```

Read [product/docs/GettingStarted.pdf](#) and follow the directions for running the Helloworld QS.