

JBoss ESB 4.2 Milestone Release 1

Getting Started With JBoss ESB

JBESB-GS-3/23/07



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

JBoss ESB 4.2 Milestone Release 1

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 2007 JBoss Inc.

Contents

Contents.....iv	Additional Documentation.....7
About This Guide.....5	Contacting Us.....7
What This Guide Contains.....5	Getting Started.....8
	Four Simple Steps to get up and running.....9
Audience.....5	The Hello World QuickStart.....9
	Components of the QuickStart.....11
Prerequisites.....5	ESB Aware and Unaware Messages.....12
	QuickStart Sequence of Events.....13
Organization.....5	TrailBlazer.....14
Documentation Conventions.....6	Overview.....14



About This Guide

What This Guide Contains

The goal of this document is assist you in getting up and running with test applications on JBossESB as quickly as possible.

Audience

This guide is anyone who is responsible for using JBoss ESB 4.2 Milestone Release 1 installations and wants to know how to install and use it.

Prerequisites

None.

Organization

This guide contains the following chapter:

1. **Chapter 1, Installation:** This chapter reviews prerequisites (software needed to operate JBossESB), downloading JBossESB, and building JbossESB.
2. **Chapter 2, Trailblazer:** A quick summary of the trailblazer example.

Documentation Conventions

The following conventions are used in this guide:

Convention	Description
<i>Italic</i>	In paragraph text, italic identifies the titles of documents that are being referenced. When used in conjunction with the Code text described below, italics identify a variable that should be replaced by the user with an actual value.
Bold	Emphasizes items of particular importance.
Code	Text that represents programming code.
Function Function	A path to a function or dialog box within an interface. For example, "Select File Open." indicates that you should select the Open function from the File menu.
() and	<p>Parentheses enclose optional items in command syntax. The vertical bar separates syntax items in a list of choices. For example, any of the following three items can be entered in this syntax:</p> <pre>persistPolicy (Never OnTimer OnUpdate NoMoreOftenThan)</pre>
Note:	A note highlights important supplemental information.
Caution:	A caution highlights procedures or information that is necessary to avoid damage to equipment, damage to software, loss of data, or invalid test results.

Table 1 Formatting Conventions

Additional Documentation

In addition to this guide, the following guides are available in the JBoss ESB 4.2 Milestone Release 1 documentation set:

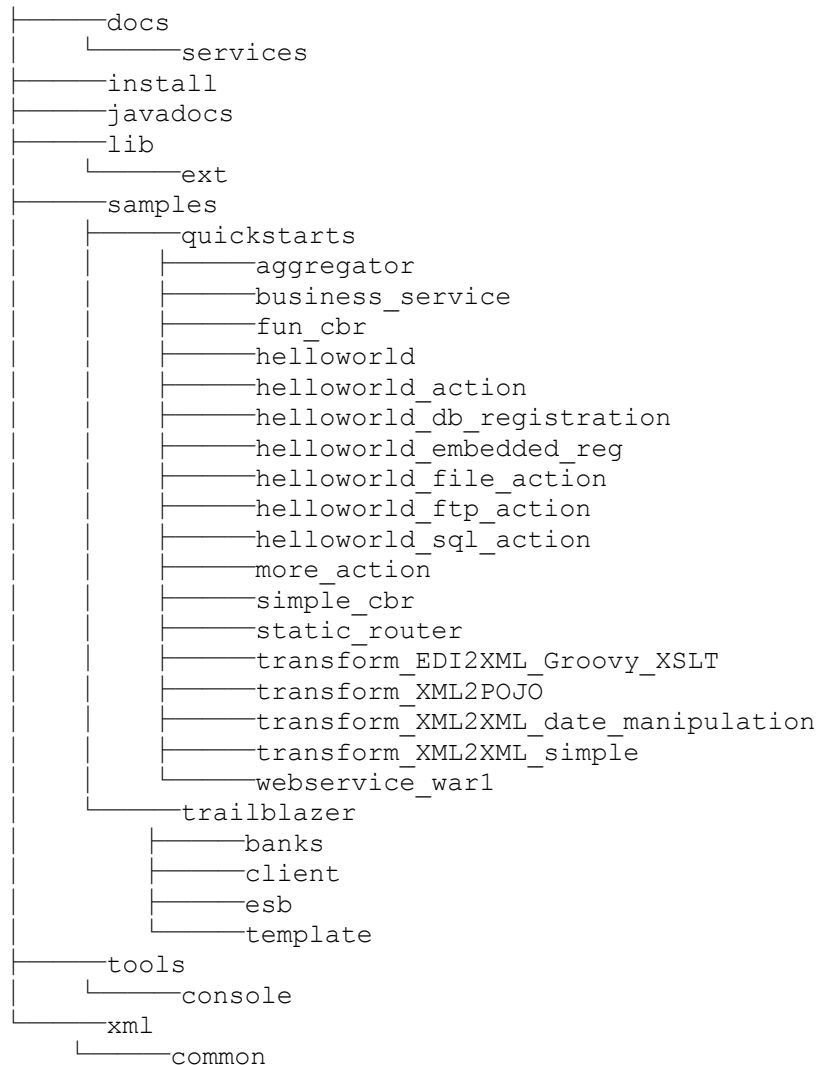
1. **JBoss ESB 4.2 Milestone Release 1 *Trailblazer Guide***: Provides guidance for using the trailblazer example.
2. **JBoss ESB 4.2 Milestone Release 1 *Administration Guide***: How to manage the ESB.
3. **JBoss ESB 4.2 Milestone Release 1 *Programmers Guide***: How to use JBossESB.
4. **JBoss ESB 4.2 Milestone Release 1 *Release Notes***: Information on the differences between this release and previous releases.
5. **JBoss ESB 4.2 Milestone Release 1 *Services Guides***: Various documents related to the services available with the ESB.

Contacting Us

Questions or comments about JBoss ESB 4.2 Milestone Release 1 should be directed to our support team.

Getting Started

After downloading and expanding the JBoss ESB 4.0 GA distribution, you should have the following file structure.



The quickest way to get started using JBoss ESB is through running one of the quickstarts in the *samples/quickstarts* folder. This will also perform a basic validation of your system. Before doing this however, be sure to check that your system meets the following minimum requirements:

1. JDK 5 (v1.5.0_06 recommended)

2. Ant (v1.6.5 recommended)
3. JBoss Application Server v4.0.5GA [with ejb3]

There are three ways to run the ESB. You can deploy it to JBossAS, to Tomcat, or run it in standalone mode. In any case you have to locate the 'deployment.properties' in the install directory, and edit the properties set in this file. This document assumes you have ant (1.6.5 or higher) and java5 installed on your machine, and that you have a fresh copy of JBoss AS. So now go and download the JBossESB 4.0GA distribution from <http://labs.jboss.com/portal/jbossesb/downloads>

Standalone

1. The jbossesb-server can be downloaded as a separate download and is fully configured.
2. You can deploy custom code in '.esb' archives.
3. Start the esb-server

Deploy to JBossAS

1. Edit your version of the deployment.properties. Open this file and edit the following lines if needed:


```
# application server root directory
org.jboss.esb.server.home=/jboss-4.0.5.GA
# the instance of jboss you are running (default)
org.jboss.esb.server.config=default
```
2. Run 'ant'. This will deploy the esb to your JBossAS instance. It copies the jbossesb.sar into your deploy directory.
3. You can deploy custom code (actions) by deploying the '.esb' archives into the deploy directory.
4. Start your appserver.

Deploy to Tomcat

1. Edit your version of the deployment.properties. Open this file and edit the following line if needed


```
#####
# Tomcat
#####
# jbossesb tomcat home directory
org.jboss.esb.tomcat.home=/apache-tomcat-5.5.20
```
2. Run 'ant tomcat'. This will deploy the esb to your tomcat instance. It creates a jbossesb.war deploy directory in your tomcat.home/webapps directory.
3. Follow the steps in the tomcat/README.txt if you do not yet have a juddi database. Note that the JBossAS deploy comes with a preconfigured hsqldb.
4. You can deploy custom code (actions) by deploying the jar to the tomcat.home/webapps/jbossesb/WEB-INF/lib directory
5. Start tomcat and see:

```
12:12:20,875 INFO [ParamFileRepository] Setting parameter repository root dir t
o [C:\apache-tomcat-5.5.20\bin\.]
```

```

12:12:21,546 INFO [Generator] Serializing ESB Aware Listener 'ConfigTree' confi
guration to [C:\apache-tomcat-5.5.20\conf\jbossesb-listener.xml].
12:12:21,578 INFO [Generator] Serializing ESB Gateway 'ConfigTree' configuratio
n to [C:\apache-tomcat-5.5.20\conf\jbossesb-gateway.xml].
12:12:21,578 INFO [ConfigurationController] Parameter reload completed.
12:12:21,593 INFO [ConfigurationController] Configuration Controller instance s
tarted.
Jan 30, 2007 12:12:22 PM org.apache.coyote.http11.Http11BaseProtocol start
INFO: Starting Coyote HTTP/1.1 on http-8080
Jan 30, 2007 12:12:22 PM org.apache.jk.common.ChannelSocket init
INFO: JK: ajp13 listening on /0.0.0.0:8009
Jan 30, 2007 12:12:22 PM org.apache.jk.server.JkMain start
INFO: Jk running ID=0 time=0/31 config=null
Jan 30, 2007 12:12:22 PM org.apache.catalina.storeconfig.StoreLoader load
INFO: Find registry server-registry.xml at classpath resource
Jan 30, 2007 12:12:22 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 2813 ms

```

ESB archives

An ESB archive is a zip file with a .esb extension, and has the following structure:

```

|-----META-INF
|       |-----jboss-esb.xml
|       |-----MANIFEST.MF
|-----<java classes>
|-----<queue-service.xml>

```

The custom action classes can be put in the root of the archive. Optionally you can provide a queue-service.xml if to bring up JMS queues or topics that are specific for this ESB package.

Finally, the configuration of the package goes in the META-INF/jboss-esb.xml.

Note that you cannot (yet) deploy esb archives to the Tomcat deployment.

The Hello World QuickStart

This QuickStart allows you get up and running with JBoss ESB, out of the box. It is located in the distribution under *samples/quickstarts/helloworld*.

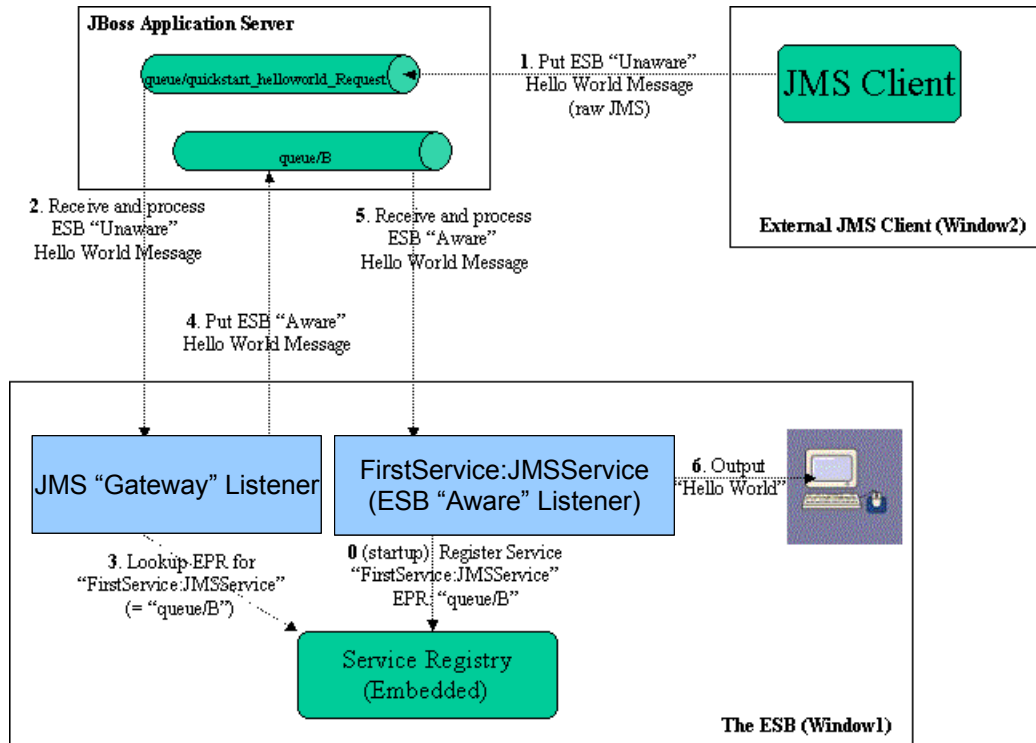
To run this QuickStart:

1. Copy *samples/quickstarts/esb-quickstart-service.xml* to your JBoss Application Server (*/jboss-4.0.5.GA/server/default/deploy*) or rely on the queue-service.xml definitions if you opt to deploy the esb package. This will bring up all the JMS Queues you need to run the quickstarts. Note that the ESB can interact with any JMS provider, but ships with JBossMessaging. So our examples use JBossMessaging, which is the default provider in JBossesb-server-4.0.1. Note that we have also tested JBossMQ, MQSeries and ActiveMQ.
2. Update the “jboss.home.dir” property setting in *samples/quickstarts/quickstarts.properties* to reference location of your Server.

3. Start your Server.
4. From a command terminal window (“Window1”), change directory into the *samples/quickstarts/helloworld* directory.
5. In Window1, type “ant” to run the ESB as a JAVA application, or run “ant deploy” to deploy the .esb archive.
6. This will start the ESB Message “Listeners”. Wait for “**Listener Ready**” to appear on the terminal window. The ESB is now running!
7. Open a 2nd command terminal window (“Window2”) and change director to *samples/quickstarts/helloworld* again.
8. In Window2, type “ant runtest”.
9. Switch back to Window1. You should soon see a “Hello World” message appear in the terminal window.
10. That's it! The QuickStart ran successfully. Your environment is properly configured for JBoss ESB.

Components of the QuickStart

The following diagram illustrates the sequence of events that take place in this QuickStart. It touches on a number of the key concepts within JBoss ESB¹.



Window1 shows each of the main "ESB" components used in this sample:

- 1. Service Registry:** This is a JAXR Registry implementation. In this QuickStart, the registry is using an embedded Datastore as the Service repository (Hypersonic). See *docs/services/RegistryConfiguration.pdf*. For more details on the Registry Service.
- 2. JMS Gateway Listener:** A "Gateway Listener" is one of the key architectural components within JBoss ESB. This listener type is, as its name would suggest, the gateway to the ESB from endpoints outside the domain of the ESB. In this case, we're using a JMS Gateway.
- 3. The ESB Aware Service Listener:** The "FirstService:SimpleJMSService" ESB Aware Service Listener listens for "ESB Aware" messages on "queue/B". This introduces you further to the concept of ESB "Aware" and "Unaware" messages. We will touch on these next.

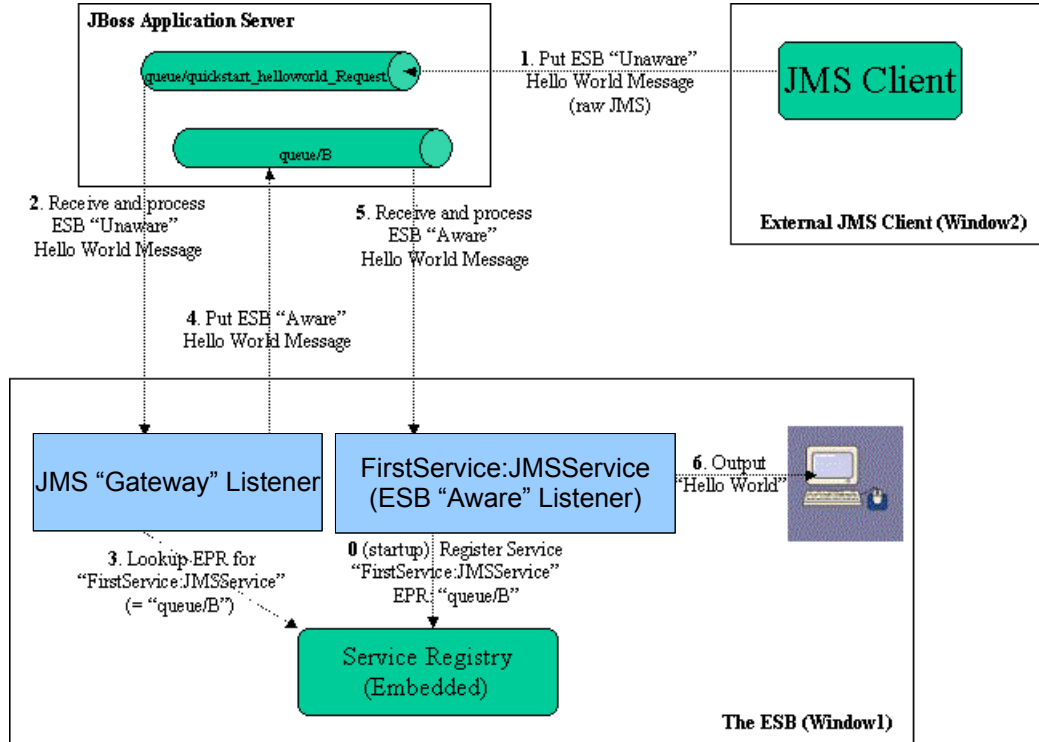
¹ Use the "Zoom" features of your viewer to see the diagram in more detail.

ESB Aware and Unaware Messages

JBoss ESB has a well defined concept of what a message is. This is defined fully in *xml/message.xsd*. This construct makes it possible to pass decorated messages payloads between components of the ESB. The message payload is typically stored in the message “Body” (see the Programmers Guide).

This makes a lot of sense from the point of Services in within the ESB domain being able to collaborate effectively. However, it is not practical to expect endpoints outside the domain of a JBoss ESB deployment to be “aware” of these internal ESB constructs. For this reason, JBoss ESB has the concept of ESB Aware and Unaware Messages and Endpoints, with the Gateway acting as the bridge (adapter) between the two worlds.

QuickStart Sequence of Events



After starting the ESB in Window1 and before any "Hello World" messages are put on the bus, the "FirstService:SimpleJMSService" Service is registered with the Registry Service.

The sequence of events in the Hello World QuickStart are as follows:

1. ESB Unaware JMS Client endpoint puts an ESB Unaware "Hello World" Message (plain String Object) into JMS Queue `"queue/quickstart_helloworld_Request"`.
2. The JMS Gateway Listener receives the ESB Unaware message. The Gateways Job is to adapt this message by making it an ESB Aware Message for consumption by an ESB Aware Endpoint.
3. The JMS Gateway Listener uses the registry to lookup the Endpoint Reference (EPR) for "FirstService:SimpleJMSService" Service. This works out to be JMS Queue `"queue/B"`.
4. The JMS Gateway Listener "adapts" the message into an ESB Aware message and places it into JMS Queue `"queue/B"`.
5. "FirstService:SimpleJMSService" Service receives the message.
6. "FirstService:SimpleJMSService" Service extracts the payload from the message and prints it to the console.

TrailBlazer

Overview

JBoss TrailBlazers and Demo Applications are designed to help you get up and running quickly with JBoss products and technologies. We encourage you to Run them, Download them, and enjoy the learning process!

The Loan Broker TrailBlazer example was developed to verify your JBossESB installation and also to exhibit some of the numerous capabilities of JBossESB. This example was based on information from www.eaipatterns.org, along with the example found at [JavaZone 2005](#).

For details of configuring and running the TrailBlazer, see the accompanying “Trailblazer” document.