

# **JBoss ESB 4.0 CR2**

---

## Getting Started With JBoss ESB

JBESB-GS-1/11/07





## 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 4.0 CR2

## 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

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



# 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 4.0 CR2 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.

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.
<b>Bold</b>	Emphasizes items of particular importance.
Code	Text that represents programming code.
<b>Function   Function</b>	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>
<b>Note:</b>	A note highlights important supplemental information.
<b>Caution:</b>	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 4.0 CR2 documentation set:

1. **JBoss ESB 4.0 CR2 Programmer's Guide:** Provides guidance for writing applications.
2. **JBoss ESB 4.0 CR2 TrailBlazer Guide:** Provides a guide to understanding, configuring, deploying and running the LoanBroker TrailBlazer.
3. **JBoss ESB 4.0 CR2 Configuring Hypersonic Guide:** This is necessary for setting up the Hypersonic database if you want to use it within the trailblazer.

## Contacting Us

---

Questions or comments about JBoss ESB 4.0 CR2 should be directed to our support team.

# Getting Started

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

```
|-- docs
|   |-- config
|   |-- services
|-- install
|   |-- conf
|   |-- data
|   |-- deploy
|   |-- jUDDI-registry
|   |-- message-store
|-- javadocs
|-- lib
|-- samples
|   |-- docs
|   |-- quickstarts
|       |-- helloworld
|       |-- helloworld_action
|       |-- helloworld_embedded_reg
|       |-- more_action
|       |-- simple_cbr
|       |-- simple_transformation
|   |-- trailblazer
|       |-- banks
|       |-- client
|       |-- docs
|       |-- esb
|       |-- loanbroker
|       |-- template
|-- tools
|   |-- console
|-- xml
```

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 v4.0.4

### **Four Simple Steps to get up and running.**

1. 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 4.0.x. So now go and download the JBossESB 4.0RC1 distribution from <http://labs.jboss.com/portal/jbossesb/downloads>
2. Go into the 'install' directory to find the deployment.properties-example. Open this file and edit the following lines:

```
# application server root directory
org.jboss.esb.appserver.home=C:/cygwin/home/kstam/apps/jboss-4.0.4.GA
# the instance of jboss you are running (default)
org.jboss.esb.appserver.config=default
#the place where the jboss libraries can be found
# Note: You should not need to change this entry unless you altered
# the jbossesb distribution directory structure
org.jboss.esb.dist.lib=../lib
```

3. Run 'ant configure'. This will deploy the esb to your jboss AS instance. It copies the jbossesb.sar directory into your deploy directory.
4. Start your appserver and see the JBossESB heartbeat:

```
11:37:35,546 INFO [Server] JBoss (MX MicroKernel) [4.0.4.GA (build:
CVSTag=JBoss_4_0_4_GA date=200605151000)] Started in 23s:843ms
11:37:50,375 INFO [EsbListenerController] Reloading parameters

-----
11:38:10,390 INFO [EsbListenerController] Reloading parameters

-----
11:38:20,421 INFO [GatewayListenerController] Reloading parameters

-----
11:38:20,453 WARN [GatewayListenerController] No value specified for:
endTime - Will run forever (or until interrupted by user)
```

That's it. You've done it.

Note that in your server/default/conf directory you now have 2 new configuration files:

```
jbossesb-listener.xml
jbossesb-gateway.xml
```

You can add xml fragments in there and when the controller reload your listeners and gateways come up !!! So try pasting in xml from the Hello World config sample. When the listeners reload their parameters it will get picked up dynamically. Note that gateways reference listeners, so add the listeners first.

Finally note that the jbossesb.sar is pretty basic and if you want to use smooks, or jboss-rules for content-based-routing you will have to add them. More detailed instructions can be found elsewhere.

### **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\_embedded\_reg<sup>1</sup>*.

To run this QuickStart:

---

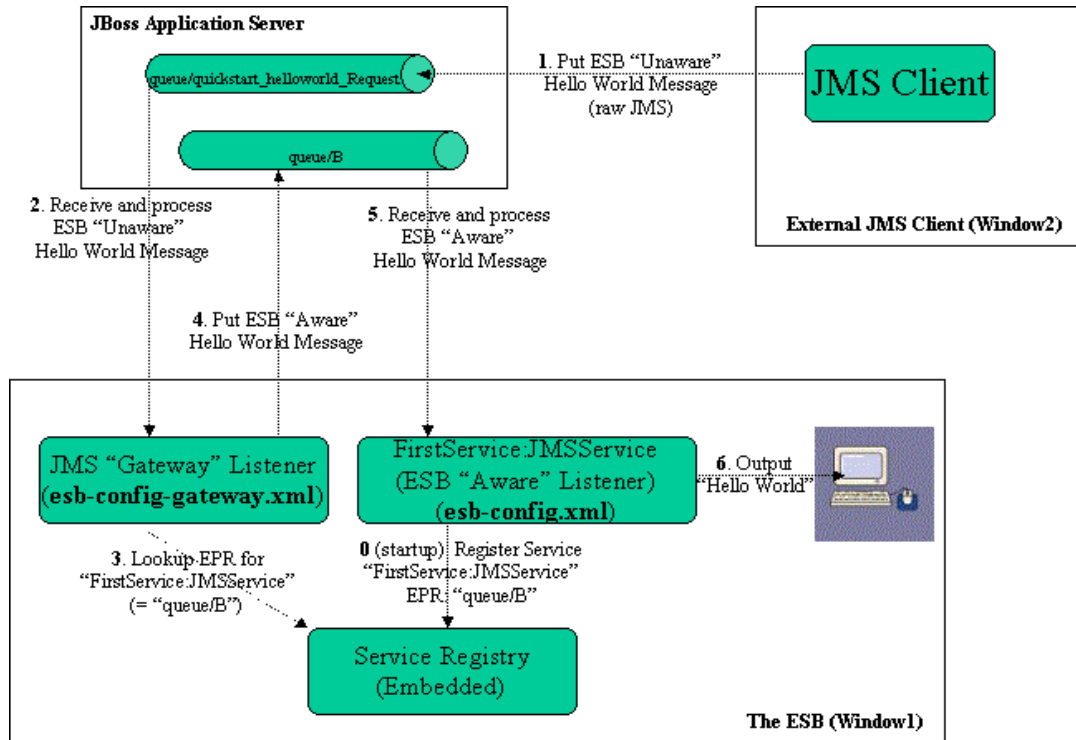
<sup>1</sup> All other Quickstarts require a local checkout of the ESB codebase from <http://anonsvn.labs.jboss.com/labs/jbossesb/trunk/> (Subversion Repository). This dependency will be removed for the GA release.



1. Copy *samples/quickstarts/esb-quickstart-service.xml* to your JBoss Application Server (*jboss-4.0.4.GA/server/default/deploy*).
2. Update the “jboss.home.dir” property setting in *samples/quickstarts/quickstarts.properties* to reference location of your JBoss Application Server.
3. Start you JBoss Application Server.
4. From a command terminal window (“Window1”), change directory into the *samples/quickstarts/helloworld\_embedded\_reg* directory.
5. In Window1, type “ant”.
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 2<sup>nd</sup> command terminal window (“Window2”) and change director to *samples/quickstarts/helloworld\_embedded\_reg* 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 ESB<sup>1</sup>.



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. See the *esb-config-gateway.xml* configuration file in the root of the QuickStart.
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.

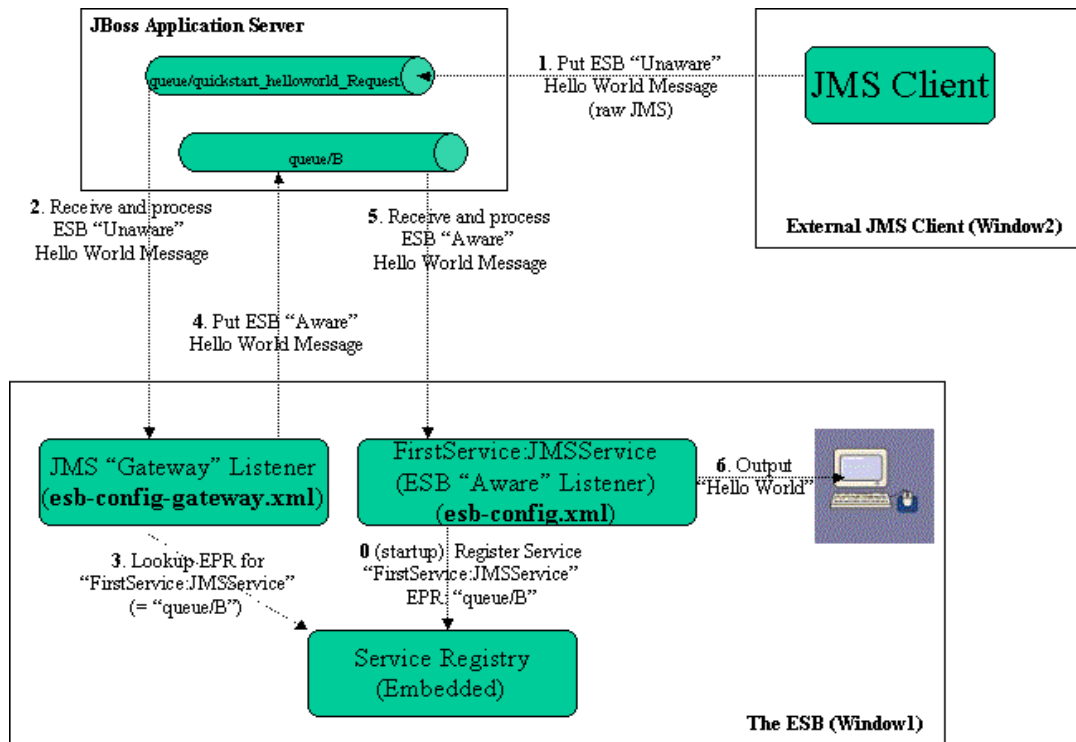
<sup>1</sup> Use the "Zoom" features of your viewer to see the diagram in more detail.

## **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](http://www.eaipatterns.org), along with the example found at [JavaZone 2005](#).

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