

# **JBoss ESB 4.2 Milestone Release 3**

Getting Started With JBoss ESB

JBESB-GS-7/19/07





## **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 3**

## **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</b> .....iv	<b>Additional Documentation</b> .....7
<b>About This Guide</b> ..... 5	<b>Contacting Us</b> .....7
<b>What This Guide Contains</b> .....5	<b>Getting Started</b> .....8
<b>Audience</b> .....5	Four Simple Steps to get up and running. ....9
<b>Prerequisites</b> .....5	<b>The Hello World QuickStart</b> .....9
<b>Organization</b> .....5	Components of the QuickStart.....11
<b>Documentation Conventions</b> .....6	ESB Aware and Unaware Messages.....12
	QuickStart Sequence of Events.....13
	<b>TrailBlazer</b> .....14
	<b>Overview</b> .....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 3 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.
<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 4.2 Milestone Release 3 documentation set:

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

### Contacting Us

---

Questions or comments about JBoss ESB 4.2 Milestone Release 3 should be directed to our support team.



# Getting Started

The quickest way to get started using JBoss ESB is by 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 4.2.0.GA or JBoss ESB Server 4.2.0.MR3

There are three ways to run the ESB. You can deploy it to JBossAS/ JBoss ESB Server, run standalone, or run it to Tomcat. The first step is 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 4.2.0.GA distribution from <http://labs.jboss.com/portal/jbossesb/downloads>

The Jboss ESB Server is provided as a convenient and lightweight container to deploy to. It contains the majority of the components of the JBoss application server, but does not contain EJB3. The main advantage of using the ESB Server over using the JBoss 4.2.0.GA application server is a much quicker boot time, which is helpful in development.

## **Deploy JBoss ESB to JbossAS 4.2.0.GA or JBossESB 4.2.0.MR3 Server**

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

```
# application server root directory
org.jboss.esb.server.home=/jboss-4.2.0.GA
# the instance of jboss you are running(default)
org.jboss.esb.server.config=default
```

2. Run 'ant' (default target). This will deploy the jbossesb.sar and a number of .esb package archives (jbossesb.esb, jbpmp.esb, jbrules.esb, smooks.esb, spring.esb, soap.esb) to your application server.
3. You can deploy custom code (actions) by deploying the '.esb' archives into the deploy directory.
4. Start your appserver.

## ESB archive deployment strategies

The ESB is packaged with base services that ship with the ESB. A service consist of action code + configuration. Both of which should be deployed in an .esb archive. You can deploy as many .esb archives as you please. You can specify the deployment order of the archives using the deployment.xml. Typically you would deploy the .esb archive to the 'deploy' directory. The idea behind the .esb archive is that it is a deployable service unit. It should enable you to move a service between servers simply by moving the .esb archive. An ESB archive is a zip file with a .esb extension, and has the following structure:

```
├── META-INF
│   ├── jboss-esb.xml
│   ├── <deployment.xml>
│   └── MANIFEST.MF
├── <java classes>
├── <jars>
└── <queue-service.xml>
```

where,

- the jboss-esb.xml contains the service configuration (listener and actions), as well as provider configuration.
- the deployment.xml is optional, but can be used for 2 reasons:
  - make this .esb archive depend on other archives, to specify classloading order.
  - make the deployment of this .esb archive scoped.
- you custom action classes.
- additional jar archives your actions depend on.
- queue-service.xml, if your 'provider' section references queues or topics you can deploy their configuration in the .esb archive. Note that any other way to deploy these queues is fine too, we just recommend this to keep your deployments as self-contained as possible and therefore to keep dependency management simple.

[JBossESB](#) ships with a number of standard service archives:

- [jbossesb.esb](#) - internal services like the [DeadLetterService](#),
- [jbrules.esb](#) - needed for rules evaluation using [JBossRules](#), for services like the [Content-Based Router](#)
- [jbpm.esb](#) - needed for default provider of Business Process Management [jBPM](#), and
- [smooks.esb](#) - default message transformation engine [Smooks](#).

These services are deployed by default, but you should be able to remove them if you don't need these service deployments.

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*.

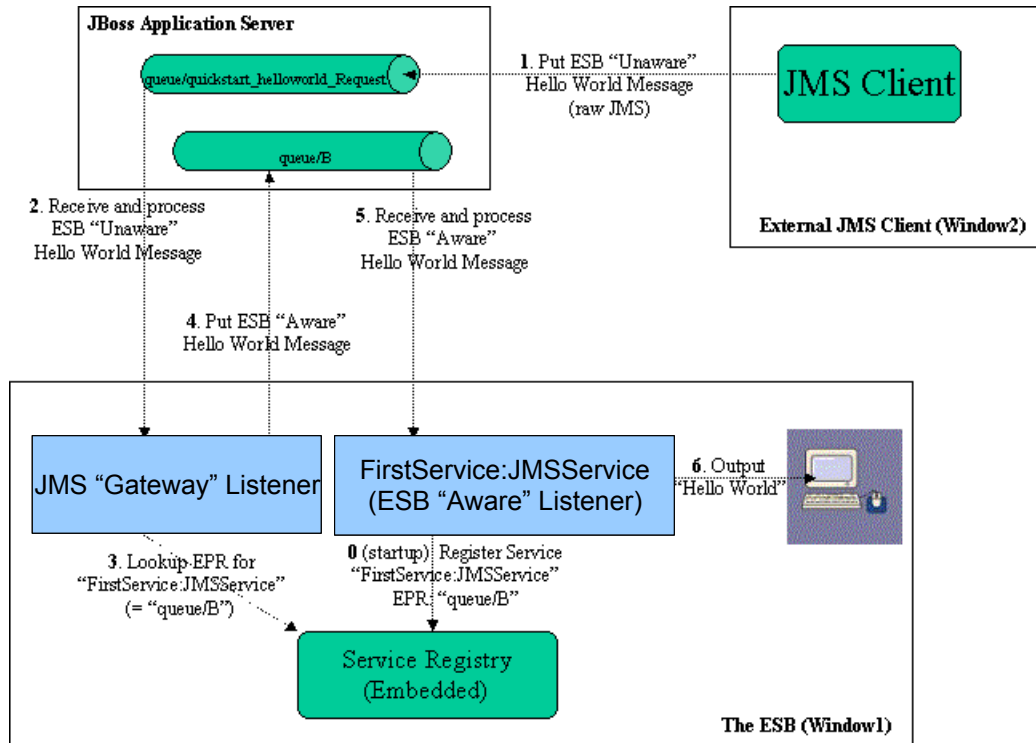
To run this QuickStart:

1. Start your Server in a way that allows you to view the output console. If you are on Windows, it is preferable not to run JBoss as a Windows Service for the purposes of this guide.
2. From a command terminal window, change directory into the *samples/quickstarts/helloworld* directory.
3. Make sure that *install/deployment.properties* has the correct config and home directory settings for your server.
4. Type “ant deploy” to deploy the helloworld .esb package archive to your application server.
5. Type “ant runtest”.
6. Switch back to your application server's console. You should soon see a “Hello World” message appear in your application server's console.
7. That's it! The QuickStart ran successfully. Your environment is properly configured for JBoss ESB.

You can find more detailed directions on how to setup the quickstart examples by running “ant help-quickstarts” under any of the specific quickstart directories. To get information on how to run a particular quickstart under different deployment scenarios, change directory to the specific quickstart and type “ant help”.

## 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 uses RMI based communication. 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:SimpleListener" ESB Aware Service Listener listens for "ESB Aware" messages on `"queue/quickstart_helloworld_Request_gw"`. This introduces you further to the concept of ESB "Aware" and "Unaware" messages. We will touch on these next.

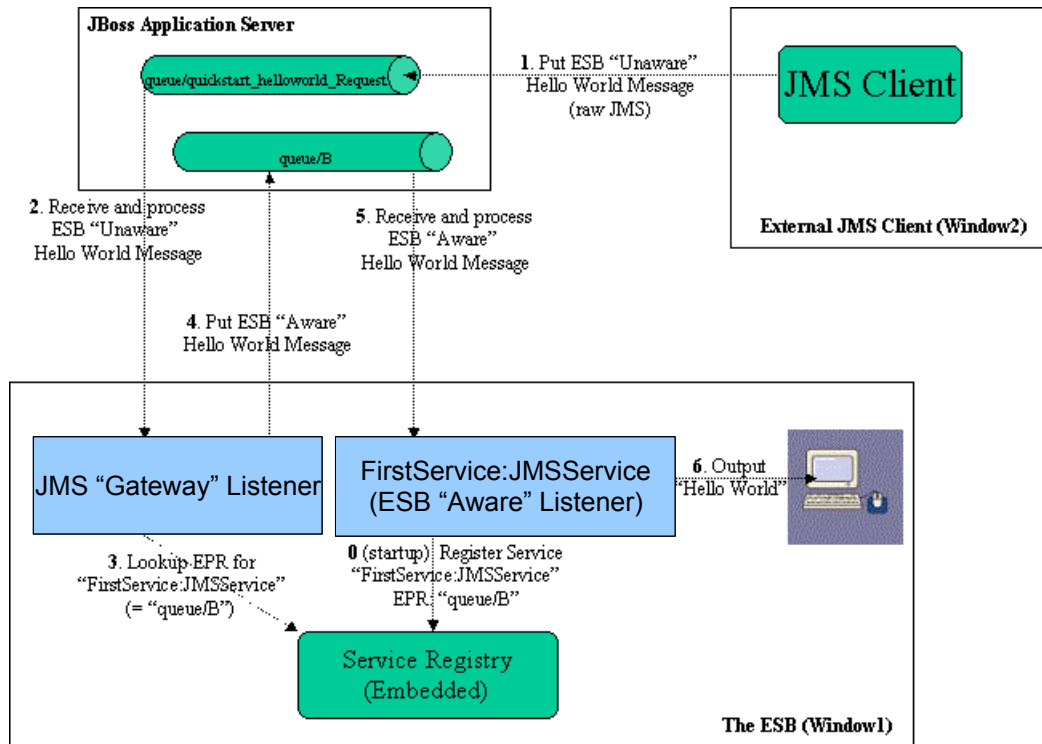
<sup>1</sup> Use the "Zoom" features of your viewer to see the diagram in more detail.

## **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



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/quickstart_helloworld_request_gw"`.
4. The JMS Gateway Listener "adapts" the message into an ESB Aware message and places it into JMS Queue `"queue/=quickstart_helloworld_request_gw"`.
5. `"FirstService:SimpleJMSService"` Service receives the message.
6. `"FirstService:SimpleJMSService"` Service extracts the payload from the message and prints it to the console.

## Running other Quickstarts

---

Once you have successfully run the Helloworld QuickStart and understand the concepts involved, there are many other Quickstarts to try. Please note that the Quickstarts have different requirements which are documented in their respective readme.txt, and that not all of the Quickstarts will run in every deployment. Below is a suggested map of Quickstarts to follow in order :

### Quickstarts Trailmap:

1. helloworld
2. helloworld\_action
3. more\_action
4. helloworld\_file\_action
5. helloworld\_ftp\_action
6. simple\_cbr
7. fun\_cbr
8. business\_service
9. business\_rules\_service
10. scripting\_groovy
11. transform\_CSV2XML
12. transform\_XML2POJO
13. transform\_XML2XML\_simple
14. transform\_XML2XML\_date\_manipulation
15. aggregator
16. bpm\_orchestration1
17. bpm\_orchestration2
18. webservice\_consumer1
19. webservice\_producer

# 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.