JBoss Enterprise SOA Platform 4.3

Getting Started Guide

For JBoss Administrators and Developers

JBoss Enterprise SOA Platform 4.3 Getting Started Guide
For JBoss Administrators and Developers
Edition 4.3.5

This manual contains content from the JBossESB Getting Started Guide by the JBoss ESB Development Team and Community. Further details about JBossESB can be found at the project's website *http://www.jboss.org/jbossesb*.

This document provides instructions for the installation, configuration and launching of the JBoss Enterprise SOA Platform product. Additionally it also includes an overview of the JBossESB quickstarts which provide an introduction to developing with the different features of the JBoss Enterprise SOA Platform.

Preface

1. Document Conventions

This manual uses several conventions to highlight certain words and phrases and draw attention to specific pieces of information.

In PDF and paper editions, this manual uses typefaces drawn from the *Liberation Fonts*[1] set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later includes the Liberation Fonts set by default.

1.1. Typographic Conventions

Four typographic conventions are used to call attention to specific words and phrases. These conventions, and the circumstances they apply to, are as follows.

`Mono-spaced Bold`

Used to highlight system input, including shell commands, file names and paths. Also used to highlight keycaps and key combinations. For example:

> To see the contents of the file **`my_next_bestselling_novel`** in your current working directory, enter the **`cat my_next_bestselling_novel`** command at the shell prompt and press **`Enter`** to execute the command.

The above includes a file name, a shell command and a keycap, all presented in mono-spaced bold and all distinguishable thanks to context.

Key combinations can be distinguished from keycaps by the hyphen connecting each part of a key combination. For example:

> Press **`Enter`** to execute the command.
>
> Press **`Ctrl`**+**`Alt`**+**`F2`** to switch to the first virtual terminal. Press **`Ctrl`**+**`Alt`**+**`F1`** to return to your X-Windows session.

The first paragraph highlights the particular keycap to press. The second highlights two key combinations (each a set of three keycaps with each set pressed simultaneously).

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in **`mono-spaced bold`**. For example:

> File-related classes include **`filesystem`** for file systems, **`file`** for files, and **`dir`** for directories. Each class has its own associated set of permissions.

Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialog box text; labeled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

> Choose **System** → **Preferences** → **Mouse** from the main menu bar to launch **Mouse Preferences**. In the **Buttons** tab, click the **Left-handed mouse** check box and click

[1] https://fedorahosted.org/liberation-fonts/

> **Close** to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).
>
> To insert a special character into a **gedit** file, choose **Applications** → **Accessories** → **Character Map** from the main menu bar. Next, choose **Search** → **Find…** from the **Character Map** menu bar, type the name of the character in the **Search** field and click **Next**. The character you sought will be highlighted in the **Character Table**. Double-click this highlighted character to place it in the **Text to copy** field and then click the **Copy** button. Now switch back to your document and choose **Edit** → **Paste** from the **gedit** menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in proportional bold and all distinguishable by context.

Mono-spaced Bold Italic or ***Proportional Bold Italic***

Whether mono-spaced bold or proportional bold, the addition of italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

> To connect to a remote machine using ssh, type **ssh *username*@*domain.name*** at a shell prompt. If the remote machine is **example.com** and your username on that machine is john, type **ssh john@example.com**.
>
> The **mount -o remount *file-system*** command remounts the named file system. For example, to remount the **/home** file system, the command is **mount -o remount /home**.
>
> To see the version of a currently installed package, use the **rpm -q *package*** command. It will return a result as follows: ***package-version-release***.

Note the words in bold italics above — username, domain.name, file-system, package, version and release. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

> Publican is a *DocBook* publishing system.

1.2. Pull-quote Conventions

Terminal output and source code listings are set off visually from the surrounding text.

Output sent to a terminal is set in **mono-spaced roman** and presented thus:

```
books        Desktop   documentation  drafts  mss    photos   stuff  svn
books_tests  Desktop1  downloads      images  notes  scripts  svgs
```

Source-code listings are also set in **mono-spaced roman** but add syntax highlighting as follows:

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;
```

```
public class ExClient
{
   public static void main(String args[])
       throws Exception
   {
      InitialContext iniCtx = new InitialContext();
      Object         ref    = iniCtx.lookup("EchoBean");
      EchoHome       home   = (EchoHome) ref;
      Echo           echo   = home.create();

      System.out.println("Created Echo");

      System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
   }
}
```

1.3. Notes and Warnings

Finally, we use three visual styles to draw attention to information that might otherwise be overlooked.

Note

Notes are tips, shortcuts or alternative approaches to the task at hand. Ignoring a note should have no negative consequences, but you might miss out on a trick that makes your life easier.

Important

Important boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring a box labeled 'Important' will not cause data loss but may cause irritation and frustration.

Warning

Warnings should not be ignored. Ignoring warnings will most likely cause data loss.

2. We Need Feedback!

If you find a typographical error in this manual, or if you have thought of a way to make this manual better, we would love to hear from you! Please submit a report in Bugzilla: *http://bugzilla.redhat.com/bugzilla/* against the product **JBoss Enterprise SOA Platform.**

When submitting a bug report, be sure to mention the manual's identifier: *SOA_Getting_Started_Guide*

If you have a suggestion for improving the documentation, try to be as specific as possible when describing it. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

Overview

This document is a guide to the JBoss Enterprise SOA Platform 4.3.CP05 product for JBoss administrators and developers.

This document contains the following content:

- Chapter 1 contains a brief overview of both this document and the JBoss Enterprise SOA Platform.
- Chapter 2 contains instructions on how to download, install, configure and launch the JBoss Enterprise SOA Platform.
- Chapter 3 contains instructions on how to launch, shutdown and troubleshoot the JBoss Enterprise SOA Platform.
- Chapter 4 contains an overview of the JBoss ESB *quick starts*. The quick starts are sample projects that demonstrate how to use various features of JBoss Enterprise SOA Platform.
- The Appendices contain additional information referenced in this Guide, such as how to install a Java Virtual Machine (JVM) and Apache Ant.

Refer to the Release Notes for more information on the components included in this product. They can be found at: *http://docs.redhat.com/docs/en-US/JBoss_Enterprise_SOA_Platform/4.3/html/4.3.CP05_Release_Notes/index.html*.

1.1. What is the JBoss Enterprise SOA Platform?

The JBoss Enterprise SOA Platform is a certified, tested and supported platform for developing *Enterprise Application Integration* (EAI) and *Service-Oriented Architecture* (SOA) solutions.

The JBoss Enterprise SOA Platform provides the following features:

- a full Java EE 5 compliant application server (the JBoss Enterprise Application Platform)
- an *Enterprise Service Bus* for message routing and service registration (JBossESB)
- a Business Process Management system (jBPM)
- a Business Rule Engine (JBoss Rules)

Set Up

Read this chapter to learn how to install and configure the JBoss Enterprise SOA Platform.

2.1. Requirements

To install and use the JBoss Enterprise SOA Platform product, your environmentmust meet the following requirements:

- to run the server, you need a Java Virtual Machine (JVM) and a database server.
- to use the Database Schema Configuration Tool and deploy the JBossESB quick start examples, you need Apache Ant.
- the presence of an archiving tool that is able to extract the contents of ZIP files.

Red Hat tests and certifies the JBoss Enterprise SOA Platform against several different hardware platforms, Java Virtual Machines, operating systems and databases. This is an ongoing process and the list of supported environments is always growing.

The list of the currently supported environments for this release can be found at *http://www.jboss.com/products/platforms/soa/testedconfigurations/*

Warning

The Hypersonic database system is included with the JBoss Enterprise SOA Platform for development and evaluation purposes only. Hypersonic is not supported by Red Hat in production environments. To learn more about this, go to *https://access.redhat.com/kb/docs/DOC-41794*.

2.2. Downloading

The JBoss Enterprise SOA Platform 4.3.CP05 from the Red Hat Customer Portal at *https://access.redhat.com/jbossnetwork/restricted/listSoftware.html?product=soaplatform*.

There are several packages available for download. The SOA Platform and SOA Standalone packages provide the software that is required for installation. Red Hat recommends performing a checksum validation on all files that you download from the Customer Support Portal to ensure that they have not been corrupted or altered. Refer to *Appendix C, Verifying Downloaded Files* for directions.

Downloadable Packages

SOA Platform
: The SOA Platform package is a complete JBoss application deployment environment. This single installation provides a complete environment one-stop solution for deploying for SOA applications, including **Seam**, **Hibernate**, clustering, transactions and other services.

SOA Standalone
: The SOA Standalone package provides a light-weight solution for deployments where only the core SOA functionality is needed. The Standalone edition does not include JBoss EAP components that are not needed by the core SOA components.

SOA Platform Source Code
: The package contains the complete source code for the JBoss Enterprise SOA Platform product.

SOA Platform JavaDocs
: The package contains the complete JavaDocs for the APIs provided by the JBoss Enterprise SOA Platform product.

2.3. Installation

This section outlines the procedure for installing the JBoss Enterprise SOA Platform.

Procedure 2.1. Installation

1. **Download the Package to install**
   Download the required JBoss Enterprise SOA Platform 4.3.CP05 package from the Red Hat Customer Portal.

   Refer to *Section 2.2, "Downloading"* for details of the available packages to download.

2. **Verify `zip` File Integrity**
   Perform a checksum validation on the downloaded package ZIP file. This is to ensure that the file has not been corrupted or altered. Refer to *Appendix C, Verifying Downloaded Files* for directions.

3. **Extract the Installation Directory**
   The package ZIP file contains a single directory, called the *installation directory*. Extract this directory from the ZIP file to the location in which it is to be installed.

   In the SOA Platform package this is called the **`jboss-as`** directory. (It is the **`jboss-esb`** directory in the Standalone edition).

   The **`jboss-as`** directory (or **`jboss-esb`** in the Standalone package) contains the application server files and is referred to as the *SOA Root* directory. This directory is usually referred to with the placeholder **`SOA_ROOT`**.

4. **Installation Complete**
   The JBoss Enterprise SOA Platform is now installed and needs to be configured.

   Refer to *Section 2.4, "Configuration"* for configuration tasks.

   Refer to *Chapter 3, Operation* for how to launch the server.

2.3.1. Server Installation

Server installation is commonly used in production environments. When installed with this procedure, the JBoss Enterprise SOA Platform server launches automatically without user intervention when the machine starts up. It requires administrative privileges.

Procedure 2.2. Service Installation on Red Hat Enterprise Linux

1. **Download the JBoss Enterprise SOA Platform `ZIP` File**
   Download the JBoss Enterprise SOA Platform from the Red Hat JBoss Customer Support Portal (CSP) at *https://support.redhat.com/jbossnetwork/restricted/listSoftware.html?product=soaplatform*.

   There are two different `zip` files available. Refer to the section on Components to determine the correct one for your needs.

2. **Verify `zip` File Integrity**
   Perform a checksum validation. This is to ensure that the file has not been corrupted or altered.

3. **Expand the `zip` File**
   Extract the folder from the zip file to the directory in which it is to be installed. This directory is referred to as the *installation directory*.

   The **`jboss-as`** directory (or **`jboss-esb`** in the Standalone edition) contains the application server files and is referred to as the SOA Root directory, or *$SOA_ROOT*.

4. **Add a JBoss User**
   Use the **`adduser`** command to create a system user account for the JBoss Enterprise SOA Platform. Do this as the `root` user.

   ```
   adduser jboss
   ```

5. **Assign ownership of files**
   Use the **`chown`** and **`chgrp`** commands on the *installation directory* to change the `owner` and `group` of the JBoss Enterprise SOA Platform files to the user created in the previous step.

   ```
   [localhost]# chown -R jboss jboss-soa-p.4.3.0
   [localhost]# chgrp -R jboss jboss-soa-p.4.3.0
   ```

6. **Edit `jboss_init_redhat.sh`**
   The script **`jboss_init_redhat.sh`** in the **`bin`** directory is the script used to launch the server as a service. Add two lines to the beginning of this script so that the first three lines look like those below.

   ```
   #!/bin/sh
   #chkconfig: 2345 80 05
   #description: JBoss Enterprise SOA Platform Server
   ```

   These lines are needed by the **`chkconfig`** command.

7. **Set values in `jboss_init_redhat.sh`**
   Edit **`jboss_init_redhat.sh`** in the **`bin`** directory so that the variables match the installation.

   Table 2.1. Script Variables

Variable	Description
`JBOSS_HOME`	This is the path of the *$SOA_ROOT* directory. This value *must* be set here. The example sets it to **`/opt/jboss-soa-p-platform/jboss-esb`**. `JBOSS_HOME=${JBOSS_HOME:-"/opt/jboss-soa-p-platform/jboss-esb"}`
`JBOSS_USER`	This is the user created previously. It is to be used for running the JBoss Enterprise SOA Platform. The example sets it to the username of **`jboss`**. `JBOSS_USER=${JBOSS_USER:-"jboss"}`
`JBOSS_CONF`	This is the name of the server configuration that will be used. The example sets it to **`default`**.

Variable	Description
	`JBOSS_CONF=${JBOSS_CONF:-"default"}`

`JBOSS_HOST`	`JBOSS_HOST` must be specified when binding the JBoss Enterprise SOA Platform server to a specific IP address. This must be done before `JBOSS_HOST` is used by `JBOSS_BIND_ADDR`.
	
	This must be configured to make the server available on the network. The default configuration binds the server to the IP address of `127.0.0.1`.
	
	The example sets it to **`10.1.1.83`**
	
	`#if JBOSS_HOST specified, use -b to bind jboss services to that address`
	`JBOSS_HOST=10.1.1.83`
	`JBOSS_BIND_ADDR=${JBOSS_HOST:+"-b $JBOSS_HOST"}`

8. **Link script into `init.d`**
   Create a symbolic link to **`jboss_init_redhat.sh`** in the directory **`/etc/init.d/`**. The name of the symbolic link's target is that of the new service.

   The example uses the name **`jboss_soa`**.

```
[localhost]# ln -s /opt/jboss-soa-p-platform/jboss-esb/bin/jboss_init_redhat.sh /etc/
init.d/jboss_soa
```

9. **Activate Service**
   Use the **`chkconfig`** tool with the *`--add`* to add the new service to the system configuration.

   The example uses the service name of **`jboss_soa`**.

```
[localhost]# chkconfig --add jboss_soa
```

10. **Installation Complete**
    The JBoss Enterprise SOA Platform is now installed.

2.4. Configuration

Read this section to learn the basic configuration steps that must be performed before the JBoss Enterprise SOA Platform server is started for the first time.

Warning

Many of these steps are of critical importance when running the JBoss Enterprise SOA Platform in a production environment as they affect the security of the server.

2.4.1. Creating Server Console User Accounts

Read this section to learn how to create user accounts for access to the server consoles. You must create at least one account as no accounts are provided for you by default.

The JBoss Enterprise SOA Platform uses the *Java Authentication and Authorization Service* (JAAS) for managing user logins. The default configuration uses plain text files to store user names,

passwords and roles. There are also modules provided for other methods of authorisation such as LDAP and, furthermore, you can create custom modules. Refer to the JBoss Enterprise Application Platform *Security Guide* to learn more about JAAS configuration.

Procedure 2.3, "Adding a new User" shows how to use the default configuration to add users.

Procedure 2.3. Adding a new User

1. **Add a username and password**
   Open the ***SOA_ROOT*/server/*PROFILE*/conf/props/soa-users.properties** file in a text editor. Add the required `user name` and `password` on a new line, using this syntax: ***username=password***.

   ```
   #admin=admin
   harold=@dm1nU53r
   ```

   Any line in this file that begins with a hash (#) is ignored. You can use this to temporarily disable a login.

2. **Assign a user role**
   Open the ***SOA_ROOT*/server/*PROFILE*/conf/props/soa-roles.properties** file in a text editor. Add the user and assigned roles to the file on a new line, using this syntax: **username=role1,role2,role3**. You can assign any number of roles. Note that a user must be assigned the `JBossAdmin`, `HttpInvoker`, `user`, and `admin` roles in order to be able to log into the server consoles.

   ```
   #admin=JBossAdmin,HttpInvoker,user,admin
   harold=JBossAdmin,HttpInvoker,user,admin
   ```

   Any line in this file that begins with a hash (#) is ignored. You can use this to temporarily disable user roles.

3. **Save**
   Save the changes to both files and then you will be able to log in to the server console. You are not required to restart the server.

If you use the default log in configuration, users and roles do not need to correspond with any other user accounts, such as operating system ones. They only relate to the JBoss Enterprise SOA Platform, so you can be create them on an arbitrary basis.

2.4.2. Using the JBoss Enterprise SOA Platform in a Development Environment

Use the `default` server profile for development and testing. The `production` server profile should not be used for development unless you are specifically testing clustering.

2.4.2.1. Disabling head-less mode

By default, every the server profile runs in *headless* mode. This means that code invoked from the server does not have access that server's physical display output device. Red Hat recommends this configuration for production use. In development and testing environments it can be useful to disable headless mode, though.

Procedure 2.4. Disabling headless mode

1. **Create a profile configuration file**
   Copy the default server configuration file (**`run.conf`** for Red Hat Enterprise Linux or **`run.conf.bat`** for Microsoft Windows) from the ***`SOA_ROOT`*`/bin`** directory to ***`SOA_ROOT`*`/server/`*`PROFILE`*`/`**

2. **Open file in editor**
   Open the new file in a text editor and locate the line on which **`java.awt.headless`** is set.

```
if [ "x$JAVA_OPTS" = "x" ]; then
   JAVA_OPTS="-Xms1303m -Xmx1303m -XX:MaxPermSize=256m -Djava.awt.headless=true
 -Dorg.apache.xml.dtm.DTMManager=org.apache.xml.dtm.ref.DTMManagerDefault
 -Dorg.jboss.resolver.warning=true -Dsun.rmi.dgc.client.gcInterval=3600000 -
Dsun.rmi.dgc.server.gcInterval=3600000 -Dsun.lang.ClassLoader.allowArraySyntax=true"
```

3. **Change value**
   Change the value on that line from **`true`** to **`false`**.

```
if [ "x$JAVA_OPTS" = "x" ]; then
   JAVA_OPTS="-Xms1303m -Xmx1303m -XX:MaxPermSize=256m -Djava.awt.headless=false
 -Dorg.apache.xml.dtm.DTMManager=org.apache.xml.dtm.ref.DTMManagerDefault
 -Dorg.jboss.resolver.warning=true -Dsun.rmi.dgc.client.gcInterval=3600000 -
Dsun.rmi.dgc.server.gcInterval=3600000 -Dsun.lang.ClassLoader.allowArraySyntax=true"
```

4. **Save and re-launch**
   Save the file. The next time you start the server, it will no longer be in headless mode.

2.4.3. Database Configuration

The JBoss Enterprise SOA Platform uses a database to *persist* registry services and the message store. Note that you must alter the database configuration before you use it in a production environment. The default database configuration uses the included Hypersonic database. Hypersonic can only be used for testing and demonstration purposes. It is not suitable for production deployment and Red Hat does not support it.

Using the Database Configuration Script

The Database Configuration Script is an Apache Ant script. Use it to change the database used by the JBoss Enterprise SOA Platform. The Database Configuration Script is in the ***SOA_ROOT*/jboss-as/tools/schema/** directory.

Warning

Only use the Database Configuration Script to initially change the database configuration. It can only be run once and must be run before any other changes are made. If you try to run the script on a configured installation, it may not work as intended.

Warning

Be aware that you can only use one database. You can not use different databases for different components.

Check that the following pre-requisites are satisfied before running the Database Configuration Script:

- Apache Ant is installed.
- The database that you wish to use already exists.
- A user with permission to make changes to that database exists.
- The JDBC driver JAR file for the database is in the server configuration's **lib/** directory.

Procedure 2.5. Running the Database Configuration Tool

1. At a command line prompt, change to the directory containing the Database Configuration script.
2. **Run Ant**
   Run the **ant** command to launch the script.
3. **Enter Data**
   Following the prompts, enter the following information when it is requested:
   - the type of database being used,
   - the name of the database,
   - the host-name or IP Address of the database,
   - the TCP port being used for the database,
   - the user-name needed to access the database, and
   - the password for this user account.

   Any of these values can be added to the **build.properties** file in the same directory before running the script. Those properties that are found in this file will not be prompted for.
4. **Done**
   The script then updates the relevant configuration files and exits.

Example 2.1. The Database Configuration Script

```
[schema]$ ant
[ schema]$ ant
Buildfile: build.xml

[echo] JBoss SOA Platform
[echo] ------------------
[echo] Database Configuration Script
[echo]
[echo] This script is used to configure the SOA platform and all it's
[echo] constituent components against a new RDBMS. Deployment scripts
[echo] are currently available for the following databases: MySQL,
[echo] Oracle9i, PostgresSQL, Oracle10g
[echo] ------------------
[echo] NOTICE: THIS SCRIPT MAY NOT WORK CORRECTLY IF YOU HAVE MADE MANUAL
[echo] CHANGES TO THE DATABASE CONFIGURATION.
[echo]
```

2.4.4. Securing Non-Remote Method Invocation Classes on Port 8083

Client applications can utilise *Remote Method Invocation* (RMI) to download Enterprise Java Bean classes through `port 8083`. However, you can also configure RMI to allow client applications to download any deployed resources you desire. Configure this behavior by changing the settings in the ***SOA_ROOT*/server/*PROFILE*/conf/jboss-service.xml** file.

Example 2.2. Configuration Setting that Permits Download of Non-EJB Classes

```
<!-- Should non-EJB .class files be downloadable -->
<attribute name="DownloadServerClasses">false</attribute>
```

Set this value to **`false`** to allow client applications to download Enterprise Java Bean classes only.

Important

This value is set to **`false`** by default in the SOA Platform's `production` profile. The value is set to **`true`** in all other cases, (including the SOA Standalone package's `default` profile.) Note that this is not a secure configuration and should only be used in development environments.

2.4.5. Running Application Servers Side-by-Side

The JBoss Enterprise SOA Platform can be made to run alongside another JBoss product such as the JBoss Enterprise Application Platform. There are two ways of achieving this:

1. by using *multi-homing*. This involves configuring a network interface so it is assigned multiple IP addresses and then launching each JBoss Enterprise Platform server using the **`-b`** parameter to *bind* each of them to a single IP address.

   Refer to your operating system's documentation for instructions on configuring multi-homing.

2. by using the *Service Bindings Manager*. The Service Bindings Manager provides a mechanism for supplying a different port configuration to each JBoss Enterprise Platform server instance.

Refer to the JBoss Enterprise Application Platform *Getting Started Guide* for instructions on using the `Service Bindings Manager`: *http://docs.redhat.com/docs/en-US/JBoss_Enterprise_Application_Platform/5/html/Getting_Started_Guide/*.

Warning

Starting multiple servers with the same database configuration simultaneously can result in database errors if the database has not been initialized previously. The server initialises a new database when it is launched with it for the first time. If more than one server attempts to do this simultaneously then errors can occur. To avoid this problem, wait until one instance has finished starting and initializing the databases before starting the other instances. This only has to be done the first time that the servers are started.

Operation

Read this chapter to learn about how to launch, shutdown and troubleshoot the JBoss Enterprise SOA Platform.

3.1. Start the server

Follow this procedure to start the JBoss Enterprise SOA Platform:

Procedure 3.1. Server startup

1. **Go to the `bin` directory**
   Launch a terminal and navigate to the ***`SOA_ROOT`*`/bin/`** directory.

2. **Run the start-up script**
   Run the **`./run.sh`** command (for Linux/UNIX) or the **`run.bat`** command (for Microsoft Windows) to start the server.

   If you run this command without any parameters, the server will use the **`default`** profile, bound to an IP address of `127.0.0.1`.

   Using command line parameters will change these settings. Some of the most common ones are:

   - *`-c`* Use this to launch the server with a specific profile.
   - *`-b`* Use this bind the server to a specific IP address.

   This example shows the server being launched bound to IP address `10.34.5.2`, with the `production` configuration specified:

   ```
   [localhost]$ ./run.sh -c production -b 10.34.5.2
   ```

   For a complete list of command line parameters, refer to the JBoss Enterprise Application Platform's *Getting Started Guide*.

3. **Verify**
   The JBoss Enterprise SOA Platform should now be running. To verify this, launch a web browser and access the server console at *http://127.0.0.1:8080*.

3.1.1. Launching and Controlling a Server Installation

A server installation has the ability to start and shut down automatically with the host operating system.

To manually start and stop it on Red Hat Enterprise Linux, switch to `root` and use the **`service`** command. The examples below assume that the service was installed using the name `jboss_soa`.

To start the server:

```
[localhost]# service jboss_soa start
```

To stop the server:

```
[localhost]# service jboss_soa stop
```

Note

If the JBoss user was created as a system account (with the *-r* switch) then a warning message is displayed. This can be ignored.

```
su: warning: cannot change directory to /home/jboss: No such file or directory
```

3.2. Stopping the server

Follow this procedure to shut down a the JBoss Enterprise SOA Platform:

Procedure 3.2. Stopping the Server

- **Press control+c**
  To halt the JBoss Enterprise SOA Platform server, press control+c in the terminal window. The server will shut down all the different services that are running and then exit.

3.3. Troubleshooting

Here are the solutions to some of the problems users most commonly encounter.

`JBOSS_HOME` set incorrectly
: If the optional environmental variable, `JBOSS_HOME`, is set then it must point to the correct directory. If you have multiple installations, check that it is pointing to the one that you are trying to run.

Warning

Do not set this variable unless you have a specific need to do so. Refer to *Appendix E, Option: Add `JBOSS_HOME` environment variable.* for more information.

Java installed incorrectly
: If the Java environment has been installed or configured incorrectly, then the JBoss Enterprise SOA Platform will not function. Refer to *Appendix A, Installing a Java Development Kit on Red Hat Enterprise Linux* for details.

VM Cannot Allocate Sufficient Memory
: This error occurs when there is not enough free memory available to the system to satisfy the JBoss Enterprise SOA Platform's requirements. You can increase the amount available in one of three ways: by exiting applications, allocating more virtual memory, or physically increasing the amount of RAM installed on the system.

Important

Refer to the *Release Notes* for details of known bugs and potential upgrade issues affecting this release.

Developer Examples

This chapter contain an introduction to the JBoss Enterprise SOA Platform for developers.

4.1. JBossESB Quickstarts

The `quick starts` are sample projects. Each one demonstrates how to use a specific piece of functionality when building JBoss ESB Services. There are several dozen `quick starts` included in the ***SOA_ROOT*/samples/quickstarts/** directory. Every quick start is built and deployed using Apache Ant.

Section 4.1.1, "The "helloworld" Quick Start" provides a detailed explanation of the **helloworld** `quick start`.

To learn more about each `quick start`:

- study the quick start's **readme.txt** file.
- run the **ant help-quickstarts** command in the `quick-starts` directory.
- run the **ant help** command in the `quick-starts` directory.

When running a quick start, make note of the following:

1. each quick start needs to be built and deployed using Apache Ant. Refer to *Appendix D, Installing Apache Ant*.
2. each quick start uses the **samples/quickstarts/conf/quickstarts.properties** file to store environment-specific configuration options. An example properties file — **quickstarts.properties-example** — is included.
3. each quick start has different requirements. These are documented in their individual **readme.txt** files.
4. not every quick starts can run under every server profile.
5. The jBPM quick starts require a valid jBPM Console user name and password. Supply these by adding them as properties in the **samples/quickstarts/conf/quickstarts.properties** file:

   ```
   # jBPM console security credentials
   jbpm.console.username=admin
   jbpm.console.password=adminpassword
   ```

   The quick starts that are affected by this requirement are **bpm_orchestration1**, **bpm_orchestration2**, **bpm_orchestration3** and **bpm_orchestration4**.
6. you can only execute some of the quick starts (such as **groovy_gateway**) if the server is not running in *headless* mode. (The JBoss Enterprise SOA Platform is configured to launch in head-less mode by default.)

   Red Hat recommends that you run production servers in headless mode only but you can change the mode for use in development environments. Refer to *Section 2.4.2.1, "Disabling head-less mode"*.

These `quick starts` demonstrate the most commonly-used features:

helloworld	**business_service**	**aggregator**
helloworld_action	**business_rules_service**	**bpm_orchestration1**
custom_action	**scripting_groovy**	**bpm_orchestration2**
helloworld_file_action	**transform_CSV2XML**	**webservice_consumer1**
helloworld_ftp_action	**transform_XML2POJO**	**webservice_producer**
simple_cbr	**transform_XML2XML_simple**	
fun_cbr	**transform_XML2XML_date_manipulation**	

4.1.1. The "helloworld" Quick Start

The **helloworld** `quick start` is a simple demonstration of the JBoss Enterprise SOA Platform's key features and concepts. It is found in the ***SOA_ROOT*/samples/quickstarts/helloworld/** directory. Follow the procedure below to run it.

Procedure 4.1. Running the **helloworld** quick start

1. **Start the Server**
   Ensure that the server is running. Refer to *Section 3.1, "Start the server"*.

2. **Set the quick start's properties**
   Check that the ***SOA_ROOT*/samples/quickstart/conf/quickstarts.properties** file has been configured to use the correct settings and home directory.

   ```
   # Location of your JBoss Application Server installation.
   org.jboss.esb.server.home=/opt/jboss-soa-p.4.3.0/jboss-as
   # JBossAS server name.
   org.jboss.esb.server.config=default
   ```

3. **Build and Deploy the Quick Start**
   Launch a terminal window and navigate to the ***SOA_ROOT*/samples/quickstarts/helloworld** directory.

   Run the **ant deploy** command. This builds the **helloworld.esb** archive and deploys it.

4. **Invoke the Service**
   Run the **ant runtest** command. This sends a JMS message to the ESB service that you deployed in the previous step.

5. **Check the server for the result:**
   The "hello world" message is now appended to the ***PROFILE*/log/server.log** file.

   It will look like this:

   ```
   17:25:12,614 INFO [STDOUT] &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
   17:25:12,615 INFO [STDOUT] Body: Hello World
   17:25:12,615 INFO [STDOUT] &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
   17:25:12,615 INFO [STDOUT] Message structure:
   17:25:12,615 INFO [STDOUT] [ message: [ JBOSS_XML ]
   ```

   Depending on the server profile you are using. it may also be output to the physical server's display. (The `production` profile only outputs it to the log file.)

Figure 4.1, "helloworld Components and Sequence of Events" shows the **helloworld** `quick-start`'s components and depicts the sequence of events that occur when it runs.

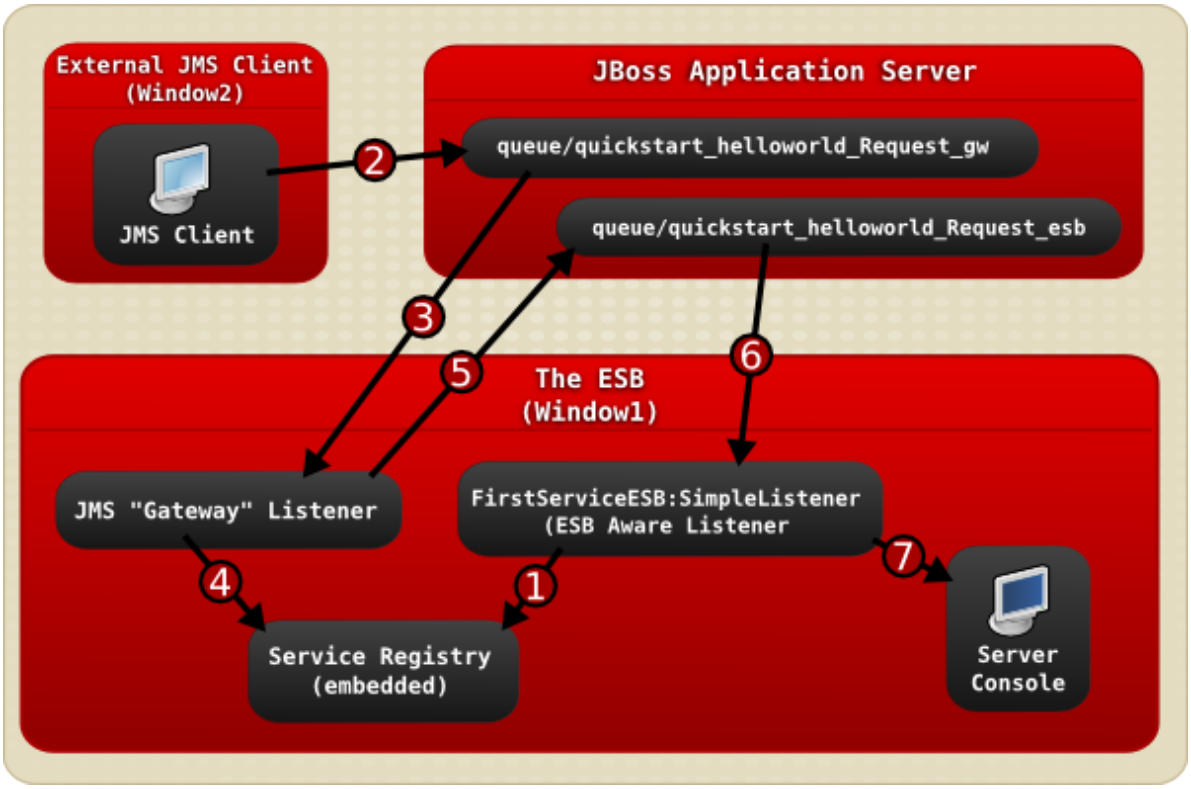

Figure 4.1. helloworld Components and Sequence of Events

The helloworld quick start makes use of the following components:

Gateway Listener
: A `Gateway Listener` is one of the key architectural components of the JBoss Enterprise SOA Platform. This type of `listener` provides an interface between the ESB and external service end-points. In this particular case, a Java Message Service Gateway is used.

Service Registry
: This is a JAXR `registry` implementation. The `registry` uses *Remote Method Invocation*-based communications for this quick start. To learn more about the `registry`, please refer to the *ESB Services Guide*.

The ESB-Aware Service Listener
: The `FirstService:ESB:SimpleListener` listens to queue/quickstart_helloworld_Request_esb for ESB-aware messages. The concepts of *ESB-aware* and *unaware* messages are discussed in the next section of this *Guide*.

Note

If application clients and services are referred to as being *ESB-aware*, this means that they can understand the message format and transport protocols used by the SOA Platform. (Gateways are used to convert messages from ESB-unaware services into formats that can be processed and routed by the Enterprise Service Bus.)

Figure 4.1, "helloworld Components and Sequence of Events" depicts the following sequence of events:

1. The JBoss Enterprise SOA Platform server is launched in `Window1` and then the `FirstServiceESB:SimpleListener` service is added to the Service Registry service when the helloworld quick start is deployed.

2. A JMS client sends an ESB-unaware "Hello World" message, (it is a plain **`String`** object), to the JMS Queue (`queue/quickstart_helloworld_Request_gw`.)

3. The JMS Gateway Listener receives the ESB-unaware message and creates from it an ESB-aware message for use by ESB-aware end-points.

4. The `JMS Gateway Listener` uses the `registry` to find the `FirstServiceESB:SimpleListener` service's *end-point reference* (EPR). In this case, the EPR is the `queue/quickstart_helloworld_Request_esb` JMS queue.

5. The `JMS Gateway Listener` takes the new ESB-aware message and sends it to the `queue/quickstart_helloworld_Request_esb` JMS queue.

6. The `FirstServiceESB:SimpleListener` service receives the message.

7. The `FirstServiceESB:SimpleListener` service extracts the payload from the message and outputs it to the console.

Appendix A. Installing a Java Development Kit on Red Hat Enterprise Linux

Follow these instructions to install a Java Development Kit (JDK) on Red Hat Enterprise Linux using Red Hat Network (RHN).

If you have difficulties performing the installation using RHN please refer to the RHN Help Desk at *https://rhn.redhat.com/rhn/help/* or contact Red Hat Support directly for assistance.

Important

The following procedure require that the commands be run with administrator privileges. This means running the commands using the **sudo** command, logging in as the `root` user, or by switching to the `root` user with the **su** command.

A.1. OpenJDK on Red Hat Enterprise Linux 5

Use this procedure to install **OpenJDK** on **Red Hat Enterprise Linux 5**.

Procedure A.1. Installing OpenJDK on Red Hat Enterprise Linux 5

1. **Subscribe to the `base channel`**
   The **OpenJDK** is available in the RHN `base channel`. By default, all installations of Red Hat Enterprise Linux are subscribed to this channel.

2. **Install the Package**
   Use the **`yum`** utility to install **OpenJDK**:

   ```
   [localhost]# yum install java-1.6.0-openjdk-devel
   ```

3. **Verify System Default JDK**
   To ensure that the correct JDK is set as the system default, run the **`alternatives`** command as described in *Section A.4, "Setting the Default JDK with the **`alternatives`** Utility"*

A.2. Oracle Java Development Kit on Red Hat Enterprise Linux 5

Use this procedure to install the Oracle Java Development Kit (JDK) on Red Hat Enterprise Linux 5.

Procedure A.2. Installing the Oracle JDK on Red Hat Enterprise Linux 5

1. **Subscribe to `Supplementary Server` channel**
   The Oracle JDK is available in the `Supplementary Server` channel. One will need to ensure that the machine is subscribed to this channel in order to install this package.

2. **Install the Package**
   Use the **`yum`** command to install the Oracle JDK package:

   ```
   [localhost]# yum install java-1.6.0-sun-devel
   ```

3. **Verify System Default JDK**
   To ensure that the intended JDK is set as the system default, run the **alternatives** command as described in *Section A.4, "Setting the Default JDK with the **alternatives** Utility"*

A.3. Installing the Oracle JDK on Red Hat Enterprise Linux AS/ES 4

Use this procedure to install the Oracle Java Development Kit on Red Hat Enterprise Linux AS or ES 4.

Procedure A.3. Installing the Oracle JDK on Red Hat Enterprise Linux AS/ES 4

1. **Subscribe to the `Extras` Channel**
   The **Oracle Java Development Kit** is available in the `Red Hat Extras` channel. Ensure that the machine is subscribed to this channel in order to install this package.

2. **Install using `up2date`**
   Run this command to install the package:

   ```
   [localhost] up2date java-1.6.0-sun-devel
   ```

3. **Check That it is the System's Default Java Development Kit**
   To ensure that the intended JDK is set as the system default, run the **alternatives** command as described in *Section A.4, "Setting the Default JDK with the **alternatives** Utility"*

A.4. Setting the Default JDK with the `alternatives` Utility

alternatives is a tool for managing different software packages that provide the same functionality. Red Hat Enterprise Linux uses **alternatives** to ensure that only one Java Development Kit (JDK) is set as default at a time.

Installing a JDK from Red Hat Network will normally result in an automatically configured system. However, if you have installed multiple JDKs it is possible that there may be conflicts. Perform the following steps to verify that the everything is set up correctly.

Example A.1. Using **alternatives** to Configure the Default Java Development Kit

```
[localhost]# /usr/sbin/alternatives --config java

There are 2 programs which provide 'java'.

  Selection    Command
-----------------------------------------------
   1           /usr/lib/jvm/jre-1.4.2-gcj/bin/java
*+ 2           /usr/lib/jvm/jre-1.6.0-openjdk/bin/java

Enter to keep the current selection[+], or type selection number:
```

Procedure A.4. Using **alternatives** to Set the Default JDK

1. **Set `java`**
   Input this command: **/usr/sbin/alternatives --config java**

   Next, follow the on-screen directions to ensure that the correct version of **java** is selected. *Table A.1, "**java** alternative commands"* shows the relevant command settings for each of the different JDKs.

Table A.1. **java** alternative commands

JDK	alternative command
OpenJDK 1.6	**/usr/lib/jvm/jre-1.6.0-openjdk/bin/java**
Oracle JDK 1.6	**/usr/lib/jvm/jre-1.6.0-sun/bin/java**

2. **Set javac**
   Enter this command: **/usr/sbin/alternatives --config javac**

   Follow the on-screen directions to ensure that the correct version of **javac** is selected. *Table A.2, "**javac** alternative commands"* shows the appropriate command settings for the different JDKs.

   Table A.2. **javac** alternative commands

JDK	alternative command
OpenJDK 1.6	**/usr/lib/jvm/java-1.6.0-openjdk/bin/javac**
Oracle JDK 1.6	**/usr/lib/jvm/java-1.6.0-sun/bin/javac**

3. **Extra Step: Set java_sdk_1.6.0**
   The **Oracle JDK 1.6** requires an additional command be run:

   /usr/sbin/alternatives --config java_sdk_1.6.0

   Follow the on-screen directions to ensure that the correct **java_sdk** is selected. It is **/usr/lib/jvm/java-1.6.0-sun**.

Appendix B. Installing the JDK on on Microsoft Windows

Procedure B.1. Installing the 32-bit Oracle JDK on Microsoft Windows

1. **Download the Software**
   Download the Oracle Java 2 Development Kit from *http://www.oracle.com/technetwork/java/javase/downloads/index.html*.

2. **Create the JAVA_HOME variable**
   Create an environmental variable called **`JAVA_HOME`** that points to directory in which the JDK will be installed, e.g. **`C:\Program Files\Java\jdk1.6.0_16\`**.

   To do this, click on the **Start Menu**, open the **Control Panel**, (if necessary, switch to **Classic View**), open the **System Control Panel** applet, select the **Advanced Tab**, and click on the **Environment Variables** button.

3. **Add the JDK to the PATH**
   Add the JDK's **`bin`** directory to the path PATH.

   To do this, open the **Control Panel** from the **Start Menu**, (if necessary, switch to **Classic View**), then edit the PATH environment variable found in **System** -> **Advanced** -> **Environment Variables** -> **System Variables**. Append a semicolon and **`%JAVA_HOME%\bin`** to the end of the PATH value.

4. **Add the JRE to the PATH**
   So that Java can be run from the command line, add the **`jre\bin`** directory to the path so that it looks similar to **`C:\Program Files\Java\jdk1.5.0_11\jre\bin`**.

Appendix C. Verifying Downloaded Files

To verify that the **ZIP** files are error-free upon download, go to the **Software Details** page for each in turn. On these pages, you will find MD5 and SHA256 "checksum" values that you can use to check the integrity of the files.

Warning

The checksum values shown in this section are for demonstration purposes only and should not be used to evaluate the authenticity of your download. Use only the checksum values on the download page for the file that you are verifying.

Having obtained the values, follow this process to test the files on a **Red Hat Enterprise Linux** machine:

Procedure C.1. Verifying File Checksums on Red Hat Enterprise Linux

1. **Run a checksum tool on the file**
   Open a terminal window and run either either the **md5sum** or **sha256sum** command, supplying the filename of the downloaded **ZIP** as an argument. The program will output the checksum value for the file. This is shown using both the **md5sum** and **sha256sum** commands below:

   ```
   [localhost]$ md5sum standalone-soa-5.0.0.zip
   4564d1a5190110dbe8170e50d7353a97  standalone-soa-5.0.0.zip
   ```

   ```
   [localhost]$ sha256sum standalone-soa-5.0.0.zip
   25b6bd3c5f47b316639b014d041cdb6a515e3a4a32d30a479141cd8ceecb853e  standalone-
   soa-5.0.0.zip
   ```

2. **Compare the checksum values**
   Compare the checksum value returned by the command to the corresponding value displayed on the **Software Details** page for that **ZIP** file.

   If the two checksum values are not identical, then download the file again. A difference between the checksum values means that the **ZIP** file has either been corrupted during download or has been modified since it was uploaded to the server. If, after several downloads, the checksum will still not successfully validate, please contact Red Hat Support for assistance.

   If the two checksum values are identical then the file has not been altered or corrupted and is, therefore, safe to use.

Important

Microsoft Windows does not come equipped with a checksum tool. Download a third-party MD5 application such as **MD5summer** from *http://www.md5summer.org/*.

Appendix D. Installing Apache Ant

You do not need the Java build tool *Apache Ant* to install or operate the JBoss Enterprise SOA Platform. However, you will need it to run the database configuration script and to build and deploy the quick starts.

To run Apache Ant, you will require a correctly installed Java Runtime Environment (JRE).

Note that, if you are running a development workstation, you may already have Apache Ant installed.

Procedure D.1. Installing Apache Ant on Red Hat Enterprise Linux

- To download and install **`Apache Ant`** on Red Hat Enterprise Linux, execute this command:

```
[localhost]$ sudo yum install ant
```

Procedure D.2. Installing Apache Ant on Other Operating Systems

1. **Download and Extract**
   Download the **`Apache Ant`** binary release from *http://ant.apache.org/bindownload.cgi*.

   Once it is downloaded, extract it in a preferred installation location, such as **`c:\Program Files\Apache\Ant\`** or **`/opt/apache-ant-1.7/`**.

2. **Add the ANT_HOME Environment Variable**
   Create an environmental variable called **`ANT_HOME`**. This variable has to contain the path created in the previous step.

   - To do this on Red Hat Enterprise Linux, add the following line to the **`~/.bash_profile`** file, substituting the path below with that which you created in the previous step.

     ```
     export ANT_HOME=/opt/apache-ant-1.7
     ```

   - On Microsoft Windows, do this by clicking on the **Start Menu**, opening the **Control Panel** and selecting **System** -> **Advanced** -> **Environment Variables**.

     Create a new variable, call it **`ANT_HOME`** and configure it to point to the directory you created in the previous step.

3. **Include `bin` in the `PATH`**
   Append the Ant installation's **`bin`** directory to the **`PATH`** environmental variable.

   - On Unix/Linux systems, do this by adding the following line to the **`~/.bash_profile`** file. (Put it after the one which sets the **`ANT_HOME`** variable):

     ```
     export PATH=$PATH:$ANT_HOME/bin
     ```

   - On Microsoft Windows, do this by opening the **Control Panel** and selecting **System** -> **Advanced** -> **Environment Variables** -> **System Variables**. Edit the PATH variable and append the text **`;%ANT_HOME%\bin`**.

To test the Apache Ant installation, run **`ant -version`** from a terminal. The output should look similar to this:

```
[localhost]$ ant -version
Apache Ant version 1.7 compiled on June 27 2008
```

To learn more about **Apache Ant**, visit *http://ant.apache.org*.

Appendix E. Option: Add JBOSS_HOME environment variable.

You do not need to configure the JBOSS_HOME environment variable to operate the JBoss Enterprise SOA Platform server. However it is sometimes used by other software to determine where the Server is installed. Build scripts, for instance, make use of it when they compile sample applications.

If you do use JBOSS_HOME, make sure you set it the server's root directory (SOA_ROOT) as described in *Procedure 2.1, "Installation"* for .

Important

If you have more than one JBoss server installed on this machine, Red Hat recommends you avoid using JBOSS_HOME if possible or only set its within those scripts that require it.

E.1. Setting JBOSS_HOME on Red Hat Enterprise Linux

To add this variable to your local user account include the following line at the end of your **~/.bash_profile** file. Replace <*path*> with the path to your *$SOA_ROOT* directory.

```
export JBOSS_HOME=<path>
```

E.2. Setting JBOSS_HOME on Microsoft Windows

To set the JBOSS_HOME variable on Microsoft Windows, follow these steps.

Procedure E.1. Adding JBOSS_HOME on Microsoft Windows

1. Open the **System Properties** dialog by double-clicking on the **System** icon in the **Control Panel**.
2. Select the **Advanced** tab, and then click on the **Environment Variables** button to open the **Environment Variables** dialog box.
3. Click on the **New** button in the User variables section at the top of the dialog box.
4. Set the variable name to **JBOSS_HOME**, and set the variable value to the ***$SOA_ROOT*** directory's path.

Appendix F. GNU General Public License 2.0

```
                    GNU GENERAL PUBLIC LICENSE
                       Version 2, June 1991

 Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.

                            Preamble

  The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users.  This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it.  (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.)  You can apply it to
your programs, too.

  When we speak of free software, we are referring to freedom, not
price.  Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

  To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

  For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have.  You must make sure that they, too, receive or can get the
source code.  And you must show them these terms so they know their
rights.

  We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

  Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software.  If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

  Finally, any free program is threatened constantly by software
patents.  We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary.  To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

  The precise terms and conditions for copying, distribution and
modification follow.

                    GNU GENERAL PUBLIC LICENSE
   TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
```

```
  0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License.  The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language.  (Hereinafter, translation is included without limitation in
the term "modification".)  Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope.  The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

  1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

  2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

    a) You must cause the modified files to carry prominent notices
    stating that you changed the files and the date of any change.

    b) You must cause any work that you distribute or publish, that in
    whole or in part contains or is derived from the Program or any
    part thereof, to be licensed as a whole at no charge to all third
    parties under the terms of this License.

    c) If the modified program normally reads commands interactively
    when run, you must cause it, when started running for such
    interactive use in the most ordinary way, to print or display an
    announcement including an appropriate copyright notice and a
    notice that there is no warranty (or else, saying that you provide
    a warranty) and that users may redistribute the program under
    these conditions, and telling the user how to view a copy of this
    License.  (Exception: if the Program itself is interactive but
    does not normally print such an announcement, your work based on
    the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole.  If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works.  But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.
```

```
In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

  3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

    a) Accompany it with the complete corresponding machine-readable
    source code, which must be distributed under the terms of Sections
    1 and 2 above on a medium customarily used for software interchange; or,

    b) Accompany it with a written offer, valid for at least three
    years, to give any third party, for a charge no more than your
    cost of physically performing source distribution, a complete
    machine-readable copy of the corresponding source code, to be
    distributed under the terms of Sections 1 and 2 above on a medium
    customarily used for software interchange; or,

    c) Accompany it with the information you received as to the offer
    to distribute corresponding source code.  (This alternative is
    allowed only for noncommercial distribution and only if you
    received the program in object code or executable form with such
    an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it.  For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable.  However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

  4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License.  Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

  5. You are not required to accept this License, since you have not
signed it.  However, nothing else grants you permission to modify or
distribute the Program or its derivative works.  These actions are
prohibited by law if you do not accept this License.  Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

  6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions.  You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
```

this License.

  7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

  8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

  9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

  10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

                            NO WARRANTY

  11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN

```
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.  THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU.  SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

  12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

                     END OF TERMS AND CONDITIONS

            How to Apply These Terms to Your New Programs

  If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

  To do so, attach the following notices to the program.  It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

    <one line to give the program's name and a brief idea of what it does.>
    Copyright (C) <year>  <name of author>

    This program is free software; you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation; either version 2 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY 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 along
    with this program; if not, write to the Free Software Foundation, Inc.,
    51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License.  Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary.  Here is a sample; alter the names:
```

```
  Yoyodyne, Inc., hereby disclaims all copyright interest in the program
  `Gnomovision' (which makes passes at compilers) written by James Hacker.

  <signature of Ty Coon>, 1 April 1989
  Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs.  If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library.  If this is what you want to do, use the GNU Lesser General
Public License instead of this License.
```

Appendix G. Revision History

Revision 1.5 **Mon Mar 21 2011** **David Le Sage** *dlesage@redhat.com*

Updated for 4.3.CP05 Release
Refactored book to synchronise with SOA Platform 5.1 Guide

Revision 1.4 **Tue Apr 27 2010** **David Le Sage** *dlesage@redhat.com*

Updated for 4.3 CP04
Backported rewritten installation instructions from the SOA-P 5.0 *Getting Started Guide*. Chapter 1.
SOA-1997 - Added details of configuring databases post-install. Section 1.4.3

Revision 1.3 **Tue Apr 20 2010** **David Le Sage** *dlesage@redhat.com*

Updated for SOA 4.3.CP03

Revision 1.2 **Fri 24 Jul 2009** **Darrin Mison** *dmison@redhat.com*

Updated for 4.3 CP02
SOA-1151 - Added details of certified and compatible configurations. Section 1.1.2
Some configuration information has been moved to this guide from the Release Notes. Section 2.1
SOA-1489 - JON ESB plugin directions have been updated. Section 1.6

Revision 1.1 **Wed 28 Jan 2009** **Darrin Mison** *dmison@redhat.com*

Updated for 4.3 CP01
Added Apache Ant installation Instructions
Updated installation and upgrade instructions

Revision 1.0 **Wed 10 Sep 2008** **Darrin Mison** *dmison@redhat.com*

Created