

Install and Configure Java

sd

2009/12/24 13:55

Table of Contents

| | |
|------------------------------------------|---|
| Install and Configure Java | 3 |
| Remove other installations of Java | 3 |
| Install the JDK | 3 |
| Self extracting-binary (bin) | 3 |
| Package manager (RPM) | 4 |
| Install the Web browser plug-in | 4 |

Install and Configure Java

Contents:

- [Remove other installations of Java](#)
- [Install the JDK](#)
 - [Self extracting-binary \(bin\)](#)
 - [Package manager \(RPM\)](#)
- [Install the Web browser plug-in](#)

These are some instructions for installing a working Java Development Kit (JDK) on Red Hat Enterprise Linux, CentOS and Fedora. This process is used in our [server setup](#) and [development setup](#). See our [Java page](#) for further detail on optimising and troubleshooting Java for Pentaho.

For most deployments, we have standardised on the [Sun JDK 1.5.0](#).

A 32-bit JDK is recommended if you want to use the GUI tools (e.g. on a [development workstation](#)). The GUI Pentaho developer tools [have not been completely ported](#) to 64-bit Java (on any operating system). The downside is that Pentaho will be restricted to 4GB of RAM, but at least all the tools will work.

This doesn't affect the [server deployment](#). The Pentaho server and Kettle command line tools will work in 64-bit mode.

More details about this are available on the [Troubleshooting GUI tools](#) page.

Remove other installations of Java

While you can have multiple versions of Java installed (which is what the *alternatives* system used below manages), it can be simpler to have only one. Modern Linux distributions often come with [OpenJDK](#) and/or [GCI](#) available.

Install the JDK

There are two main ways to install the Sun Java Development Kit. Select the installation type (bin/RPM) and architecture (e.g. x86, x64) most appropriate for your needs. You can download them from [here](#).

You can install the JDK any way you like. Pentaho will use the installation defined by the PENTAHO_JAVA_HOME or JAVA_HOME environment variable.

Self extracting-binary (bin)

The self-extracting binary is downloaded as a file with a .bin extension. You just execute the file and it extracts itself to the working directory. This is the simplest way to install, and it can be performed as a non-privileged user (not root).

On the downside, the installation will be specific to that user account only. If this is the only user that will run Java applications, then this is fine. Many of our BI Server deployments are dedicated to that single purpose, in which case giving Java to the user running Pentaho is sufficient. See the [RPM instructions below](#) to make a multi-user installation.

1. Extract the downloaded bundle and accept the licence

```
$ sh jdk-1_5_0_15-linux-amd64.bin
```

You'll notice that it has been extracted to a directory named something like *jdk1.5.0_15*.

2. Move that directory to a manageable location within the current user's home directory

```
$ mv jdk1.5.0_15 $HOME
```

3. We recommend that you create a generic [symbolic link](#) so that you can easily change the JDK later

```
$ cd $HOME
```

```
$ ln -s jdk1.5.0_15 jdk64
```

Pentaho Tutorial - Install and Configure Java

The symbolic link should point to the JDK that you want to use.

4. Configure environment variables: open the file `$HOME/.bashrc` in a text editor and add the following to the end

```
JAVA_HOME=$HOME/jdk64
PENTAHO_JAVA_HOME=$JAVA_HOME
PATH=$PATH:$JAVA_HOME/bin
export PATH JAVA_HOME PENTAHO_JAVA_HOME
```

By pointing the `JAVA_HOME` variable at a symbolic link, we can change the JDK simply by redirecting the link. The `PENTAHO_JAVA_HOME` variable allows you to use a different Java installation for Pentaho. Normally this is not necessary, so here we have set it to be the same as `JAVA_HOME`.

Any new terminals you open will have these settings. To test:

```
$ echo $PENTAHO_JAVA_HOME
/home/pentaho/jdk64
```

If you're not on a Linux or Mac OS X system (e.g. Solaris) you may need to use a different file to store these values, such as `~/.profile`.

Package manager (RPM)

For an installation that integrates better with the rest of the system, use the operating system's package manager. Java will be usable by all users. This requires root permissions.

These instructions work on Red Hat based systems (including CentOS and Fedora).

In the steps below, replace the JDK version numbers in the file/directory names to match what you're using. Also pay attention to the architecture.

1. Extract the downloaded bundle and accept the licence

```
# sh jdk-1_5_0_15-linux-amd64-rpm.bin
```

2. Install the RPM (extracting the bundle as root should do this for you)

```
# yum localinstall --nogpgcheck jdk-1_5_0_15-linux-amd64.rpm
```

3. Register this copy of Java as an alternative (alter the path according to your installation)

```
# /usr/sbin/alternatives --install /usr/bin/java java /usr/java/jdk1.5.0_15/bin/java 100
# /usr/sbin/alternatives --install /usr/bin/javac javac /usr/java/jdk1.5.0_15/bin/javac 100
# /usr/sbin/alternatives --install /usr/bin/jar jar /usr/java/jdk1.5.0_15/bin/jar 100
```

4. Set this Java to be the default (you will be prompted after you run each command)

```
# /usr/sbin/alternatives --config java
# /usr/sbin/alternatives --config javac
# /usr/sbin/alternatives --config jar
```

5. Check that this Java is now [default](#) :

```
# java -version
```

6. Set the `JAVA_HOME` environment variable in your `/etc/profile` (system-wide) or `~/.profile` (per user)

```
JAVA_HOME="/usr/java/jdk1.5.0_15"
export JAVA_HOME
```

(alter the path according to your installation)

7. Logout and log back in, then test the variable:

```
$ echo $JAVA_HOME
/usr/java/jdk1.5.0_15
```

Install the Web browser plug-in

This is not required for Pentaho operation, but can be handy on a desktop system for other things. Generally, the OpenJDK/IcedTea plug-in is fine (and preferred, since it's open source), but some applets work better with the Sun JDK. On Firefox, you can check to see what you're using by typing *about:plugins* into the location bar.

Pentaho Tutorial - Install and Configure Java

Installation is just a simple case of creating a symlink from the Java installation into the Mozilla plugins directory. In the commands below, adjust the file paths as appropriate.

The command for an [x86_32 system](#) is:

```
# ln -s /usr/java/jdk1.5.0_15/jre/plugin/i386/ns7/libjavaplugin_oji.so /usr/lib/mozilla/plugins/libjavaplugin_oji.so
```

On an x86_64 system, you should use Sun JDK 1.6 Update 12 or higher to get a Web browser plug-in. Then run:

```
# ln -s /usr/java/default/jre/lib/amd64/libnpjp2.so /usr/lib64/mozilla/plugins
```

Restart Firefox and check *about:plugins*. Enable the plug-in: *Edit # Preferences # Content* tab # tick *Enable Java*

You can test your plug-in at java.com and javatester.org .