# Getting the software

The previous chapter showed you all of the wonderful things that we can do with Drools. But we will not get very far if we don't install the software first. So, in this chapter we will see how to install the software.

**What are we going to install?**

We will be installing four pieces of software. All of these are open source (that is, free), can be downloaded easily from the Internet, and are available under a business-friendly license.

* Java: This is the core computer language upon which all of the other tools are built.
* BRMS/Guvnor and JBoss App Server: This is a web-based rules editor aimed at business users. We install JBoss App Server as the easiest place to run this editor.
* Maven, a build tool that takes the various Java scripts (source) and transforms them into a package that we can deploy on a web server. Using Maven makes our examples easier to understand, as Maven automatically downloads all the other software required.
* Eclipse and the Drools plug-in: Eclipse allows us to edit the Java files that we will use for transporting information around the system. The Drools plug-in gives us a more technical editor for rules, and the ability to see what is going on inside the rules engine.
* Drools examples for this book— hands-on samples so that you get to know Drools inside out.

**Who should install it?**

Broadly speaking, there are two types of people reading this book.

* Technical people, who are already familiar with Java, but who want to understand business rule technology
* Business people who have the domain knowledge, but to whom the technology (and Java) may be new territory

Although the setup guide here is suitable for both groups, don't be afraid to ask for help. The reason why all of the setup instructions are here in one place is that you can ask your nice, friendly, technical support person to 'set up everything in Chapter 2', and then return to Chapter 3 knowing that everything is in place. Indeed, many companies have their desktops locked so that regardless of your knowledge, you're going to have to request this technical assistance.

Why are the instructions in this guide only for Windows? What about the Mac and Linux users?

As a complement to the Linux users, we'll assume that you know enough about a computer to translate the instructions for your platform. Mac users are in a trickier position. I am sorry that we had to concentrate on the most popular platform. The software here will work on the Mac (it is Java, after all), but you may need to follow up the links at the end of the chapter to get Mac-specific instructions.

If you're technically adept, allow about one to two hours to install all of the software, assuming you have a fast Internet connection so that you're not waiting too long for downloads. Many of these instructions may be obvious (or you've done them before). So feel free to plough on, but quickly check through to ensure that you've got things set up correctly.

## Installing Java

Java is the computer language in which Drools and all other products used in this book are written. So it's pretty important that we install it. Fortunately, it's an open source product from Sun that will run on almost every computer platform.

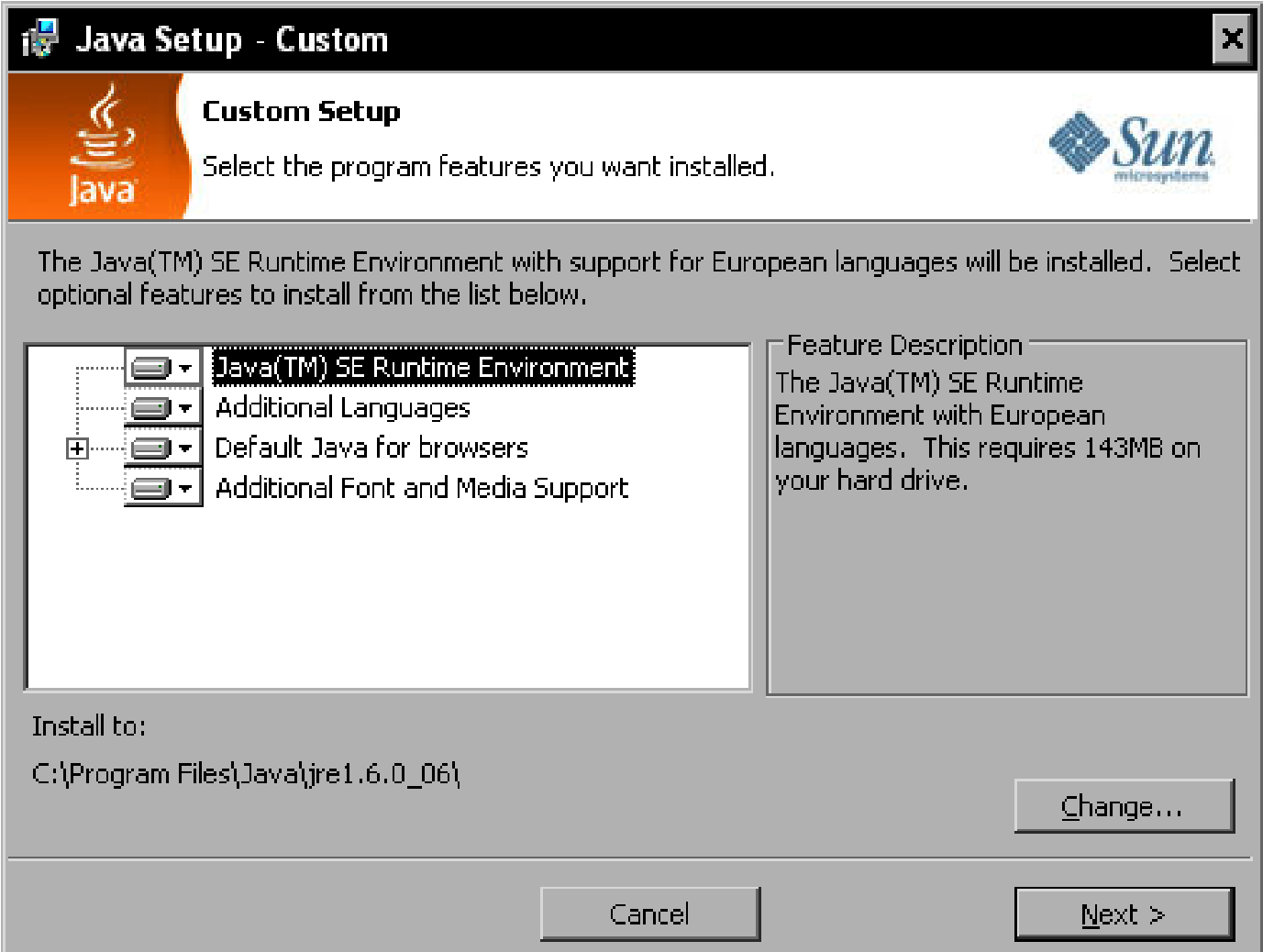
It's important to note that there are different versions of Java, such as:

* The **JRE** (**Java** **Runtime** **Engine**) is intended for end users. The **JDK** (**Java** **Development** **Kit**) contains this runtime, plus tools for the people developing using Java (that's us!).
* **Standard** **Edition** (**SE**) is what we'll be using. There is also an **Enterprise** **Edition** (**EE**) that takes the SE and adds a few more powerful services. While it is likely that your business rules will be deployed in an enterprise system, the SE is enough for the topics that we will cover in this book.
* Some versions of Java come bundled with the Netbeans IDE (a Java editor). This is optional, as we use the Eclipse Java editor instead (because we can get a Drools plug-ins for Eclipse, but not Netbeans).

When downloading Java, remember that the version we use is the JDK (developers' edition) of Java SE. To start, go to the web site **http://java.sun.com/javase/ downloads/index.jsp**.

Download the latest version (at least 6 or 6.1) of Java, selecting the correct language and operating system for your computer. On the next screen, select the download method that you want to use. If you're not sure, click the link **Windows** **Offline Installation** and save the file in a place that you'll remember.

Once the download is complete, open this file. Click on **Run**, and then **Accept** the license agreement, (but only if you don't intend to export it to North Korea, as per the licence!). Unless you've specific reasons for doing otherwise, accept the defaults of features and location by clicking **Next**. But be sure to make a note of the **Install to** folder first (for example **C:\Program Files\Java\jre1.6.0\_06\**).



The install should chug away for a couple of minutes. You'll also be asked to install the runtime. Again, unless you've any specific reasons for doing otherwise, accept the default features and locations by clicking **Next**.

Let the install chug away for another couple of minutes. Amuse yourself by looking at the advertisement for OpenOffice that appears. (By the way, OO is a very good, and free, replacement for Microsoft Word, and is being used to write this book. So if you have (ahem) a less-than-legal copy of Word, consider using OpenOffice as a 'drop in' replacement/upgrade available from **www.OpenOffice.org**.)

After another couple of minutes, you should see the 'Install complete' screen. Click on **Finish**.

The installer may open a web browser asking you to register. This is an optional step. Feel free to provide your personal details to Sun if you wish. It's a reputable company, but I prefer to keep my private details, well, private.

Congratulations, you now have the Java development tools installed!

## Installing JBoss

The BRMS/Guvnor is a web-based business rules editor. So we need a web server to install it on. If you don't happen to have a web server that can run Java to hand (although many companies do), then it's easy enough to install one. We're using JBoss App server as it has the fewest steps to get the BRMS up and running.

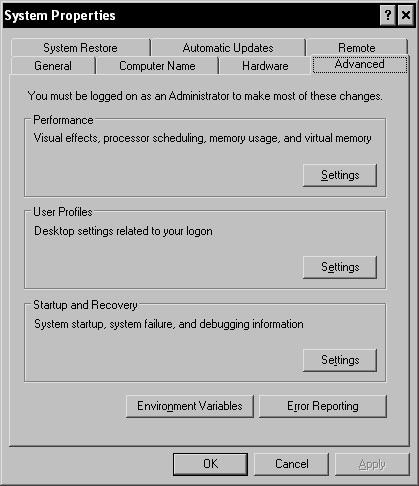
Even though we're installing JBoss here, it is possible to run the BRMS/ Guvnor on other Java-based App/Web servers such as Websphere, Weblogic, Tomcat, or Oracle Application server. See the wiki on **www. jboss.org/drools** for more details.

Before we start the process, we need to tell JBoss where to find the version of Java that we just installed. We do that by carrying out the following steps:

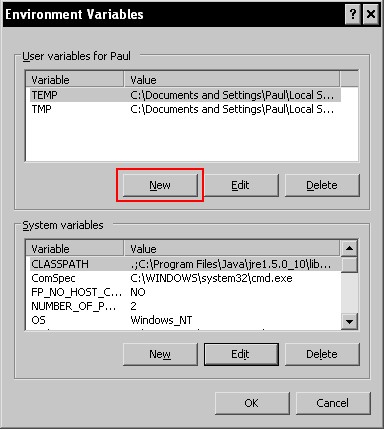
1. First, right-click on the **My Computer** icon on your computer desktop.



1. Then, from the pop-up menu, select **Properties,** and the following screen should appear. (A quicker way to carry out these two steps is to press the *Windows* + *Pause* keys at the same time.)

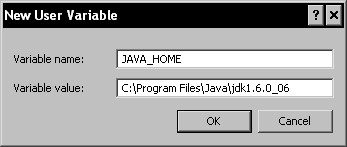


1. Click on the **Advanced** tab in the window that appears.
2. Then click on the **Environment Variables** button.



1. Click on the **New** button in this window. In the **New User Variable** window, enter the variable name **JAVA\_HOME** and the location at which you installed Java on the previous step. On my machine this is

**C:\Program Files\Java\jdk1.6.0\_06**, but it may be different on yours.



1. Click on **OK** (multiple times) to close the windows that have been opened during the preceding steps.

### Actual install

Now you're ready to download JBoss from **http://www.jboss.org/jbossas/ downloads/**. Click on the download link on this page (take the latest stable version). At the time of writing it's 4.2.2.GA, although 5 should be stable by the time you read this. You'll be taken to the SourceForge download page. Select the filename ending in **.zip** (and no other letters). Your download should begin. Save this file in a place that you'll remember.

At the time of writing, there is an issue between JBoss 5 and the (in progress) Guvnor. While this is likely to be resolved by the time you read this, if you do encounter any problems (Error setting attribute SecurityManagement) please try the 4.2.3 version

When the download has finished, unzip (that is, extract) the files to a folder of your choice. By default, I use **c:\software\JBoss**. It might be helpful to follow this convention on your machine so that all path names given in the book will be exactly that same as on your PC.

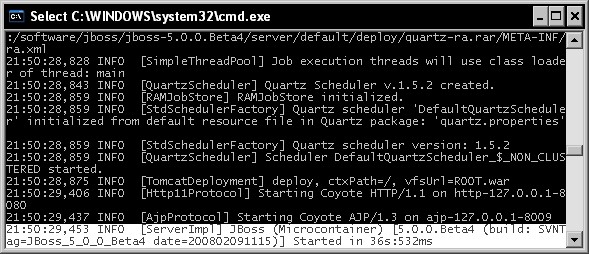
If you don't already have a ZIP program (such as WinZip) installed on your machine (that is, if you double-click on the ZIP file and nothing happens, or Windows asks you which program you want to use), then install a ZIP utility. (There are several. I tend to use the open source 7-zip utility, which is available from **http://www.7-zip.org/**.)

That's it! We now have the JBoss App Server installed (which was painless!). Now, to run it, open the folder we just created (using Windows Explorer). Double-click on the bin folder and you'll see a set of files. To start JBoss, click on **run.bat**.

After a couple of seconds, you'll see a new window with white text on a black background. Look for the words **Starting Jboss (Microcontainer)**, which indicate that JBoss has found Java on your machine.

You may get a security question from Windows (or whichever firewall software that you use). This is normal, so click on **OK** or **Unblock**.

After a few seconds churn away, the text whizzing past should stop and you should get a message that JBoss has started successfully (this message has been highlighted for emphasis in the following screenshot).



As a final step to confirm that everything is working OK, open the following address in your web browser:

**http://localhost:8080**



Congratulations, you now have the JBoss Web/App Server running on your PC!

## Installing the BRMS/Guvnor

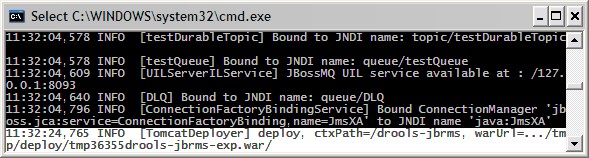
Open **http://www.jboss.org/drools/downloads.html** in your web browser and look for the 'Drools BRMS' download section). This may have been renamed to Guvnor/version 5 by the time that you read this. In general, take the latest available version.

Download this file to a place that you will remember. Once the download has completed, extract (unzip) these files to a temporary folder—it should contain at least one file, with the name **drools-jbrms.war** or **guvnor.war**.

Copy this **.war** file to the JBoss deploy directory. If you followed the same directory name as I did when installing JBoss, this should be **C:\software\jboss\jboss5.0.0.Beta4\server\default\deploy** (that is, [wherever-you-installed-jboss]\ server\default\deploy).

Make sure that JBoss is running. (If it isn't, start it as per the previous step by clicking on **run.bat**.)

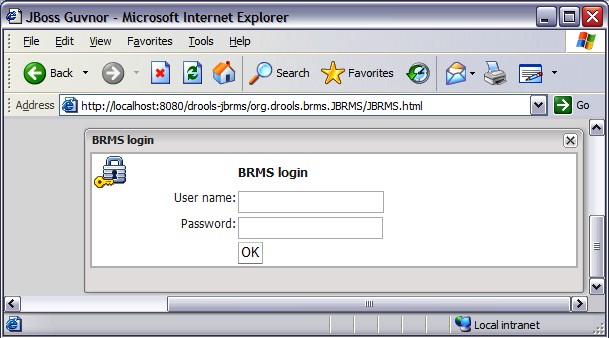
You should see the following line (highlighted) appear in the console telling you that the application has been successfully deployed.



To check this, open Internet Explorer (or your browser of choice) and go to the web page **http://localhost:8080/drools-jbrms/**.

This web link will change depending on the name of the war file you copied. If the name changes to **guvnor.war**, then the web page that you need is **http://localhost:8080/guvnor/**.

After a couple of seconds you should see more activity in the console (make sure that you have no text highlighted there, as this will block JBoss). Eventually, you should see the **BRMS login** screen in the browser, which look somewhat similar to the following screenshot:



Congratulations, you now have the BRMS/Guvnor successfully installed! If you're tempted, just click on **OK** (using a blank **User name** and **Password**) to log in and have a look around.

By default, BRMS/Guvnor doesn't have security enabled. However, it's easy to switch it on and use the same password security that you use to log into Windows (via LDAP). See the Drools documentation for more details.

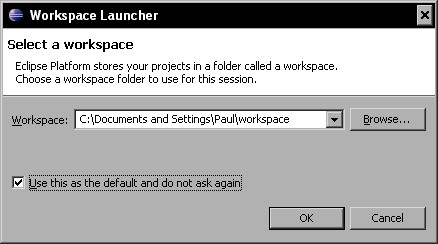
## Installing Eclipse

We use Eclipse to edit the Java files that transport data to and from Drools. It's also the basis for the Drools plug-ins, including the advanced rule editor. To get started, download Eclipse from **http://www.eclipse.org/downloads/.**The version that we want is the Eclipse IDE for Java EE developers, as this version pre-packages a lot of additional tools and features.

On the following screen, select your nearest mirror, and the Eclipse download should start. Save the downloaded file in a place that you will remember (probably the same place where you downloaded Java and JBoss). When the download is complete, unzip the file to a folder of your choice. (I use **c:\software\eclipse**.)

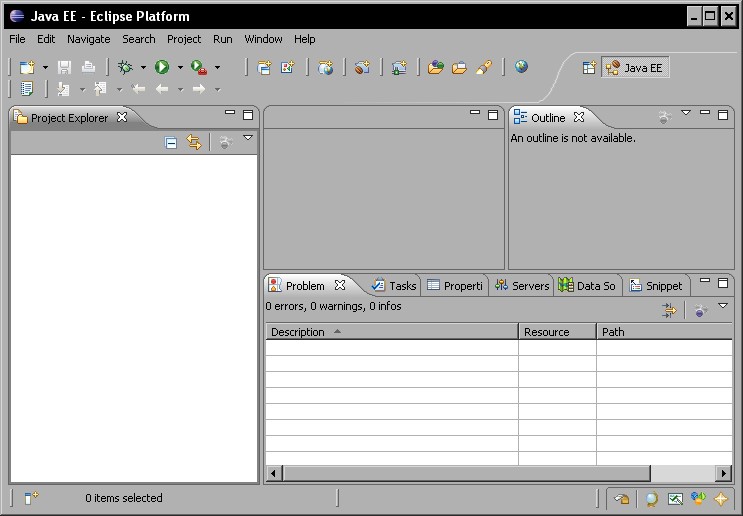
Opening this folder in Eclipse shows a set of files including **eclipse.exe**. This is the Eclipse IDE file. Congratulations, you now have Eclipse installed! (This was too easy.)

Clicking on the **eclipse.exe** starts Eclipse. You should see the splash screen, and then a request for where Eclipse should save its internal files. Normally I just accept the default, making sure that the checkbox is selected (on the bottom left) so that I am not asked the question again.



After that, a 'quick start screen' will appear with several useful links (feel free to click around). When you are ready to go to the workbench, click on the curved arrow icon on the far right of the screen.

A blank Eclipse workspace should be displayed, as shown in the following screenshot:



That's it. Not only have you installed Eclipse, but you have it up and running.

## Installing the Drools plug-in

Eclipse is not just a Java editor, but also a platform. This means that we can extend it with any tool that we require. In this case, we're going to add the Drools plug-ins, making it easier to edit and debug business rules.

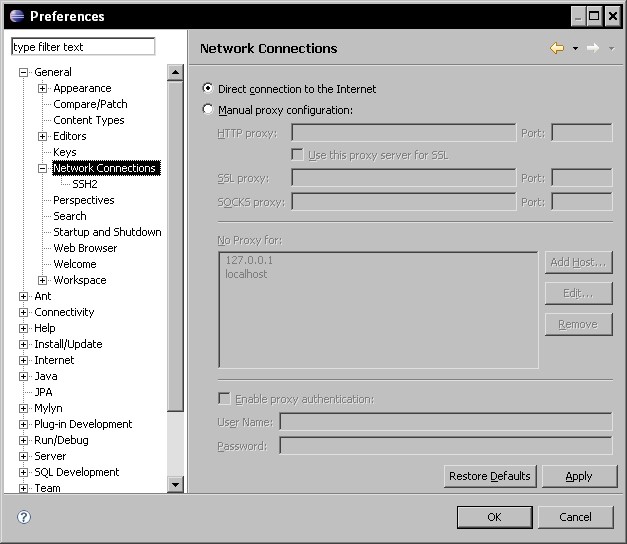
The easiest way to install a plug-in is via the Eclipse update manager. This can find plug-ins on the Internet and then download them, so it will need to know your connection details. If you have a direct connection to the Internet (dial-up or broadband) you won't need to change these.

However, in most corporate situations you will need to tweak the Eclipse settings. The good news is that the Internet connection details you need will probably be exactly the same as the connection details in Internet Explorer (and other web browsers).

To find your Internet connection details, open Internet Explorer. From the toolbar at the top of the screen, select menu option **Tools | Internet Options**. In the dialog box that appears, click on the **Connections** tab and then on the **LAN Settings**. Make a note of the details that appear in the pop-up box.

To copy these settings to Eclipse, open the Eclipse IDE. From the Eclipse toolbar

(which is at the top of the screen), select menu option **Preferences** | **General |** **Network Connections**. You should see a window similar to the following, in which you can enter in your connection details:



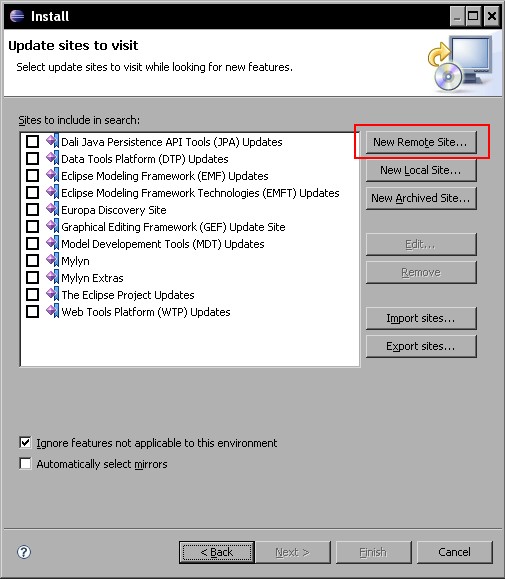
If you are unsure of what to enter here (or if you need to change anything in the first place) ask your colleagues—the answer will vary from organization to organization.

### Finding the plug-in

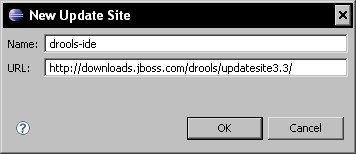
To find out the latest Drools Eclipse update site, open Internet Explorer and go to **http://www.jboss.org/drools/downloads.html**. Look for the text 'Eclipse Workbench update site'—the update site link will be shown next to this. You'll probably want the most recent version (unless you specifically downloaded an older version of Eclipse).

Right-click on the **Update** site link and go **Copy shortcut**—the text will be similar to **http://downloads.jboss.com/drools/updatesite3.3/**. We'll need this address in a minute.

Back in Eclipse, open the update site wizard from the main Eclipse toolbar (via menu option **Help | Software Updates | Find | Install**). On the screen that appears, select **Search for new features to install,** and then click on **Next**. Now we're shown a list of already-installed features. As we want to add a new one, we click on **New Remote Site**.



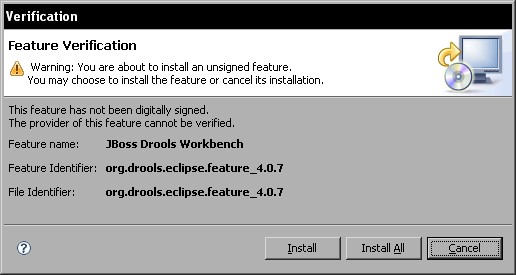
In the pop up dialog box, give the new remote site a name (for example **drools-ide**), and the URL of the update site that we searched for and copied earlier.



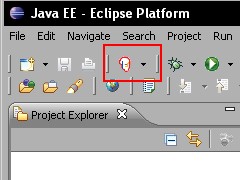
Click on **OK** to return to the previous **Update Sites** dialog box (the **New Update Site** should now have been added to the list, with a tick mark against it) and then click on **Finish**. Eclipse should then contact the update site to see which (new) plug-ins are available. A new screen will appear showing the plug-ins that Eclipse has found. Make sure that the checkbox next to the Drools IDE is selected, and then click on **Next**.

On the next screen, accept the terms in the license agreement (by selecting the checkbox), and then click on **Next**.

Unless you've any particular reason for doing otherwise, accept the default install directory and click on **Finish**. Eclipse should now take several minutes to download the Drools IDE software. After all of the features have been downloaded, you'll get a message displaying the jars that have not been digitally signed. This is OK (most Eclipse plug-ins don't have signatures). Click on **Install All** to proceed with the installation.



Everything going well, you should (after a moment or two) get the **Restart Eclipse** message. Click on **Yes** to complete the installation. You'll know that the install of the Drools tools went smoothly if you can see the Drools icon toolbar as part of your Eclipse screen when Eclipse re-opens.



Congratulations, you now have the Drools IDE running on your PC!

## Installing Maven

Maven is a Java Build system from Apache (the same people who built the popular web server). It takes Java source files and converts them into a format that we actually deploy and use. We use Maven for two things: to build our samples (makes the samples much easier to download and use), and later to build some of our own Java code. More Maven documentation is available from the Maven site at **http://maven.apache.org/**.

To get Maven, download it from **http://maven.apache.org/download.html**.

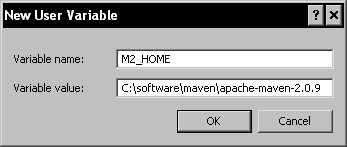
Select the ZIP file (all versions are the same, but compacted for download in different ways). It's normally best to take the latest stable version. You'll then be asked to select a mirror. Pick the one nearest to you; it should normally be OK.

Save the file to an easily-remembered place, and then unzip the file to a folder of your choice. Following the same convention as before, I use a folder such as **C:\software\maven\**.

Open the folder that you just created in Windows Explorer, and there should be one more folder inside it (named **apache-maven-x.x-.x**). Copy the folder name in the address bar. In this case it's **C:\software\maven\apache-maven-2.0.9**, but it is likely to be different on your computer. We'll use this address to tell Maven which directory it is located in.

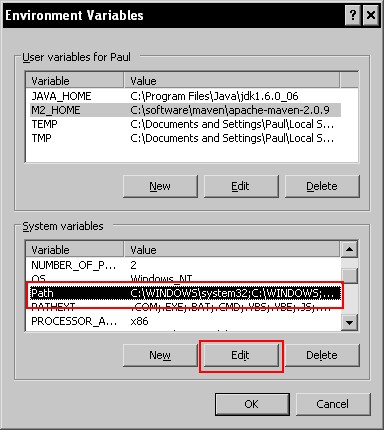
If you can’t see the address bar, select menu option View | Toolbars | Address Bar from the very top of the screen in Windows Explorer.

Open the environment variables as we did for setting the **JAVA\_HOME** earlier (right-click on **My Computer** andgo to **Properties | Advanced | Environment variables**). This time, click on the **New** button on the screen to create a new environment variable.



Add an entry for **Variable name** (**M2\_HOME**), and a **variable value** of the folder (for example, **C:\software\maven\apache-maven-2.0.9** that we copied earlier. This value could be different on your machine; make sure that you use the correct path!). Make sure there is no trailing '\' character.

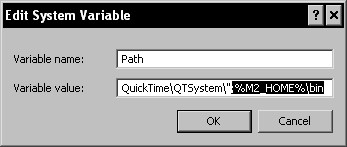
Click **OK**, and then highlight the **Path** entry in the **Environment Variables** dialog box (this is the dialog box that you returned to when you clicked **OK**). Now click on **Edit**, as shown in the following screenshot:



Add the value to end of the variable value as shown. This tells Windows where to find Maven, no matter where we try to start it from.

**;%M2\_HOME%\bin**

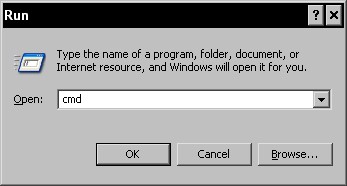
Warning! Paths can be temperamental at times, so play around with this setting a bit if you have any problems with installing Maven..



Click on **OK** a number of times to close all of the open dialog boxes.

Now, to check that we have installed Maven correctly, open a command window (you may remember this as being called the DOS prompt). We'll be doing it a couple of times, so it's worth remembering how to do it.

1. In Windows, press *Windows + R* and a dialog box similar to one shown below should appear. (The windows key is the one with the windows logo on it, often found on the bottom left of your keyboard next to the *Alt* key.
2. In this box, type **cmd** then click **OK.**

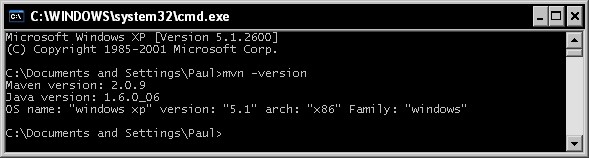


1. This will open a command window with white text on a black background. If you're older than 30, you might remember all computers as looking like this!

Another way to open a command window is to find it from the Windows start menu. This will vary depending on your version of Windows, but in Windows XP, it is found under, **Start | All Programs | Accessories| Command Prompt**. Just look for the following icon on the menu.



1. In the command window, type:  **mvn -version**
2. Then press *Enter*. If your Maven installation has been successful you should see something like the following screenshot:



If you need to change the details of your Internet connection in Eclipse, then you will need to do something similar for Maven too. This allows Maven to automatically find and download all the libraries required. The details of your Internet connection will be same as the ones we used before. To pass these details to Maven, carry out the following steps:

1. Find the Maven configuration file (which is named **settings.xml**) and open it in an editor such as Microsoft Notepad.
2. Find the section beginning with **<proxies>** and edit it so that it is similar to the following example. Of course, the values will be different for your system.

**<proxies>**

**<proxy>**

**<id>optional</id>**

**<active>true</active>**

**<protocol>http</protocol>**

**<username>user-name-if-required-or-delete-line</username>**

**<password>password-if-required-or-delete-line</password>**

**<host>url-of-proxy-host</host>**

**<port>80</port>**

**</proxy>**

**</proxies>**

Note that we've removed the lines beginning with **<!--** , **|**, and **-->** as these are comments. We've also deleted the line beginning with **<nonProxyHost>**. For more information on configuring proxies in Maven, refer to the guide found at **http://maven.apache.org/guides/mini/guide-proxies.html**.

## Installing sample projects for this book

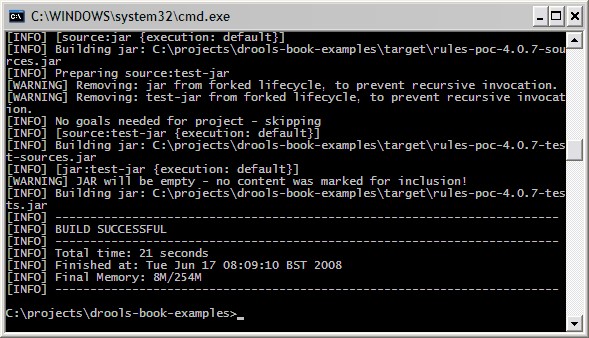
All of the samples in this book follow the same format. So, a good time to download one of the samples is when you have technical people around. Our first downloadable example is from Chapter 6, which can be downloaded from the sample site at **http://code.google.com/p/red-piranha/**. Unzip the file to **C:\ projects\drools-book-examples**. Open a command window (using the *Windows +R* key as we used earlier). Move to the directory that we just created by typing the following line:

**cd C:\projects\drools-book-examples**

Now we can use Maven to build the samples project by typing the following command and pressing the *Enter* key:

**mvn clean package**

Maven will automatically download all of the required software and libraries. This can take a couple of minutes. If the download is successful, you should see output similar to the following:



Congratulations! You have now successfully downloaded the samples for this book, and all of the required software. Now we're going to set up the samples in Eclipse to make it easier to look around them.

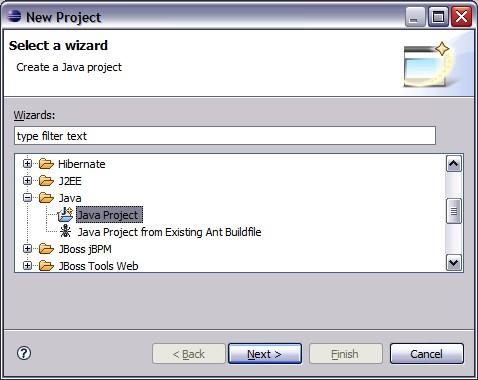
### Setting up the sample project in Eclipse

Maven can automatically set up the Eclipse project for us. In the same command window as the one shown earlier, type the command:

**mvn eclipse:clean eclipse:eclipse**

This is much quicker and should only take a couple of seconds to run.

Now open Eclipse (using the previous steps, double-click on **eclipse.exe**). When Eclipse opens (and you've selected the workspace), select **File | New Project** from the menu (on the top left) and the following dialog box will appear:

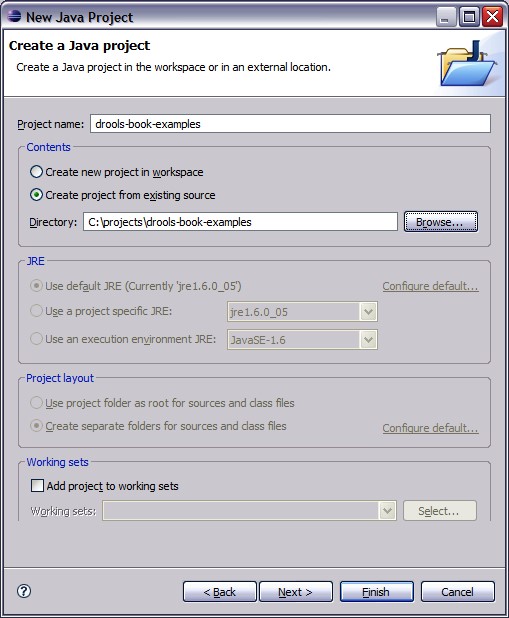


Select **Java Project** under the **Java** folder, and then click on **Next**. In the dialog box that appears next:

* Give the project a name (I've used **drools-book-examples**, as I find it easier to keep the project name the same as the folder name)
* Uncheck the **use default location** checkbox
* For the location, enter the folder to which we unzipped the sample files

(for example, **C:\projects\drools-book-examples**)

* Leave all of the other values as they are
* Click on **Next**



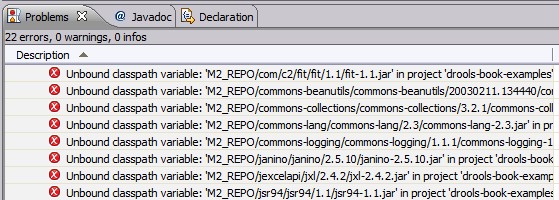
Eclipse will find the project files that Maven created for us at this location, so all our libraries and source paths are already set up for us on the next screen.

We can safely click on **Finish**. Eclipse will ask if you wish to switch to the Java Perspective. Click on **Yes**.

If you see a screen without a red cross next to the project name (unlike the following screenshot) then rejoice, as this means that the samples for this book are set up successfully and you're ready to go to Chapter 3.

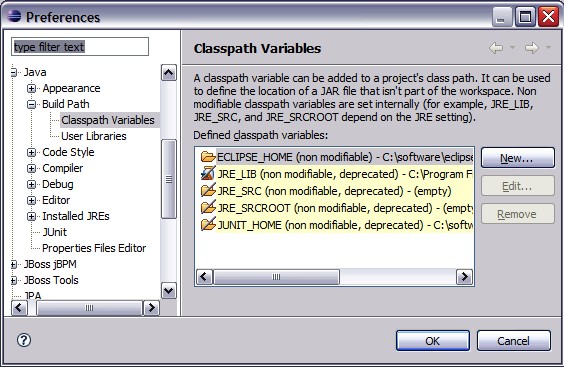
### Getting Maven and Eclipse to work together

If you see a red cross next to the project name or if you see the following errors in the **Problem** tab, then you need to tell Eclipse where Maven stores its files.

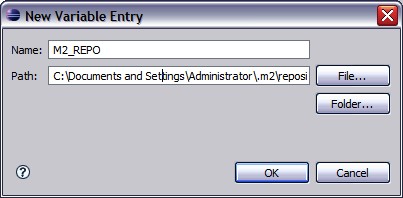


In Eclipse, select **Windows | Preferences** from the toolbar at the top of the screen.

In the dialog box that appears, select menu option **Java | BuildPath | Classpath** **Variables.** You will see a screen similar to the following:



Click on the **New** tab on the righthand side of the dialog box. In the dialog box that appears next, enter the **Name** (**M2\_REPO**) and the **Path** where Maven stores its files.



On Windows XP machines, this is likely to be in the format: **C:\Documents and Settings\Administrator\.m2\repository** Replace **Administrator** with your username.

On Windows Vista machines, this is likely to be in the format:

**C:\Users\Administrator\.m2\repository**

Again, replace **Administrator** with your username.

If there is any doubt, open Windows Explorer and try to navigate to the files. Alternatively, use the search facility in Windows Explorer to find the **.m2** folder on your machine. It's best to search all of the files under the C drive. This takes longer, but there's a better chance of finding the file.



Back in Eclipse, when you've finished entering the variable name and value, click on **OK**. You will be asked if you want to do a full build. Click on **Yes**.

After a few seconds, the red crosses should disappear and you should see the project set up as per the screenshot in the previous step.

## Troubleshooting

Here is a list of things for you to check if anything goes wrong:

1. Ensure that you have the developer version of Java installed, and not just the runtime.
2. Check the proxy settings for Eclipse and Maven. If your download stops or hangs half way through, try running the command again.
3. Check that you don't have two copies of JBoss or Eclipse running at the same time.
4. Check the versions of JBoss and BRMS/Guvnor that you have installed. If BRMS/Guvnor is not working, try dropping back to an older version (of both JBoss/BRMS) and repeat the steps again.
5. If you have any problems with setting up the Eclipse project for the samples (or even if you're seeing strange Eclipse errors later), remember that Maven is the master build, and Eclipse (in this case) is just a glorified text editor. So try these steps; Close Eclipse, Delete the **.classpath** and **.eclipse** files in the root of the folder and the .**settings** folder, then run the Run the command mvn eclipse:eclipse again. Then follow the instructions to setup the eclipse project again
6. If you get a specific error message, try searching that term in Google.
7. Double-check the instructions on the Java, JBoss, Maven, or Eclipse web site (as appropriate).
8. If everything else fails, read *How to ask for help* from the previous chapter.

## Summary

Whew! We have covered a lot. If you've just re-joined us (because somebody has kindly set up the technical items for you), then this is what we did:

1. Set up Java.
2. Set up BRMS/Guvnor running on the JBoss App Server.
3. Set up Eclipse and installed the Drools plug-in.
4. Installed the Drools examples for this book and the Maven to build them.

We'll be using all of these tools in the next couple of chapters, starting with the BRMS/Guvnor to edit our business rules.