

# Configuring PAMGUARD and the Eclipse IDE

Doug Gillespie. April, 2008

## 1 Introduction

Absolutely everything you need can be downloaded for free from the Internet. I've given the names of the latest files available on 26 April 2006. With time, you may expect some of these to be updated with newer versions. I've also taken file files suitable for installation on a 32 bit version of Windows XP. If you're working on a Mac or Linux system, then you may need to download different files.

You are of course under no obligation to use the Eclipse IDE – it's what most of the PAMGUARD core development team have been using, but if you have a different way that you like to do things, then go ahead. I assume that you know what you're doing and will take what you need from the instructions below (such as the JAMA and RXTX library information) and ignore the rest.

You will need:

1. Eclipse
2. Java software development kit (JDK) version 1.6 or later
3. The Rxtx plug in (for serial port communications)
4. The Jama plug in (for matrix calculations)

You may also want

1. The One-Jar plugin which enables you to create a single jar file for execution without Eclipse

These instructions are for installing on Windows XP, they will be very similar for other platforms.

## 2 What you must do to get started on a clean machine:

1. Install the JAVA jdk. Generally it's worth taking the latest one from <http://java.sun.com/javase/downloads/index.jsp>. The file you're looking for will probably have a name something like [jdk-6u6-windows-i586-p.exe](#).
2. Install Eclipse – this is simply a question of downloading and unzipping Eclipse into a suitable directory on your computer (such as program files/Eclipse and making a shortcut to Eclipse.exe on your desktop or some other convenient location.
3. Download the rxtx library from <http://users.frii.com/jarvi/rxtx/>. The file you're looking for is rxtx-2.1-7-bins-r2.zip. There are two types of files in there that you need, a .jar file and two .dll files. Copy the .jar file RXTXComm.jar into the \lib\ext folder for wherever you've installed the Java run time environment (this should have installed along with the JDK). On my machine, this is currently C:\Program Files\Java\jdk1.6.0\_03\jre\lib\ext. Then copy the two files rxtxSerial.dll and

rxtxParallel.dll into the jre\bin folder (on my machine, this is C:\Program Files\Java\jdk1.6.0\_03\jre\bin).

4. Download the Jama matrix library Jama-1.0.2.jar from <http://math.nist.gov/javanumerics/jama/> and put it in the same location as RXTXComm.jar.

Those four steps will have configured Java, the Eclipse IDE and a couple of library files the PAMGUARD needs. If you think you'll want to create your own executable jar files for PAMGUARD then you should also install the one-jar Eclipse plug in from <http://one-jar.sourceforge.net/>. Download the file one-jar-sdk-0.96.jar and execute as described on the one-jar sourceforge site. i.e. open a command prompt and type 'java -jar one-jar-sdk-0.96.jar', then follow the instructions.

### 3 Get the PAMGUARD source code

The PAMGUARD source code is all on a CVS server at sourceforge. Again, these instructions assume that you are using Eclipse to access it.

#### 3.1 *If you have not been given the source code and need to start from scratch*

Make a new folder on your hard drive, Call it what you like, but I'll assume it's called c:\PamguardCode

Start Eclipse and you should get a window asking you to select a workspace. Browse to the folder you just created. Eclipse should then start, probably showing a blue screen with a number of buttons you can press to learn more about Eclipse. From the file menu, select new/project. Click on cvs and select 'Projects from CVS' and click

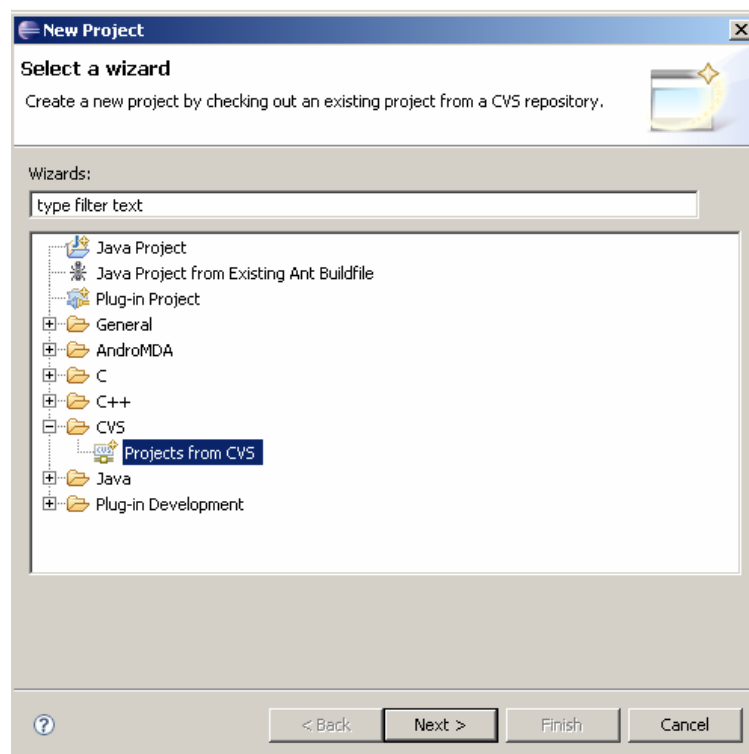
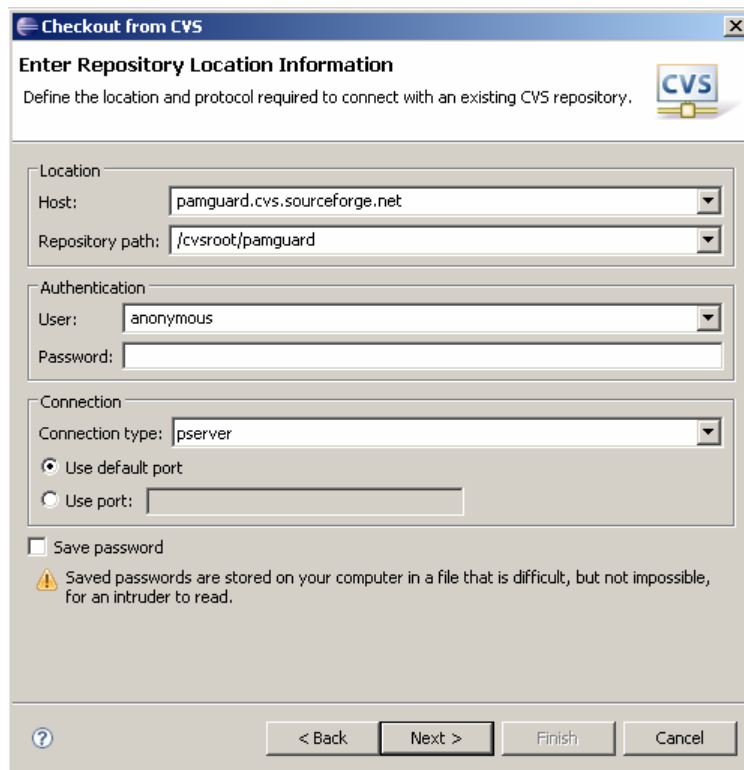


Figure 1. Creating a new Eclipse project

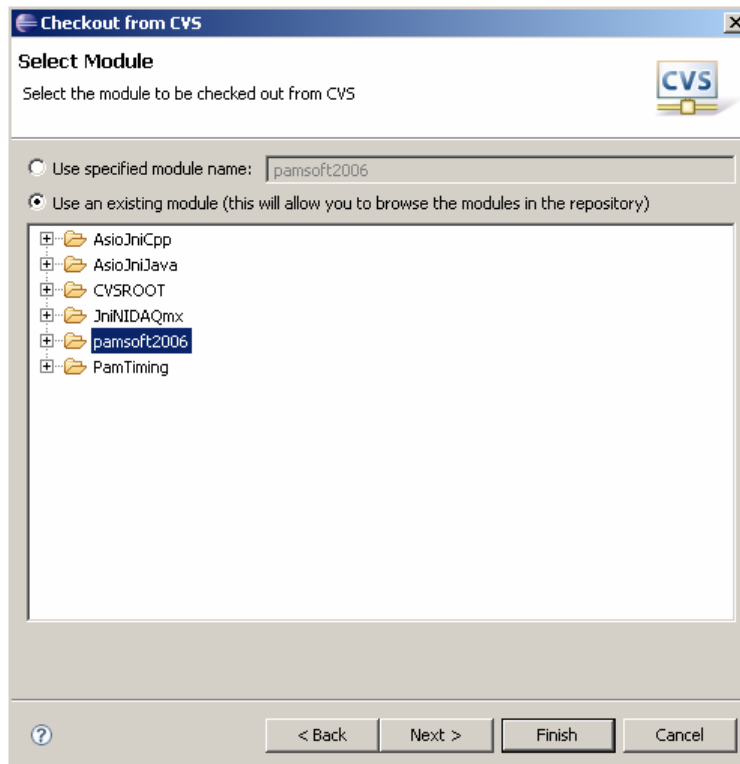


**Figure 3 Accessing the CVS Repository**

Next

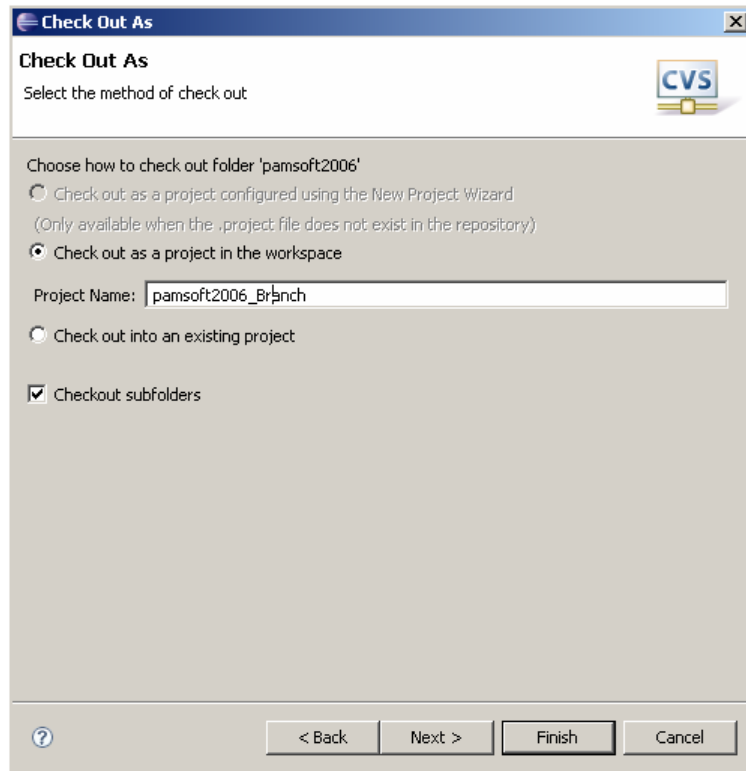
If you don't have a password and just want read access to the code, then enter the following information (copy and paste it from here)

Host: panguard.cvs.sourceforge.net

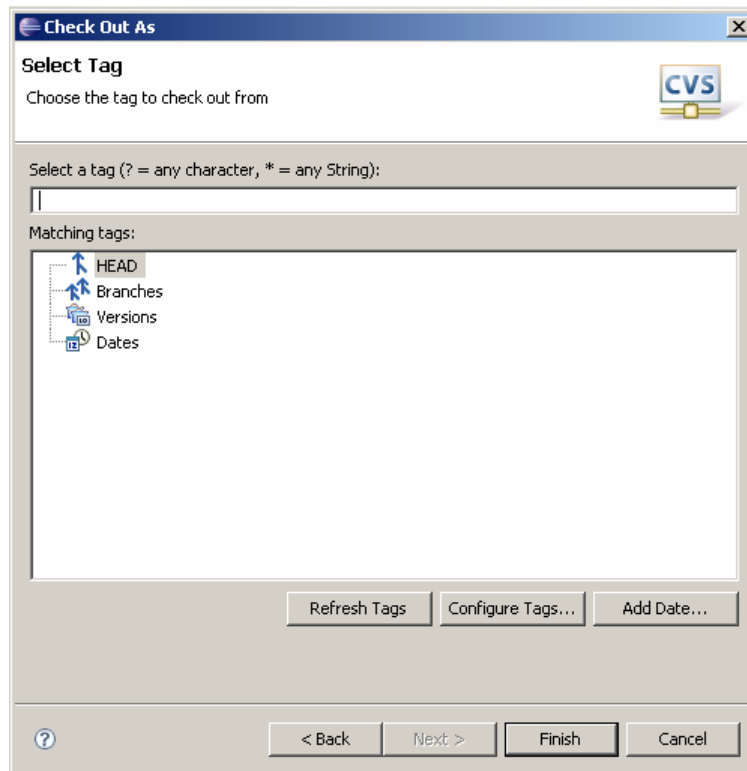


**Figure 2. Selecting the PAMGUARD Java source code**

Repository path: /cvsroot/panguard



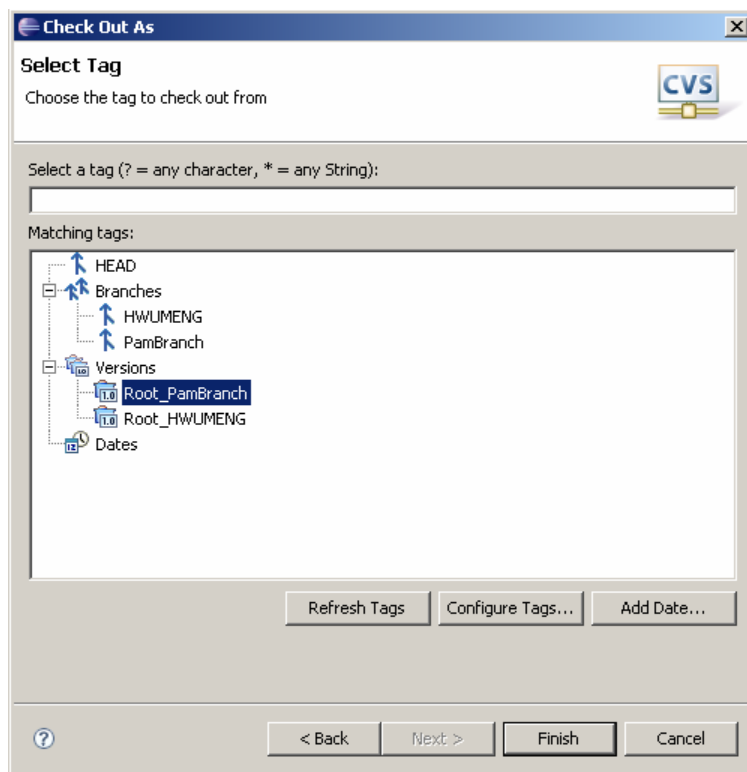
**Figure 4** Set the project name as you want it to appear on your machine.



**Figure 5** Selecting the correct branch

User: anonymous  
 Connection type pserver

If you have a user name and password and wish to connect as a developer, it's pretty much the same, but set the Connection type to 'extsh'.



**Figure 6** Selecting the correct branch

code for these and the library files associated with each of the C projects is included with pamssoft2006.

Select Next to get the following:

You can probably select the default name as shown and hit Next again. If you intend to set up CVS links to more than one version however, it might be worth changing the name at this point to 'Pamguard Head' or 'Pamguard Branch' as shown above.

Hit Next a couple more times to get the 'tags' screen

Press the Refresh Tags button

Expand the 'versions'

And select Root\_PamBranch for the latest development code. If you want the latest released version and something a bit more stable, select HEAD.

Click Finish and wait while the source code is uploaded from the server. This may take several minutes depending on the speed of your internet connection.

Once the code has finished downloading, select Window/Open Perspective/Java Browsing.

### 3.1.1 Errors

Scroll down through the packages. It is possible that some will have a little red cross beside them. This is probably because they can't find either the rtxcomm or the Jama jar files. This is probably because you have multiple JRE's on your computer and Eclipse is trying to use one that doesn't have rtxcomm.jar and jama.jar in it's lib/ext folder. There are two ways of solving this problem.

1. The scatter gun approach, which is to put those jar files in every lib/ext folder you can find.

Click 'Next' and select the radio button 'Use and existing modules ...' and you should get a list of projects currently checked into the PAMGUARD sourceforge site.

Select pamssoft2006 for the main PAMGUARD Java code. AsioJniCpp, JniDAQmx and PamTiming are all JNI C projects associated with PAMGUARD for accessing Windows functions not available directly from JAVA. For general operation, you will not need the source

2. Select the correct JRE or JDK. TO do this, go to the Window/Preferences... menu, in the Preferences dialog, open the Java and Installed JRE's items in the tree view. Press the 'Add' button on the top right of the dialog and browse to the JRE or JDK you wish to use (try in the Program files/Java folder). Then select this JRE or JDK with the check box in the list view.

Wait a few seconds (or restart Eclipse) and the red cross error indicators should disappear.

### 3.2 Updating the code

Even if you are not a developer, you may want to update the code to get latest bug fixes and new features from the CVS repository. If you are a developer and wish to check in new code, then please contact the PAMGUARD guardians before doing so. Starting in the Eclipse Java Browsing view,

1. Right click on the Pamguard project you wish to update and in the menu, go to 'Team / Synchronise with Repository'. The first time you do this, the dialog shown in Figure 7 will appear. Select 'remember my decision' and say Yes.
2. You will then be taken to the Eclipse Synchronise view where you can more changes from the server onto your machine and, if you have write access, from your machine to the server.

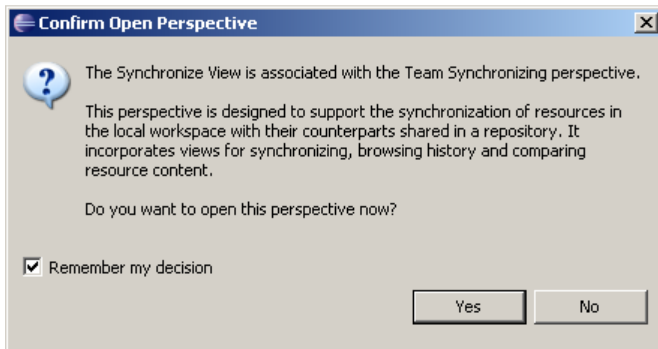


Figure 7. Switch view dialog when connecting to repository

### 3.3 Getting the code the easy way

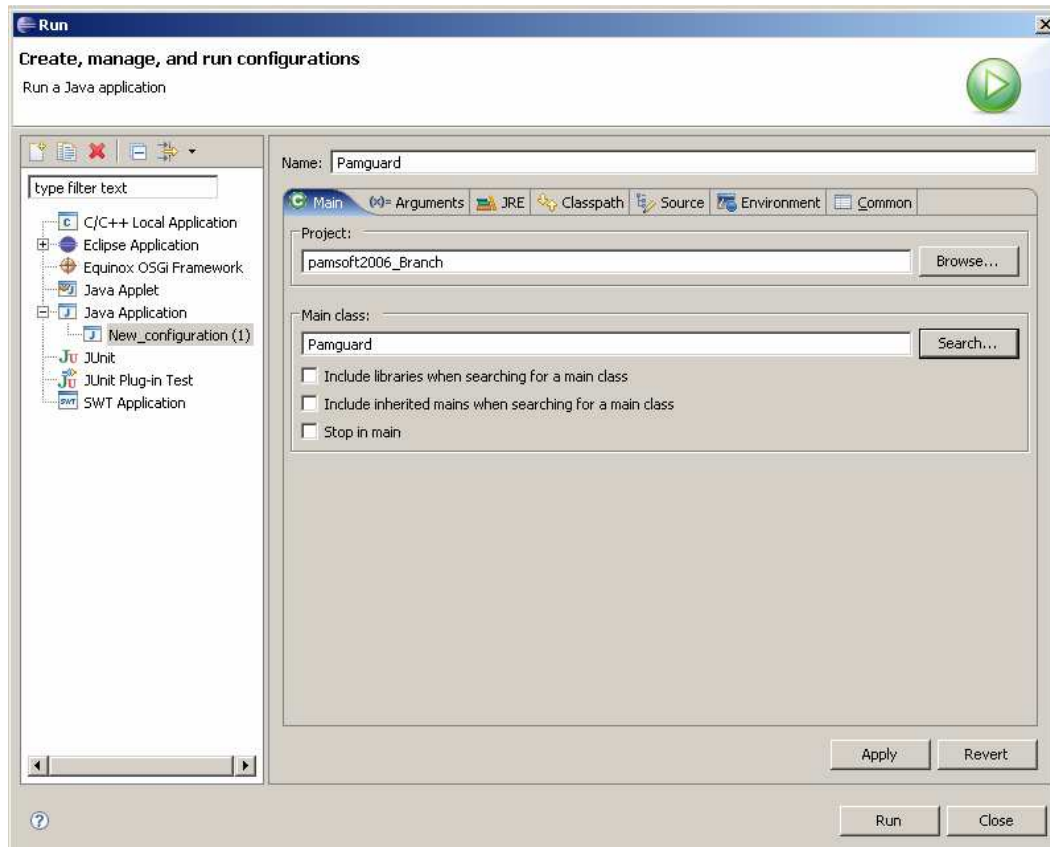
If you've been given a folder with the source code in it, then it should already include Eclipse settings which have all the DVS information in it, so all you need to do is:

1. Copy the source code directory onto your computer
2. Start Eclipse and select as your workspace the folder containing a sub folder called '.meatdata'. Do not select the .metadata folder itself, it's the folder ABOVE that that you want. It will also contain a folder called 'pamsoft2006\_Branch' or similar.
3. You may still need to sort out your jar files for rxtx and jama.jar as described in section 3.1.1. above.
4. You should then update to the latest version of the code as described in section 3.2

## 4 Running PAMGUARD from Eclipse

### 4.1 The first time

Set Eclipse to Java Browsing view.



**Figure 8. The Eclipse Run dialog**

Press the green run arrow in the Eclipse toolbar and the run dialog (Figure 8) will appear. In the left hand panel, double click 'Java Application' and 'New\_configuration (1)' will appear. At the top, by Name write 'PAMGUARD', on the Main tab, the project should already be selected, but the 'Main class' field will be empty. Either type 'Pamguard' or press the search button and you'll get a list of runnable Java classes and select Pamguard from the list.

Press the 'Arguments' tab (Figure 9) and in the VM arguments field type '-mx1024m' which will tell the Java virtual machine to allocate a gigabyte of memory to PAMGUARD (If you have more RAM you could try -mx2048m instead).

Press Run and PAMGUARD should start.

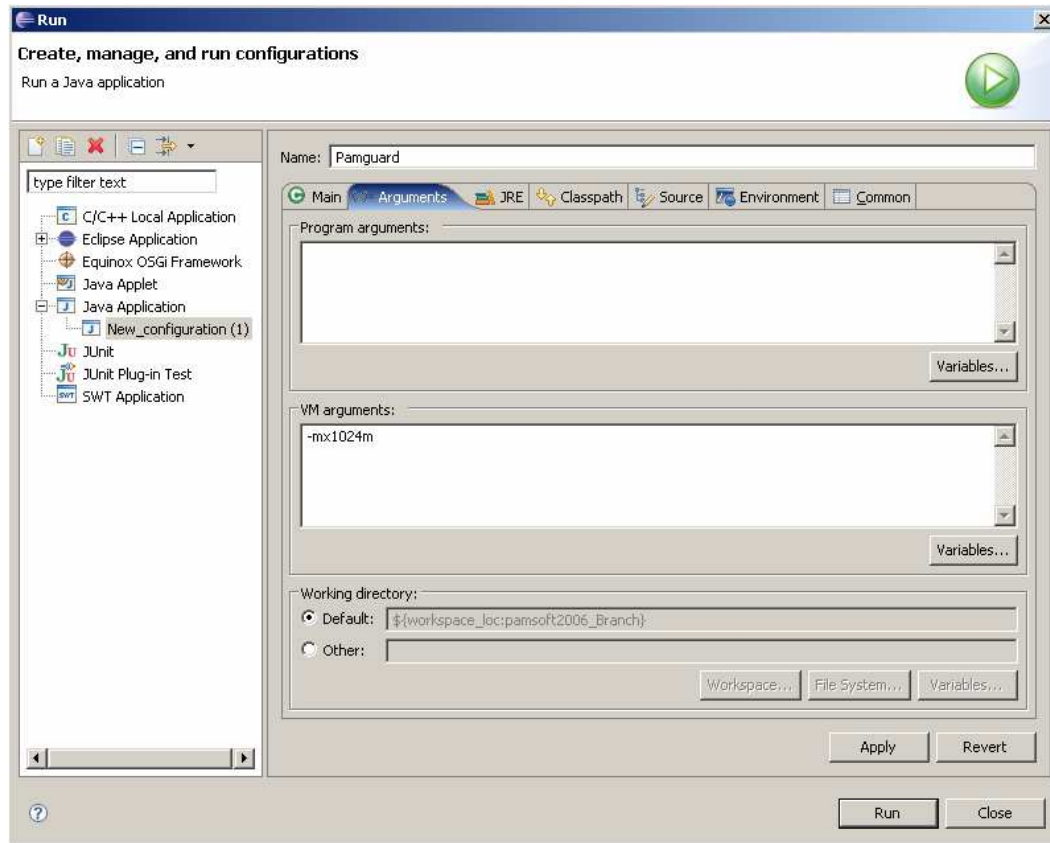


Figure 9 Run dialog Arguments tab

## 4.2 Subsequently

The above steps only need to be done once. After that, you can just press the green button and PAMGUARD will start.

If you got a disk of source code containing Eclipse configuration, then the steps described in section 4.1 should not be necessary and you can go straight to section 4.2

## 5 Running the PAMGUARD Jar File

If you have a PAMGUARD jar file which is up to date, you don't need to run Eclipse and can run the Jar more or less as you would a Windows executable.