eclipse.ini

(Redirected from Eclipse.ini)

Overview

Eclipse startup is controlled by the options in `$ECLIPSE_HOME/eclipse.ini`. If `$ECLIPSE_HOME` is not defined, the default `eclipse.ini` in your Eclipse installation directory (or in the case of Mac, the `Eclipse.app/Contents/MacOS` directory) is used.

`eclipse.ini` is a text file containing command-line options that are added to the command line used when Eclipse is started up. There are many options available, please see here.

Important:

   1. Each option and each argument to an option must be on its own line.

   2. All lines after `-vmargs` are passed as arguments to the

JVM, so all arguments and options for eclipse must be specified before `–vmargs` (just like when you use arguments on the command-line)

3. Any use of `–vmargs` on the command-line replaces all `–vmargs` settings in the .ini file unless `--launcher.appendVmargs` is specified either in the .ini file or on the command-line. (doc)

By default, `eclipse.ini` looks something like this (the exact contents will vary based on operating system and which Eclipse package you have):

-startup

../../../plugins/org.eclipse.equinox.launcher_1.2.0.v20110502.jar

--launcher.library

../../../plugins/org.eclipse.equinox.launcher.cocoa.macosx.x86_64_1.1.100.v20110502

-product

org.eclipse.epp.package.jee.product

--launcher.defaultAction

openFile

-showsplash

org.eclipse.platform

--launcher.XXMaxPermSize

256m

--launcher.defaultAction

openFile

-vmargs

-Dosgi.requiredJavaVersion=1.5

-XX:MaxPermSize=256m

-Xms40m

-Xmx512m

Among other things, this sets the heap space to 40MB initially and a maximum of 512MB, and also specifies a maximum PermGen size of

256MB. A max heap of 512MB might be OK for some users, but it's often necessary to bump that value up for large project sets or when some third-party plugins are installed.

Specifying the JVM

One of the most recommended options to use is to specify a specific JVM for Eclipse to run on. Doing this ensures that you are *absolutely certain* which JVM Eclipse will run in and insulates you from system changes that can alter the "default" JVM for your system. Many a user has been tripped up because they thought they knew what JVM would be used by default, but they thought wrong. eclipse.ini lets you be **CERTAIN**.

The following examples of `eclipse.ini` demonstrate correct usage of the -vm option.

Note the format of the -vm option - it is important to be exact:

- The -vm option and its value (the path) must be on separate lines.
- The value must be the full absolute or relative path to the Java executable, not just to the Java home directory.
- The -vm option must occur before the -vmargs option, since everything after -vmargs is passed directly to the JVM.

Here is an example of what `eclipse.ini` *might* look like on a Windows system after you've added the `-vm` argument and increased the maximum heap space:

-startup
plugins/org.eclipse.equinox.launcher_1.2.0.v20110502.jar
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_1.1.100.v20110502
-product
org.eclipse.epp.package.java.product
--launcher.defaultAction
openFile
--launcher.XXMaxPermSize

256M

-showsplash

org.eclipse.platform

--launcher.XXMaxPermSize

256m

--launcher.defaultAction

openFile

-vm

C:\Java\JDK\1.6\bin\javaw.exe

-vmargs

-Dosgi.requiredJavaVersion=1.5

-Xms40m

-Xmx1024m

Remember that the exact values will differ slightly depending on operating system and Eclipse package.

-vm value: Windows Example

This is how the `-vm` argument might look on Windows (your exact path to `javaw.exe` could be different, of course):

-vm

C:\Java\JDK\1.6\bin\javaw.exe

This might not work on all systems. If you encounter "*Java was started but returned exit code=1*" error while starting the eclipse, modify the `-vm` argument to point to `jvm.dll` (exact path could be different):

-vm

C:\Development\Java\64bit\jdk1.7.0_09\jre\bin\server\jvm.dll

-vm value: Linux Example

This is how the -vm argument might look on Linux (your exact path to `java`could be different, of course):

 -vm

/opt/sun-jdk-1.6.0.02/bin/java

-vm value: Mac OS X Example

On a **Mac OS X** system, you can find `eclipse.ini` by right-clicking (or Ctrl+click) on the Eclipse executable in Finder, choose Show Package Contents, and then locate `eclipse.ini` in the `MacOS` folder under `Contents`.

To specify Java 6 for OS X:

 -vm
/System/Library/Frameworks/JavaVM.framework/Versions/1.6.0/Home/bin/java

For versions of Mac OS X 10.7+ the location has changed to /Library/Java/JavaVirtualMachines/<"jdk_name_ver">/Contents/Home/ ...

To be safer, determine the location for the JDK you intend to use via the utility /usr/libexec/java_home and put this value with .../bin/java appended into the Eclipse.ini file.