Debugging with the Maven Jetty Plugin inside Eclipse

Debugging with the Maven Jetty Plugin in Eclipse

Step 1
Go to the Run/External Tools/External Tools ..." menu item on the "Run" menu bar. Select "Program" and click the "New" button. On the "Main" tab, fill in the "Location:" as the full path to your "mvn" executable. For the "Working Directory:" select the workspace that matches your webapp. For "Arguments:" add `jetty:run`.

Move to the "Environment" tab and click the "New" button to add a new variable named `MAVEN_OPTS` with the value:

```
-Xdebug -Xnoagent -Djava.compiler=NONE
-Xrunjdwp:transport=dt_socket,address=4000,server=y,suspend=y
```

If you supply suspend=n instead of suspend=y you can start immediately without running the debugger and launch the debugger at anytime you really wish to debug.

Step 2

Then, pull up the "Run/Debug/Debug ..." menu item and select "Remote Java Application" and click the "New" button. Fill in the dialog by selecting your webapp project for the "Project:" field, and ensure you are using the same port number as you specified in the `address=` property above.

Now all you need to do is to Run/External Tools and select the name of the maven tool setup you created in step 1 to start the plugin and then Run/Debug and select the name of the debug setup you setup in step2.

From instructions provided by Rolf Strijdhorst on the Maven mailing list

Stopping Jetty

In order to stop the jetty server the "Allow termination of remote VM" should be checked in debug dialog in Step 2. When you have the jetty server running and the debugger connected you can switch to the debug perspective. In the debug view, right click on the Java HotSpot(TM) Client VM[localhost:4000] and chose terminate. This will stop the debugger and the jetty server.