MyEclipse

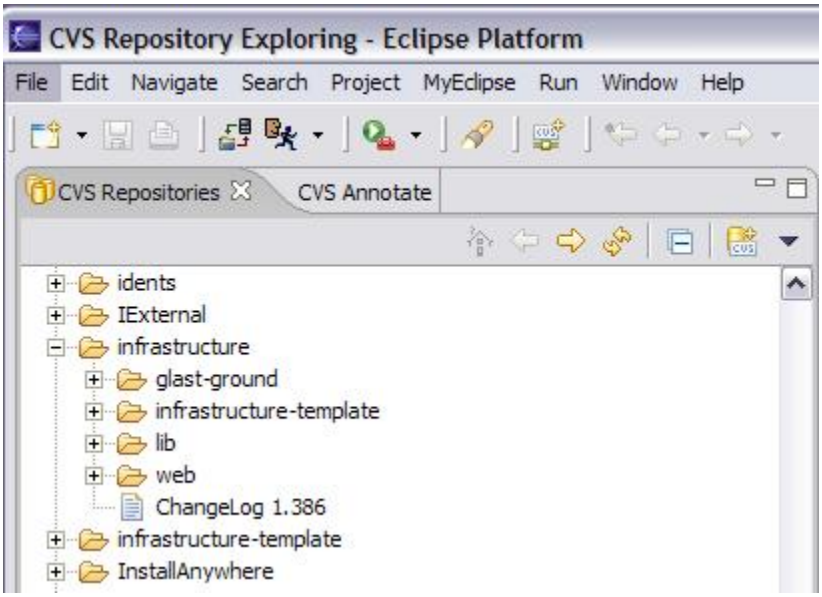
Introduction

This is intended to lead you from scratch (well, you have Eclipse and MyEclipse installed and can look at our cvs repository) to obtain, configure, build and run applications in the glast-ground project.

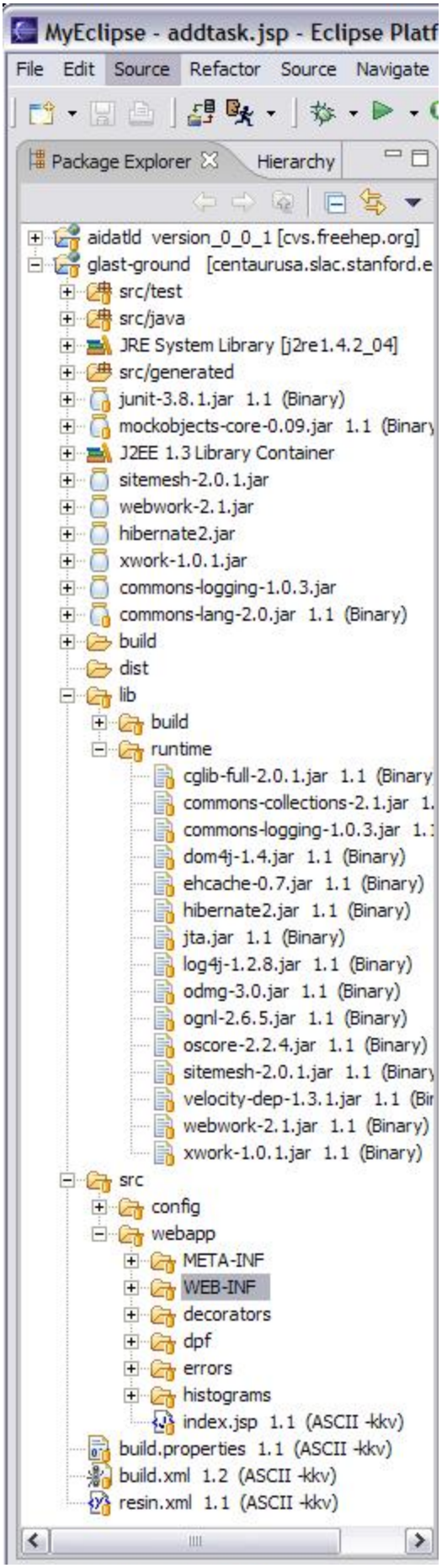
All the projects currently live in the glast-ground directory under infrastructure in the cvs repository.

Getting the Project

Open the cvs repository view and navigate to the infrastructure directory. You'll see glast-ground under it. Right click on glast-ground and check it out. This will set up automatically as a project in Eclipse.



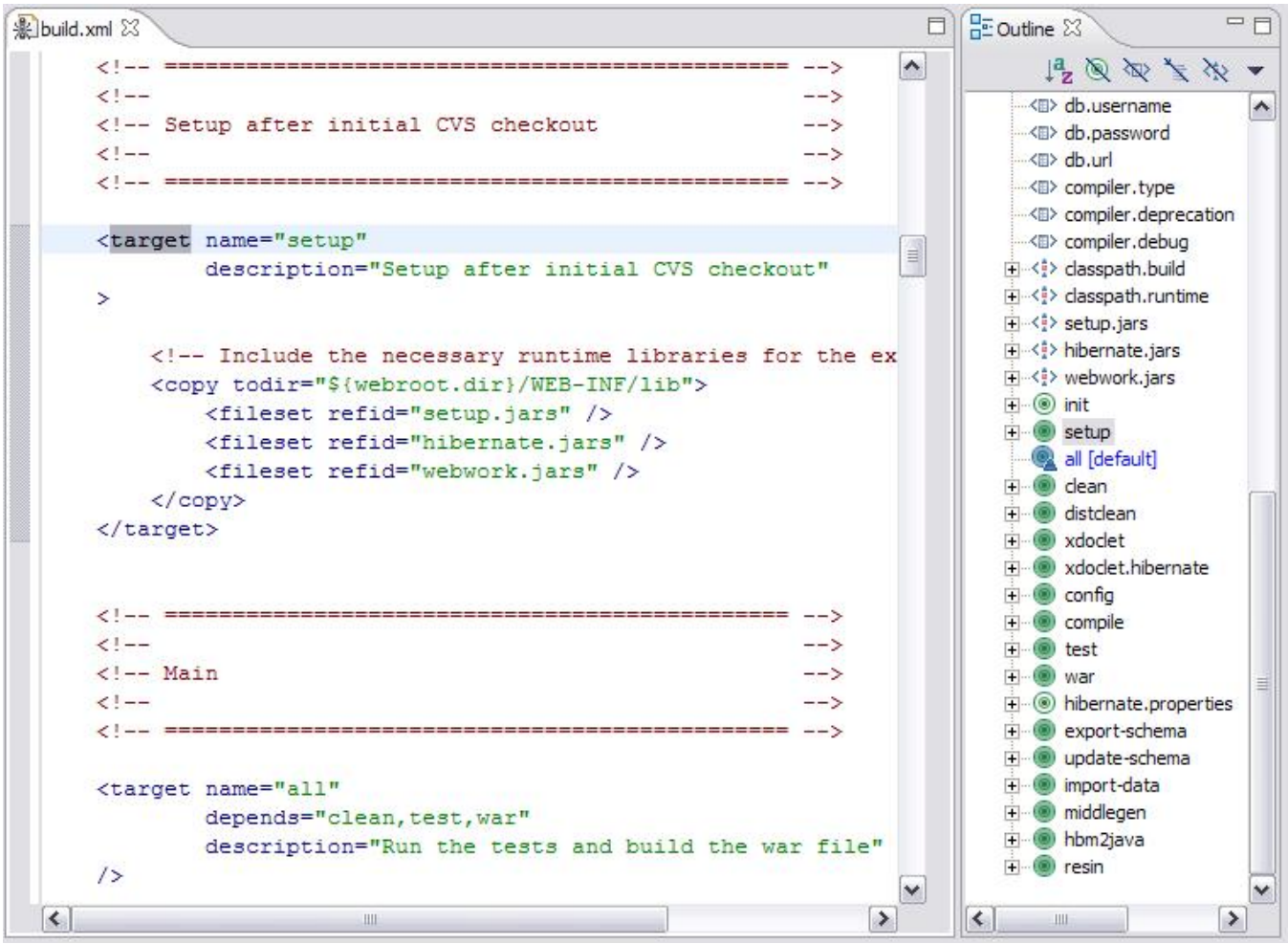
Once you have checked out the project you should see something like this:


MyEclipse - addtask.jsp - Eclipse Platf
File Edit Source Refactor Source Navigate
Package Explorer
Hierarchy
aidatld version_0_0_1 [cvs.freehep.org]
glast-ground [centaurusa.slac.stanford.e
src/test
src/java
JRE System Library [j2re1.4.2_04]
src/generated
junit-3.8.1.jar 1.1 (Binary)
mockobjects-core-0.09.jar 1.1 (Binary
J2EE 1.3 Library Container
sitemesh-2.0.1.jar
webwork-2.1.jar
hibernate2.jar
xwork-1.0.1.jar
commons-logging-1.0.3.jar
commons-lang-2.0.jar 1.1 (Binary)
build
dist
lib
build
runtime
cglib-full-2.0.1.jar 1.1 (Binary
commons-collections-2.1.jar 1.
commons-logging-1.0.3.jar 1.
dom4j-1.4.jar 1.1 (Binary)
ehcache-0.7.jar 1.1 (Binary)
hibernate2.jar 1.1 (Binary)
jta.jar 1.1 (Binary)
log4j-1.2.8.jar 1.1 (Binary)
odmg-3.0.jar 1.1 (Binary)
ognl-2.6.5.jar 1.1 (Binary)
oscore-2.2.4.jar 1.1 (Binary)
sitemesh-2.0.1.jar 1.1 (Binary
velocity-dep-1.3.1.jar 1.1 (Bi
webwork-2.1.jar 1.1 (Binary)
xwork-1.0.1.jar 1.1 (Binary)
src
config
webapp
META-INF
WEB-INF
decorators
dpf
errors
histograms
index.jsp 1.1 (ASCII -kkv)
build.properties 1.1 (ASCII -kkv)
build.xml 1.2 (ASCII -kkv)
resin.xml 1.1 (ASCII -kkv)

WEB-INF - glast-ground/src/webapp

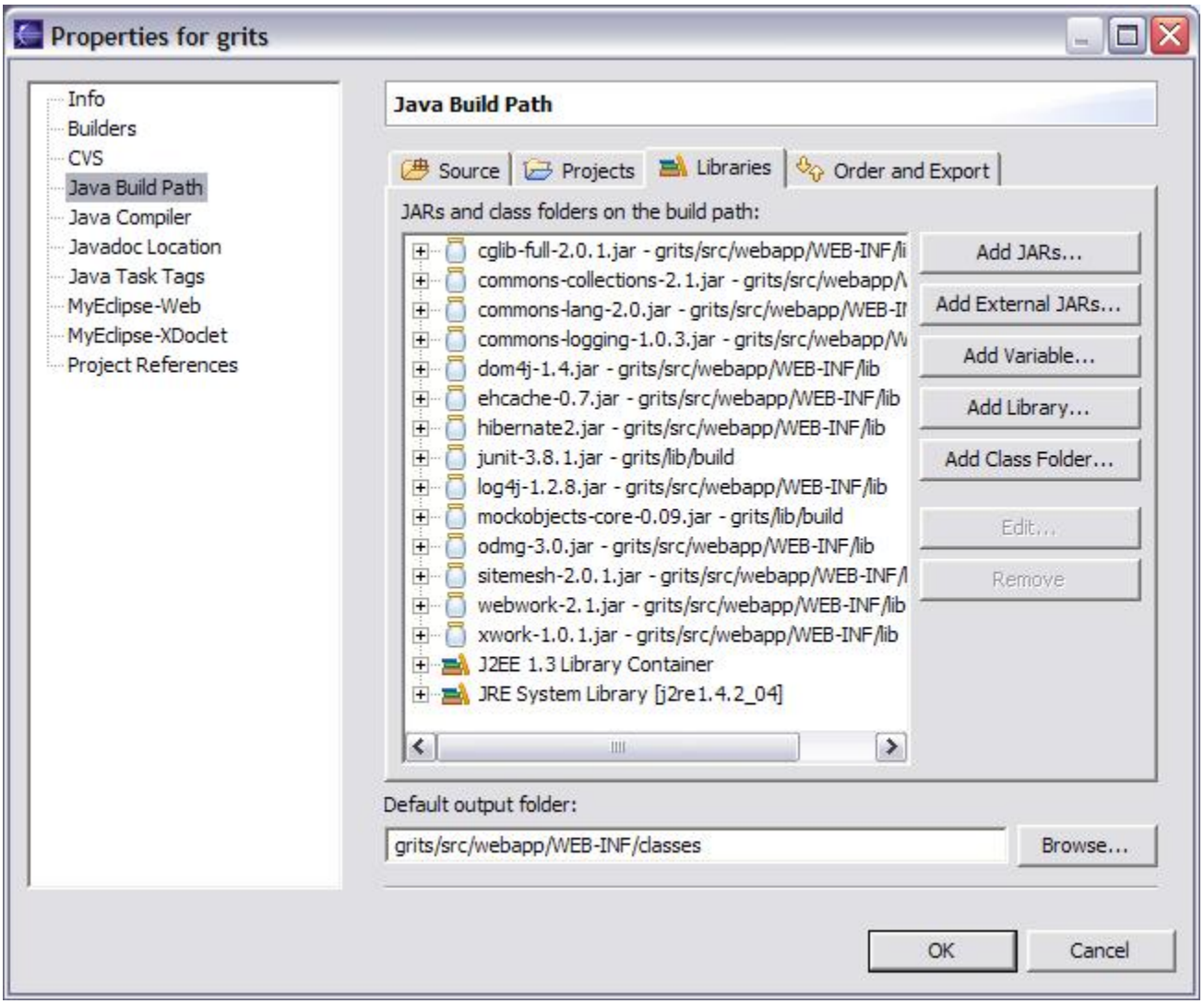
Access to Needed Jar Files

You need to run the ant setup target to copy jar files to their appropriate home in classpath. Do this by right-clicking on setup in the Outline view and selecting Run->Ant Build.



These will leave you two jar files short.

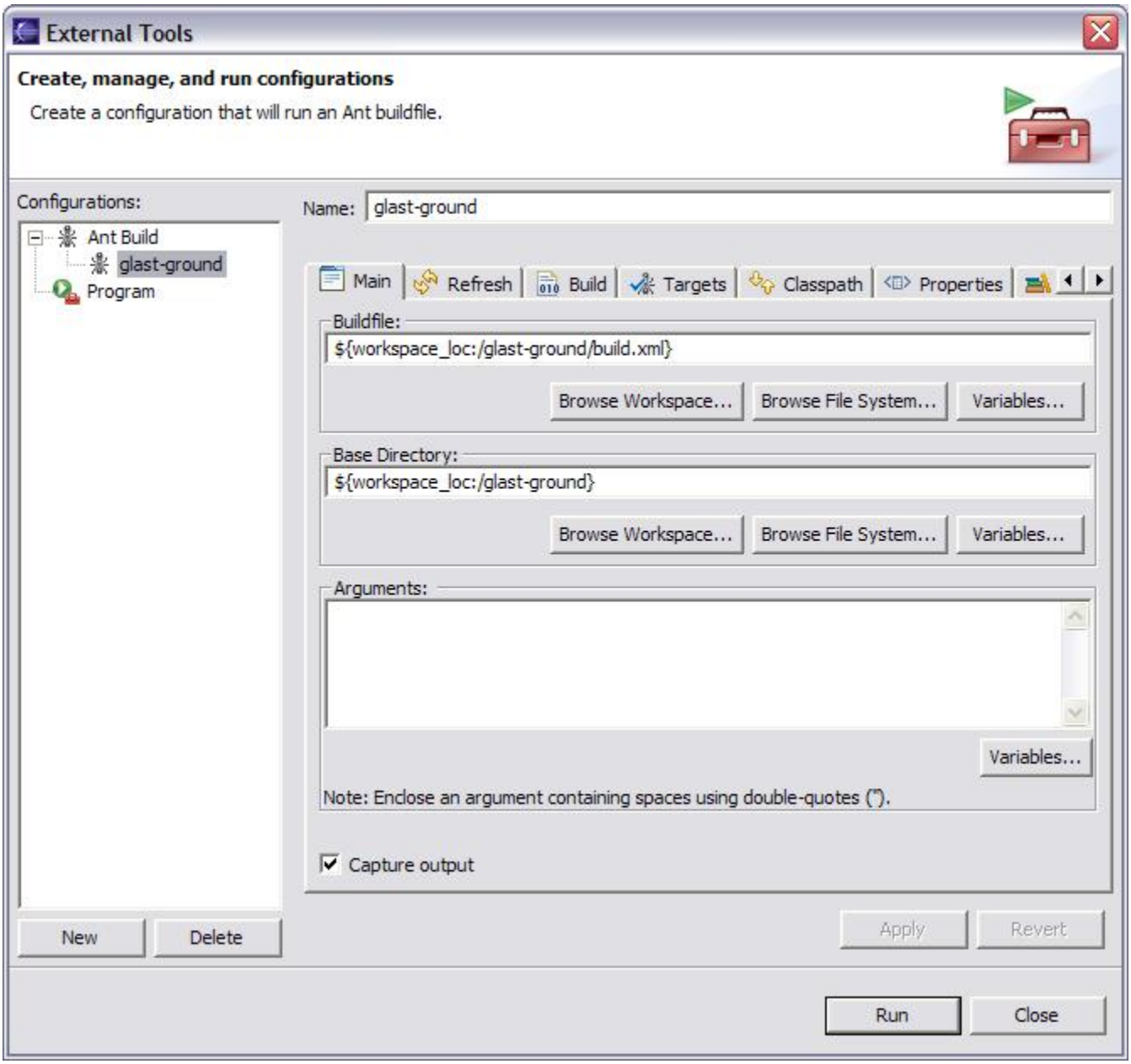
Add them to the Java build path by right-clicking on the grits project and selecting Properties, then Java Build Path. Add junit and mockobjects from lib /build.



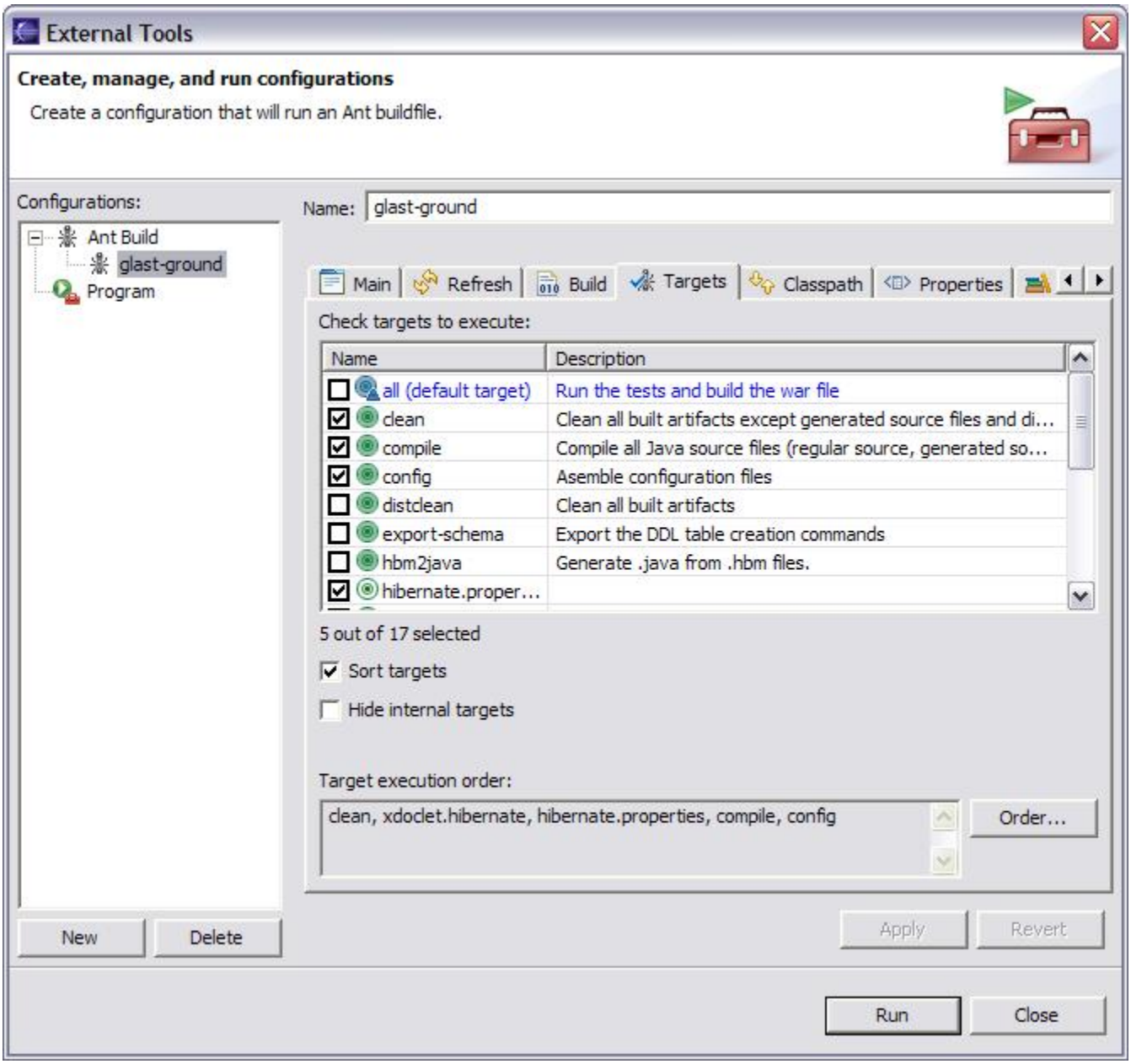
If you didn't get them in the right place, they are generally to be found in the lib/runtime directory and should be copied into the afore-mentioned lib directory.

Configuring to Build: Ant

Basically you'll want to tell Eclipse to use your workspace to build into. You get there by clicking on External Tools... option from the Run tool  in the tool bar to see:



Next you'll want to set up your build targets, **in order**. Click the names you see in the bottom box that echoes the targets - in that order.



Matt: why isn't ant set up in the repository version?

Now you can run the build by hitting the run button, which builds glast-ground by default. After you build your project, you should also recompile your JSPs, via a right-click on the project following MyEclipse->Recompile All JSPs.

J2EE server

We're using Tomcat 5 as the server. You can get it from http://jakarta.apache.org/tomcat/. Install it so you can have a local server to test with. Set it to port 8080.

Next you'll need to tell Eclipse that you're using Tomcat. I think it just figures out you have Tomcat installed and you can start it from 

Once it is started, you can deploy the project to the server via the toolbar button  .

JUnit tests

You can run the JUnit tests from the run button  pulldown menu.

Matt: did you do anything to configure my MyEclips to see the JUnit tests?

Hopefully you'll see something like this in the JUnit tab in the bottom right of the IDE