

org.lcsim in Eclipse

Using org.lcsim with Eclipse

Instead of NetBeans, you can also use the Eclipse IDE to build org.lcsim and write your applications.

Download Eclipse: <http://eclipse.org/downloads/index.php>



Source Path

You may have to change the source path of the project, otherwise Java may be inclined to look for packages `src.org.lcsim.*` instead of `org.lcsim`.
*. This can be done by right-clicking on the project and selecting Properties->Java Build Path. The `src/` folder should be selected as the only source folder.



External Dependencies

When working with external libraries such as org.lcsim, you have to add those libraries to the build path. This can be done by right-clicking on the project and selecting "Properties->Java Build Path->External Libraries". Select add external jar and choose the Jars from your JAS3 installation path. They are in the subdirectories `extensions` and `libs` as well as in `$HOME/.JAS3/extensions`

LCSim with Europa

First, install [org.lcsim](#) using command-line Maven 2.

1. Checkout [GeomConverter](#) and [org.lcsim](#).
2. Build GeomConverter, first, and then org.lcsim with the `mvn` command. (This ensures that all the lcsim dependencies are built/downloaded before we start messing with Eclipse.)
3. If either build complains about missing `lcsim-base`, then checkout that project from SLAC CVS and build it, too.

Now, setup Eclipse with Maven 2.

1. Get an appropriate [Eclipse](#) Europa build.
2. Unpack and install the Eclipse build and open Eclipse by double-clicking on its icon. (In OSX, this is `eclipse.app`.)
3. Go to **Help -> Software Updates -> Look for new features to install**. Click **Next**.
4. Click **New Remote Site**.
5. Put "q4e" in the name field and use this URL:

```
http://q4e.googlecode.com/svn/trunk/updatesite/
```

6. Click through the rest of the q4e install and agree with all the EULAs.

Finally, create a new project in Eclipse for org.lcsim.

1. Select **File -> New -> Java Project**.
2. Name the project "LCSim".
3. Select **create project from existing source** and select your org.lcsim directory.
4. Hit **Next** a and then **Finish**.
5. I usually use command-line Maven 2 here:

```
mvn eclipse:eclipse
```

6. You can also try right-clicking on the LCSim project and selecting **Maven 2 -> Execute Goal**. In the command box, put `eclipse:eclipse` and **ok**.
7. Right-click on the LCSim project in the Project Explorer window and click **Refresh**.
8. Select **Maven 2 -> Reload Maven** and let Eclipse think.
9. Now select **Maven 2 -> Enable Maven dependency management**.

If the last step executes successfully, then all the nasty red errors should go away and command completion should work.



You can quickly build Maven 2 projects from within eclipse by executing the `install` goal with `maven.test.skip` set to `true`. This should rebuild the JAR files if you have made changes.

Known Working Config

Platform	Eclipse	Java	Q4E
----------	---------	------	-----

Linux	3.3.2	1.5	0.7.0.200805270033
OSX 10.4	3.3.1	1.5	0.6