

Integrating 3rd-party libraries

Integrating existing Java libraries is easy in CShell. All you need to do is run the "Plugin from Existing Jar Archive" wizard, with only a couple non-default values.

First, in Eclipse, go to "File -> New -> Other". You should see a wizard selection menu that looks like this:

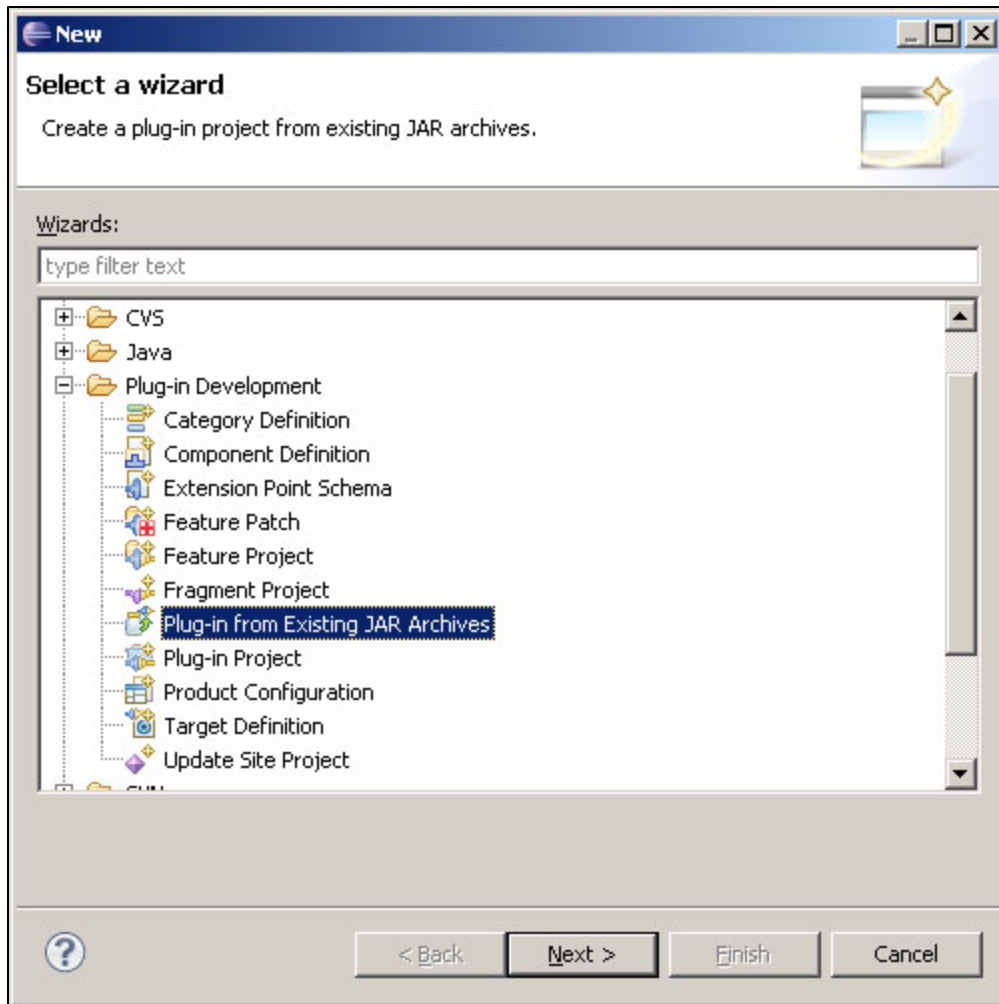


Figure 1: New Plug-in Wizard.

Select "Plug-in from existing JAR archive" from the "Plug-in development" folder, and click "Next>".

The next page should look like this:

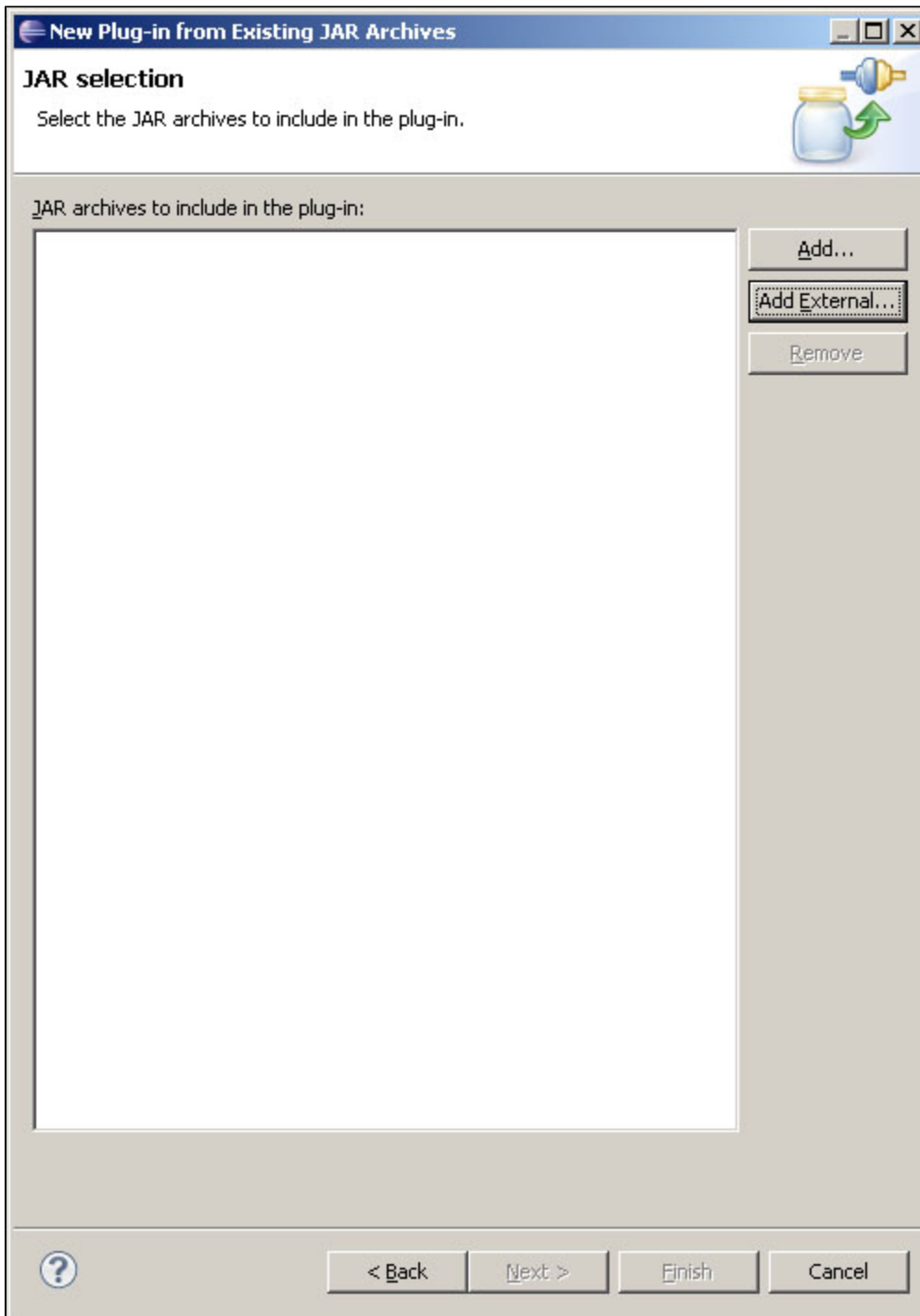


Figure 2: Plug-in from existing JAR Archive.

Select "Add External...", and include all the .jar libraries you would like to include in the library plugin we are creating. When you are done, click "Next>".

The following page should look like this:

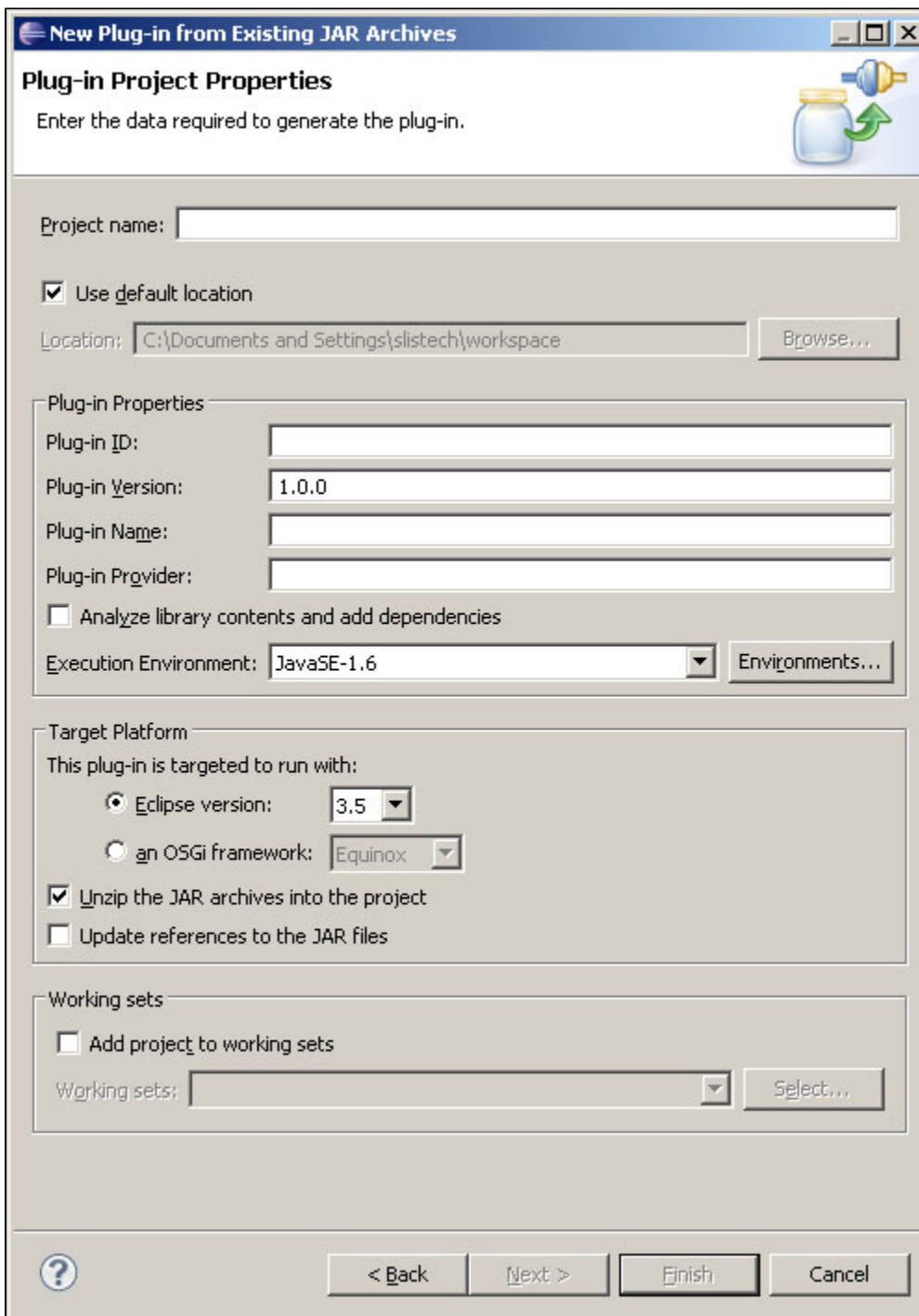


Figure 3: Adding external JARs.

For the project name, you may use any name you like. Set the execution environment to be compatible with whatever version of the JRE you are targeting for your tool.

In the "Target Platform" area of the page, select the "an OSGi Framework" and "Equinox".

Uncheck the "Unzip the JAR archives into the project" check box.

When you are finished, click "Finish".

You should now see a project in your workspace containing all the jar archives we selected earlier. Notice that all the library packages are exported in the project's META-INF/MANIFEST.MF. This allows our other CShell plugins to import these packages, thus making use of the libraries.

Whenever you create algorithm plugins which depend on library plugins, make sure to export the library plugins into your CShell tool along with the algorithm plugins. Otherwise, your algorithm plugins will not be able to resolve their package dependencies when they are executed from within the tool.