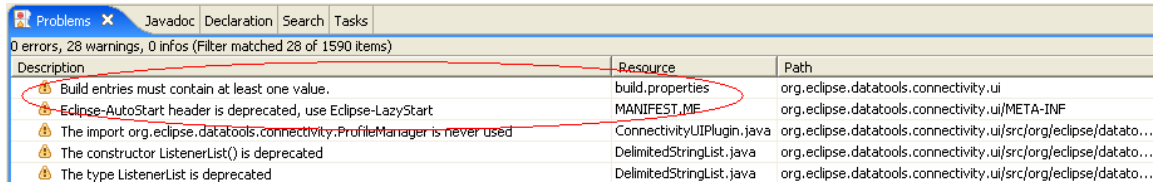


The following instructions are derived from those used by the platform team:

http://www.eclipse.org/eclipse/platform-core/documents/3.1/run_from_jars.html

By way of illustration, we will walk through jarring *org.eclipse.datatools.connectivity.ui*. All of these modifications are made using Eclipse 3.2M5a with auto-build enabled.

1. When the plug-in is loaded into Eclipse and built, a number of warnings might appear in the Problems View. In particular, we are interested in the following:



Description	Resource	Path
Build entries must contain at least one value.	build.properties	org.eclipse.datatools.connectivity.ui
Eclipse-AutoStart header is deprecated, use Eclipse-LazyStart	MANIFEST.MF	org.eclipse.datatools.connectivity.ui/META-INF
The import org.eclipse.datatools.connectivity.ProfileManager is never used	ConnectivityUIPlugin.java	org.eclipse.datatools.connectivity.ui/src/org/eclipse/datato...
The constructor ListenerList() is deprecated	DelimitedStringList.java	org.eclipse.datatools.connectivity.ui/src/org/eclipse/datato...
The type ListenerList is deprecated	DelimitedStringList.java	org.eclipse.datatools.connectivity.ui/src/org/eclipse/datato...

The first warning can be fixed by opening *build.properties* and deleting the following line:

```
jars.extra.classpath =
```

The second warning is a result of DTP 0.7 backward compatibility with Eclipse 3.1. For DTP CVS head (that is, DTP versions later than 0.7), Eclipse 3.1 support is not required, so this deprecated property can be changed. Open the *manifest.mf* file and change:

```
Eclipse-AutoStart: true
```

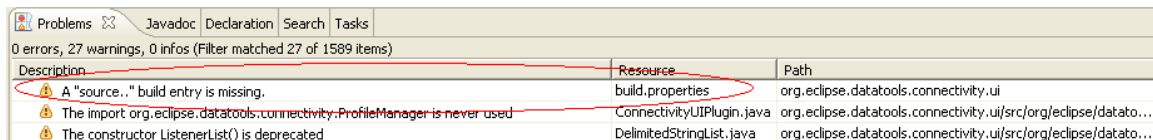
to the following:

```
Eclipse-LazyStart: true
```

Also, if you find the *Eclipse-AutoStop*, then delete it completely. While you are editing the manifest file, also remove the plug-in jar property:

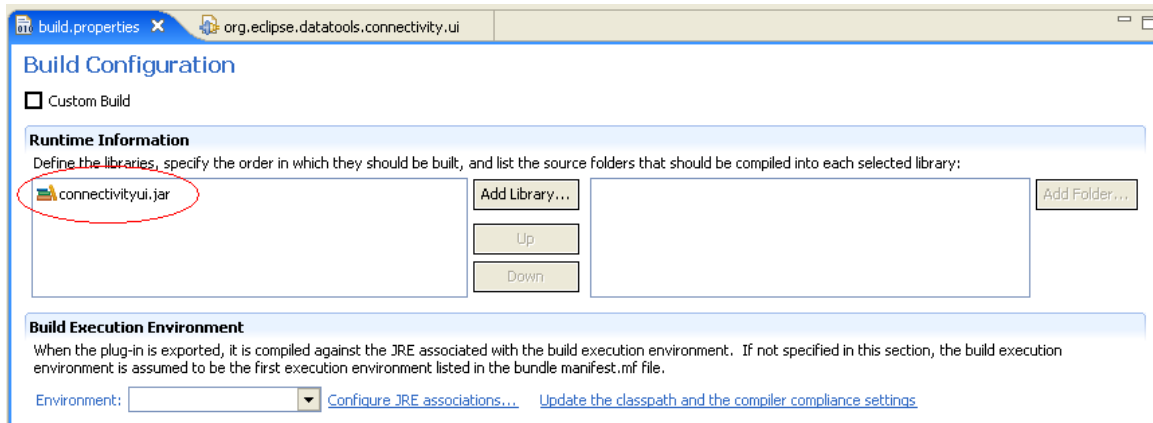
```
Bundle-ClassPath: connectivityui.jar
```

Finally, save and close both the manifest and build.properties files. After building, a new set of warning should appear:

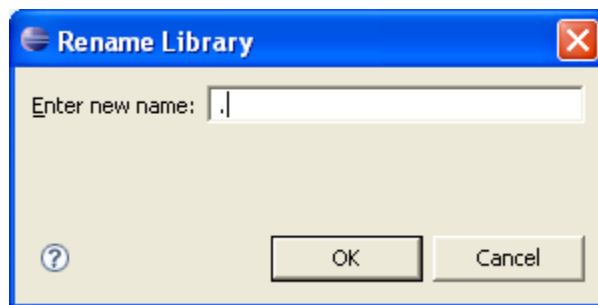


Description	Resource	Path
A "source.." build entry is missing.	build.properties	org.eclipse.datatools.connectivity.ui
The import org.eclipse.datatools.connectivity.ProfileManager is never used	ConnectivityUIPlugin.java	org.eclipse.datatools.connectivity.ui/src/org/eclipse/datato...
The constructor ListenerList() is deprecated	DelimitedStringList.java	org.eclipse.datatools.connectivity.ui/src/org/eclipse/datato...

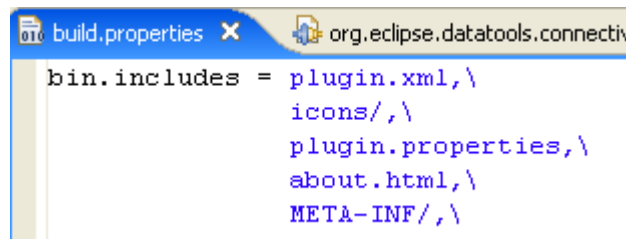
Open the *build.properties* file in the build file editor, and flip to the “build” tab. You will see the old plug-in jar name still present:



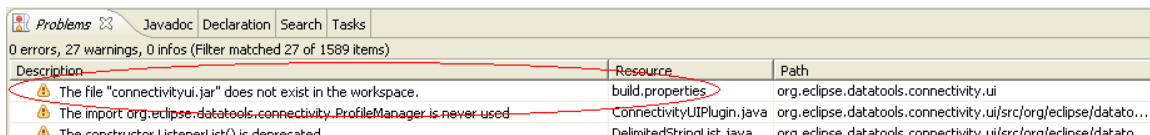
Right click on this library name, and choose “Rename.” When the rename dialog appears, delete the old name and enter a single dot:



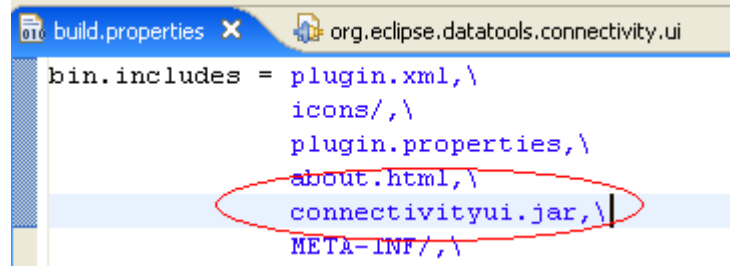
Click on “OK” then save build.properties. Finally, make sure that the *bin.includes* section was updated correctly. Flip to the “build.properties” tab and look at bin.includes. It should have a dot (.), but not the name of the old library:



If it has the name of the old library, then you should see a warning in the Problems View:

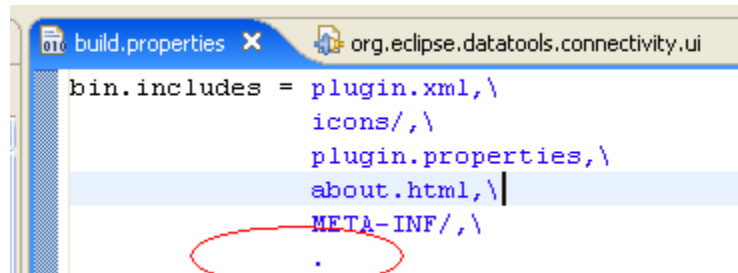


This means that the bin.includes was not updated correctly. Change it to remove the old library name:



```
build.properties x org.eclipse.datatools.connectivity.ui
bin.includes = plugin.xml,\
               icons/,\
               plugin.properties,\
               about.html,\
               connectivityui.jar,\
               META-INF/,\
```

Which results in:



```
build.properties x org.eclipse.datatools.connectivity.ui
bin.includes = plugin.xml,\
               icons/,\
               plugin.properties,\
               about.html,\
               connectivityui.jar.\
               META-INF/,\
               .
```

Notice the dot added to bin.includes.

You should now be able to export the plug-in as a deployable jar (the plug-in exporter creates a plug-in jar automatically for those plug-ins who library is designated with the dot). Many things can go wrong when a plug-in is jarred: for example, resources such as icons may fail to load. Therefore, you will need to thoroughly test your plug-in after jarring, to verify that it still works as expected.