

## Kepler - Bug #4191

### Eclipse build is slow the first time "Copying resources to the output folder"

06/26/2009 03:57 PM - Christopher Brooks

<b>Status:</b>	Resolved	<b>Start date:</b>	06/26/2009
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	jianwu jianwu	<b>% Done:</b>	0%
<b>Category:</b>	build system	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	2.2.0	<b>Spent time:</b>	0.00 hour
<b>Bugzilla-Id:</b>	4191		

#### Description

I just verified that under Windows, the Eclipse build has errors because of apple-extension.

David Welker wrote:

Hi Christopher,

That this problem would occur in Windows makes total sense. Right now, the 'ant eclipse' command is not aware of operating system exclusions. I am going to discuss a proposed solution with Chad. In the meantime, the solution is to delete the apple-extensions project folder, which has Mac OS X-only code.

David

Hi David,

I just installed Eclipse-rcp-galileo under Windows 2003 Server with Java 1.6.0\_14 and after adding tools.jar to the path, I'm still getting errors in the compilation:

Description	Resource	Path	Location	Type
Application cannot be resolved	KeplerOSXExtension.java	/apple-extensions/src/org/kepler/osx	line 17	Java Problem
Application cannot be resolved to a type	KeplerOSXExtension.java	/apple-extensions/src/org/kepler/osx	line 17	Java Problem
ApplicationEvent cannot be resolved to a type	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 19	Java Problem
ApplicationEvent cannot be resolved to a type	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 26	Java Problem
ApplicationEvent cannot be resolved to a type	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 31	Java Problem
ApplicationEvent cannot be resolved to a type	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 36	Java Problem
ApplicationEvent cannot be resolved to a type	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 41	Java Problem
ApplicationEvent cannot be resolved to a type	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 49	Java Problem
ApplicationEvent cannot be resolved to a type	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 77	Java Problem
ApplicationListener cannot be resolved to a type	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 17	Java Problem
The import com.apple.eawt cannot be resolved	KeplerOSXExtension.java	/apple-extensions/src/org/kepler/osx	line 3	Java Problem
The import com.apple.eawt cannot be resolved	KeplerApplicationListener.java	/apple-extensions/src/org/kepler/osx	line 3	Java Problem

Can you take a look?

This is with a clean checkout.

Also, how difficult would it be to create a test that would build Eclipse using the output of "ant eclipse"? It seems like until we have a regularly run test for the IDEs then we will always have problems.

## History

---

### #1 - 06/26/2009 04:33 PM - Christopher Brooks

The first time Eclipse is used to build Kepler, the build hangs for several minutes.

I had one user who thought that the Kepler/Eclipse build was broken because it was taking several minutes for him.

What happens is that during the build after importing the projects into Kepler, "Refreshing workspace" appears in the lower right. If I click on the progress bar next to "Refresh workspace", I get a Progress tab with a countdown on the number of resources to refresh. For me, on a reasonably sporty server, this stage took 4 minutes. This is not so bad, it appears that we are making progress.

I then get "Building workspace (12%)" and after about a minute I get "Copying resources to the output folder".

Then, for about 8 minutes there is nothing but "Copying resources to the output folder". This is where the user gave up, they thought Eclipse was dead. Is there anything we can do here?

All told, the build took 16 minutes, which is not so bad, but the problem is the 8 minutes of dead time in the middle.

### #2 - 08/06/2009 11:30 AM - Chad Berkley

I ran into this when trying to setup eclipse this morning. I couldn't see any reason for it. I ended up killing eclipse several times because I thought it was hung.

### #3 - 08/06/2009 02:05 PM - Aaron Aaron

I think this has to do with having many projects in eclipse that depend on each other. Try turning off "build automatically" and then importing the projects. Then build only the top level project (wrp for example). Eclipse will then build the other projects via dependency. From what I can tell this is quite a bit faster although I haven't experimented much with it. I suspect it takes longer to auto build because of duplicately building projects through other project's dependencies.

### #4 - 09/17/2009 04:17 PM - Aaron Aaron

I have explored this problem a little further. Turns out Eclipse will copy all the .svn folders to the classes directory if you are not sharing the project using the subversive plugin.

A quick experiment

Building the ptolemy project without svn sharing: 14 min 03 sec 206 MB copied

Building the ptolemy project with the project shared: 3 min 18 sec 93 MB copied

You can control what resources are copied at build time in Preferences->Java->Compiler->Building->Output folder->Filtered resources

Using the following filter

**.launch,.svn,makefile,.tcl,\*.htm,\*.html**

Building the ptolemy project with the filter and svn: 2 min 08 sec 82 MB copied

You can configure that filter specifically for a project too which the kepler build system should do automatically.

To close this bug:

1) figure out the best filter for copying resources

2) generate the org.eclipse.jdt.core.prefs and org.eclipse.jdt.launching.prefs files in the .settings directory whenever the "ant eclipse" build target is run

### #5 - 07/07/2010 05:23 PM - Christopher Brooks

The bug is that build is slow the first time under Eclipse.

For me on an older Windows 2003 Server machine, I have about 8.5 minutes of "Refreshing Workspace", which makes it look like the build is hung.

For me, the build then finished in about 2 minutes, but I have 31 errors because I did not update the path to include tools.jar and because of problems with apple-extensions (Bug 4342) and

Invalid archive:

src/lib/swing-workers-license.htm

src/lib/ptll.properties.in

I'm targeting this to 2.1.0 and marking it as 4 hours. Aaron outlined a possible solution.

**#6 - 08/09/2010 11:55 AM - jianwu jianwu**

The status I found is that only ptolemy module doesn't filter .svn correctly. I updated the build-area\resources\eclipse\PtolemyClasspathStart file to add more filter for .svn.

My experiments on Windows 7:

build ptolemy before update: 1 min 30 sec 97 MB copied

build ptolemy after update: 1 min 93 MB copied

**#7 - 08/09/2010 03:29 PM - jianwu jianwu**

I exclude test directories in ptolemy.classpath file so that the build will be faster. Now building ptolemy takes about 45 seconds on my machine and the size is reduced to 60 MB.

If I exclude demo directories in ptolemy, the time will be about 35 seconds and the size is reduce to 35 MB. But I didn't check it in trunk since there are some classes in demo directories (such as ./ptolemy/actor/lib/net/demo/Datagram/Datagram.class). If I did it, some demo workflows might not working.

I also tested updating org.eclipse.jdt.core.prefs file, but my tests didn't show much time difference. So I didn't follow that way.

**#8 - 03/27/2013 02:25 PM - Redmine Admin**

Original Bugzilla ID was 4191