

Kepler - Bug #3997

Supporting display actor conversion for command line execution.

04/20/2009 02:20 PM - jianwu jianwu

Status:	Resolved	Start date:	04/20/2009
Priority:	Immediate	Due date:	
Assignee:	jianwu jianwu	% Done:	0%
Category:	build system	Estimated time:	0.00 hour
Target version:	2.4.0	Spent time:	0.00 hour
Bugzilla-Id:	3997		

Description

When executing workflow for batch mode, users may want the display actors in the workflow can be converted automatically. We can enable it by adding one more option: "-displayRevert path/to/put/diplay/files". We can use the jar of Hydrant to filter the display actors to the corresponding file writing actors before workflow execution. This is already implemented in Kepler Web service and CAMERA project. Tristan also said his jar can be updated and re-distributed.

History

#1 - 04/20/2009 02:30 PM - Christopher Brooks

Am I to understand that Hydrant has this code already?
Cool.

It seems a little heavyweight to require Hydrant here, perhaps a lighter weight solution, such as adding a MoML filter could be used? I'd be willing to fold changes into the ptII tree to more easily support this.

#2 - 04/20/2009 02:51 PM - jianwu jianwu

What's in the Hydrant is just a new MoML filter and a set of new classes which extend display related actors, like Display and PlotterBase, and rewrite their postfire()/wrapup() to write the data to files.

Will it be the same way if support it from PTII?

#3 - 06/13/2012 11:10 AM - jianwu jianwu

This bug needs to be fixed in version 2.4. I have tried locally using hydro library. It works.

We can either have a new module called hydro, or put it in existing modules.

#4 - 06/15/2012 08:16 AM - Christopher Brooks

The issue here is to make it possible to capture Display and Plotter data.

A team of CMU students modified how the plotter and Display actors work.

Basically, we now have `ptolemy.actor.lib.gui.DisplayInterface`.

```
actor.lib.gui.Display has
protected DisplayInterface _getImplementation()
which reads
ptolemy/actor/ActorModule.properties
```

The thing to do would be to make it so that we can substitute in a different ActorModule.properties file that uses different actors as an implementation.

To do this, we would need a class that calls
`ptolemy.actor.injection.ActorModuleInitializer.setInitializer()`

The class would need to be invoked via the command line.

I changed this from a blocker to an enhancement because Kepler works as defined, so this is not a bug, it is missing functionality.

I can see how it would be important to get this done, so I've left it P1.

#5 - 06/28/2012 05:10 PM - jianwu jianwu

Hi Christopher,

Thanks for your info. I tried a simple `DisplayInterface` implementation for saving the content to a file. It works! :)

Currently, Hydrant also supports this function for `ImageJActor`, `MonitorValue` and `ImageDisplay` actors. If we adopt the similar `DisplayInterface` solution for these actors, we need to re-write these actors.

#6 - 06/28/2012 05:19 PM - Christopher Brooks

Jianwu,

Cool that it worked!

It might be more work to use code like the `DisplayInterface` code, but the `DisplayInterface` style will be supported in the future. It is a more elegant solution than the filtering solution that is in Hydrant.

So, given a choice, I say move forward with `ImageJActor`, `MonitorValue` and `ImageDisplay` with the `DisplayInterface` style.

However, I'd understand if you wanted to fall back to the filter style that is in Hydrant.

I'm not so sure I want that filter-style code in the pttl tree though.

#7 - 06/28/2012 05:26 PM - jianwu jianwu

Christopher,

I agree that `DisplayInterface` solution is a better one, although it needs more work. I also prefer this way.

To move forward, I can update `ImageJActor` actor first since it is in Kepler. After that, we can discuss how to update `MonitorValue` and `ImageDisplay` in Ptolemy.

#8 - 07/17/2012 02:56 PM - jianwu jianwu

Now it works with `Display`, `Plot` and `ImageJ` actors at version 30211. A new module called "display-redirect" is added to Kepler suite to support their batch execution and file saving.

Hi Christopher,

Hydrant also used to support the batch execution of `MonitorValue` and `ImageDisplay` actors in Ptolemy. But these two actors haven't use the similar way like `display` implementation. Do you plan to update these actors in Ptolemy using the same way? If so, I'm happy to provide their batch execution support.

#9 - 07/17/2012 04:04 PM - Christopher Brooks

I saw that the hydrant code was checked in awhile ago.

I have no plans to modify `MonitorValue` and `ImageDisplay` so that they are similar to `Display`. Someone else would need to make the changes in Ptolemy II.

If the Hydrant code works and you don't want to make the changes, then we should stick with the Hydrant code. I prefer the method we are using with `Display` over adding the Hydrant code, but I don't have time to make further changes. The reason I prefer updating `MonitorValue` and `ImageDisplay` is because the amount of code will be smaller than the Hydrant code and thus `MonitorValue` and `ImageDisplay` are less likely to have bugs and are easier to maintain.

#10 - 07/31/2012 10:55 AM - jianwu jianwu

I updated ptolemy version 64086 and kepler version 30298 to support redisplay for `MonitorValue` and `ImageDisplay` actors.

Closing the bug.

#11 - 03/27/2013 02:25 PM - Redmine Admin

Original Bugzilla ID was 3997