

Kepler - Bug #2352

Preferences System needs to be resolved

02/09/2006 09:51 AM - Christopher Brooks

Status:	Resolved	Start date:	02/09/2006
Priority:	Normal	Due date:	
Assignee:	Christopher Brooks	% Done:	0%
Category:	core	Estimated time:	0.00 hour
Target version:	1.0.0beta1	Spent time:	0.00 hour
Bugzilla-Id:	2352		

Description

I'm creating a bug for this because
http://bugzilla.ecoinformatics.org/show_bug.cgi?id=2343

"add welcome screen for release 1.0" depends on it.

I wrote:

Hi Edward,

I'm hacking up the Welcome window and would like to have a way the window so it has a "Please don't show me this dialog again" checkbox and I have some questions about VergilPreferences.

- vergil.VergilPreferences should probably be renamed to PtolemyPreferences and moved to ptolemy.actor.gui so that the actor.gui.WelcomeWindow class can get at it and ptolemy.actor.gui does not depend on vergil.

In fact, getting a preference is a fairly basic task, so this should go in some non-gui base class. If we use plain old java properties, then perhaps we could have a preferences class in ptolemy.util? I'd like to be able to get at preferences without requiring moml.

Another preference I'd like to see is a way for the user to adjust how much detail the GraphicalMessageHandler window initially shows. GraphicalMessageHandler is in ptolemy.gui and only imports ptolemy.util, so MoML is not available.

- We need a way to access a global property when we don't have a container. In general, I think the Kepler developers are confused about how we should get a Configuration from anywhere in the code. So, we need a preferenceValue(String preferenceValue) that does not require a context or container and gets the value of the global preference.

Comments?

I've included some email below that has further details

_Christopher

To: Matthew Brooke <brooke@nceas.ucsb.edu>
cc: eal@eecs.berkeley.edu
From: "Christopher Brooks" <cxb@eecs.berkeley.edu>
Subject: Re: [Bug 2343] - add welcome screen for release 1.0
In-reply-to: Your message of Fri, 20 Jan 2006 16:28:29 -0800.
<43D1802D.3000607@nceas.ucsb.edu>
Date: Sat, 21 Jan 2006 18:07:54 -0800
Sender: cxb@carson.EECS.Berkeley.EDU

I've taken the liberty of ccing Edward here.

Matthew Brook writes:

Christopher:

Perhaps Edward's preferences manager
(ptolemy.vergil.VergilPreferences) can be used?

When i started the SVG icon stuff, I needed to access properties files from various parts of the code, and spent a lot of time trying to use ptll's existing configuration.xml, to no avail. I finally figured it just wasn't 'available' for enough of the codebase at runtime, without hard-coding paths in order to re-load it (or maybe I'm just too dumb...).

Right this is a problem in general. It is not always easy to get at the configuration, you usually need a model that was read in.

So - I went ahead and used java's built-in resourcebundle/properties classes for storing prefs. This is overkill/inappropriate in some places (for example, many of the settings are not localizable, so do not really need resourcebundle files), but it works marvelously. I guess in the meantime, Edward created the VergilPreferences thing, which I haven't looked at - so now we have a proliferation on our hands... :-)

Anyway - I have put all the ui-related settings in **.properties files in configs/ptolemy/configs/kepler/, and they all have names like: ui***.properties**. The 'main' one is uiSettings.properties.

Up to now, these only contain application settings (ie read-only) rather than user settings - so we still need to create a user-settings file for these read/write props. Is that what VergilPreferences currently does? if it's ultra-simple to use, and easy to get a reference to the props class (say via a static method somewhere) let's use it! If not, then let's use POJ!

m

Right, VergilPreferences saves its preferences in
~/ptolemyII/VergilPreferences.xml
or c:/Documents and Settings/username/ptolemyII/VergilPreferences.xml

I think it is fine to have the ui read only settings in
configs/ptolemy/configs/kepler

The primary entry point in VergilPreferences is:
/** Check to see whether a preference of the specified name is

- defined in the specified context, and if it is, return it's value.
- Note that if there is an error in the expression for the preference,
- then this method will return null and report the error to standard out

t.

- This is done because we assume the error will normally be caught before this method is called.
 - @param context The context for the preference.
 - @param preferenceName The name of the preference.
 - @return The value of the preference, or null if it is not set.
- */

```
public static Token preferenceValue(NamedObj context, String preferenceName)
```

me

) {

I'm not sure what the context would be for the "Show this dialog at startup". This is sort of a global context.

Perhaps we should extend VergilPreferences to provide a default global context with a static accessor? Also, it would be nice to have a static method that returns a String:
public static String preferenceValueAsString(String preferenceName);

Also, I don't see how I add a new preference? (Edward?)

I'm not particularly wedded to using .xml for the global preferences, but using xml makes sense in this context, where we are setting things like Relation size, Link bend radius and Show Parameters, and we want these things to be inherited and overridden in the model hierarchy.

However, there is quite a bit of appeal to using Plain Old Java (POJ) properties, like what we do with ptII/lib/ptII.properties, which is created by configure and contains POJ properties that are merged in by VergilApplication. POJ properties do not require MoMLParser, so they are more useful for codegen runtime and other small footprint programs.

BTW - Perhaps we should use ptII/lib/ptII.properties for global properties and first load ptII/lib/ptII.properties and then load ~/.ptolemyII/ptII.properties (if it exists) so we can override these properties.

Anyway, seems like we have lots of properties systems, we should hash something out and use it.

1) We could extend VergilPreferences:

- static accessor without a context
- static accessor that returns a String
- static setter of new Properties (with automagic save?)

2) Hack up ptII.properties

- configure sets properties, but we need to make it easy for users to override
- static setter of new Properties (with automagic save?)

3) Hack up the Kepler properties system

- Make it useful in Ptolemy only

Comments?

_Christopher

History

#1 - 04/13/2006 12:06 PM - Christopher Brooks

I'm closing this bug because

http://bugzilla.ecoinformatics.org/show_bug.cgi?id=2343

"add welcome screen for release 1.0"

is now complete

I don't think we have the perfect solution to the Preferences problem.

However, I was able to use Edward's preferences system to save the global "Show this window next time" preference for the Welcome Window.

See \$PTII/ptolemy/actor/gui/WelcomeWindow.java for details.

To do the preferences work, I had to add some code to the Preferences system, but I think we are in a good place for shipping Kepler-1.0 and Ptolemy II 6.0.

Thus, I'm closing this bug.

#2 - 03/27/2013 02:19 PM - Redmine Admin

Original Bugzilla ID was 2352