

Kepler - Bug #3986

RExpression confounding R working directory and .kepler folder

04/14/2009 06:22 PM - Oliver Soong

Status:	Resolved	Start date:	04/14/2009
Priority:	Normal	Due date:	
Assignee:	ben leinfelder	% Done:	0%
Category:	actors	Estimated time:	0.00 hour
Target version:	2.0.0	Spent time:	0.00 hour
Bugzilla-Id:	3986		
Description If I have two RExpression actors, I might expect them to operate in different working directories, yet still be able to pass data from one to the other. This fails, as shown in the example at the URL. NOTE: The example will first fail because of another bug. To see this problem, you have to first do the workaround described in the second comment of bug 3985.			
Related issues: Blocks Kepler - Bug #3985: Types resolved to unacceptable types Resolved 04/14/2009			

History

#1 - 04/23/2009 10:39 AM - ben leinfelder

retarget and reassign to myself
(hunch is that working directory hasn't been "working" in a while since it subverts the idea of token passing)

#2 - 04/27/2009 03:42 PM - ben leinfelder

i see that it fails until you game the type lattice, but afterward I believe this functions as designed.
The "R working directory" is not really being used like "working directories" might usually be considered, it's more just a hint about where complex data can be serialized. We used to allow (maybe still do?) you to use load() to get workspaces from an upstream R actor available downstream, but this is capital-b Bad.
What I currently see happening with this sample workflow:
-if the R working directories exist, a temporary folder (with timestamp) is created in that folder and the dataframe is serialized there (to be read by the downstream R actor using the information it gets on it's input port).
when the Actors are disconnected, the second actor annot find the irisdf object because it does not share the same workspace (and never will even if the working dirs were the same).

- I may have completely misunderstood the issue here - in which case please reopen the bug.

#3 - 04/28/2009 11:56 AM - Oliver Soong

I was under the impression the R working directory (Rwd) was a parameter to set the working directory of the R actor, for whatever reason the user might desire. I thought it was a redundant convenience feature just like the --save parameter, as the user could certainly change working directories in the actor, just as the user could save.image().

It's not clear to me (in the abstract -- I get the technical reasons) why Kepler's ability to pass data between actors should be dependent on the parameterization of those actors.

All in all, it seems like Kepler already has a place to store temporary stuff (.kepler), R is perfectly capable of accepting full paths to files, workflow execution is not dependent on the user correctly parameterizing the Rwd, and the user is free to muck with the working directory in R (which I just realized will also break workflows).

From the previous comment, it sounds like the actor functions as designed, so I presume there are other reasons I'm not aware of for having this type of working directory that outweigh the above considerations. I'm leaving as resolved.

#4 - 04/28/2009 12:01 PM - ben leinfelder

we might consider removing this parameter since it's not really useful and will loose any utility it provides once we have a good run-management/reporting system in place.

#5 - 03/27/2013 02:25 PM - Redmine Admin

Original Bugzilla ID was 3986