

## Kepler - Bug #4002

### RExpression2 - handle arbitrary R data structures

04/20/2009 05:06 PM - ben leinfelder

<b>Status:</b>	New	<b>Start date:</b>	04/20/2009
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	ben leinfelder	<b>% Done:</b>	0%
<b>Category:</b>	actors	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	3.X.Y	<b>Spent time:</b>	0.00 hour
<b>Bugzilla-Id:</b>	4002		

#### Description

Not everything in R fits into the Ptolemy types (nor the JRI types, for that matter).

The `lm()` method, for instance, returns something special.

It's been proposed (a few times) that there should be an `RObjectToken` to handle such cases so that at least R actors can communicate between each other with out loss/mutation of data.

The `RExpression1` actor used file-based serialization to get around this limitation. We could attempt to use a hybrid approach for `RExpression2` - mixing JRI with file-based serialization (best of both worlds?).

#### History

##### #1 - 04/20/2009 05:07 PM - ben leinfelder

Here is an example script that creates different data objects:

```
x <- 1:10
y <- x + rnorm(10)
f <- y ~ x
l <- lm(f)
s <- summary(l)
```

##### #2 - 06/14/2009 09:14 AM - ben leinfelder

added handling (from the old `RExpression1` implementation) for [un]serializing complex R data objects to disk. This is working for dataframes at the moment. More testing for complex types to come...

##### #3 - 08/18/2009 03:14 PM - ben leinfelder

in the example cases given, all but the structure produced by `~` were transferred from one R actor the the other. This is the debug output:

```
f <- (y ~ x)
```

Result: nil

##### #4 - 08/18/2009 03:18 PM - ben leinfelder

tests the various structures are transferred by the JRI implementation.

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

Original Bugzilla ID was 4002

#### Files

R2-arbitrary.xml	33.9 KB	08/18/2009	ben leinfelder
------------------	---------	------------	----------------