

Kepler - Bug #1543

create workflow scripting language

04/30/2004 10:40 AM - Chad Berkley

Status:	Resolved	Start date:	04/30/2004
Priority:	Low	Due date:	
Assignee:	Chad Berkley	% Done:	0%
Category:	interface	Estimated time:	0.00 hour
Target version:	Unspecified	Spent time:	0.00 hour
Bugzilla-Id:	1543		
Description			
it would be useful to have a simple scripting language to write out workflows without using the visual editor and without having to write a moml file by hand. this could be done using javacc.			

History

#1 - 05/03/2004 02:34 PM - Christopher Brooks

Ptolemy II includes Jacl, which is a 100% Java implementation of a subset of Tcl. We use Jacl for the test suite. Tcl is fairly easy to pick up, and there are plenty of examples about how to run models. There is some documentation about Jacl at

<http://ptolemy.eecs.berkeley.edu/ptolemyII/ptII4.0/ptII4.0-alpha/doc/coding/testing.htm>

Another language to look at is Jython, which is a Java implementation of a subset of Python. There is also a Python script actor that might be useful. See <http://www.jython.org/>

I think Python probably has a larger user community than Tcl these days. Both Jacl and Jython are not really being actively developed, which is good in that they are not wildly changing over time. I've been fixing bugs in the Ptolemy copy of Jacl, and there is also a Sourceforge TclBlend/Jacl project. Last I looked, Jython had not been updated in a couple of years, but we recently recompiled Jython to solve a Java 1.5 runtime problem.

Another language to look at is Caltrop, which is a language that also shipped with Ptolemy II:

"Its research focuses on the definition of CAL, a textual notation for writing dataflow actors, and uses and applications of that language inside and outside of Ptolemy and the Ptolemy II software."

Ptolemy II uses JavaCC in several places, but you might want to look at using CUP and JFlex instead of JavaCC. Basically, using JavaCC as a LL parser can result in ambiguities as a language grows. I'm told, and I vaguely remember, that LALR parsers such as CUP/JFlex tend to have fewer problems with ambiguity.

One trick we use when defining models in Java is to add an exportMoML() call.

For example, `ptolemy/domains/sdf/demo/OrthogonalCom/OrthogonalCom.java` says:

```
// Uncomment the next line dump out xml.  
// System.out.println(exportMoML());
```

One can do similar things in Jacl and presumably Jython as well.

Basically, I encourage you to look at an off the shelf language and see if you can use it. One trick is to define some utility procedures to do common tasks. For example, in Ptolemy II, we have Tcl code in `$PTII/util/testsuite` that defines `sdf` and `de` models. These utility methods are used to set up models for testing.

-Christopher

#2 - 10/12/2004 10:59 AM - Chad Berkley

Since this is already implemented via JACL in PTII, this bug is moot.

#3 - 03/27/2013 02:17 PM - Redmine Admin

Original Bugzilla ID was 1543