

Kepler - Bug #4049

Import actor that already exists appends strange numbering scheme

05/01/2009 03:37 PM - Derik Barseghian

Status:	New	Start date:	05/01/2009
Priority:	Normal	Due date:	
Assignee:	Derik Barseghian	% Done:	0%
Category:	core	Estimated time:	0.00 hour
Target version:	2.X.Y	Spent time:	0.00 hour
Bugzilla-Id:	4049		
Description When I import an actor that already exists, saying Yes to use the next available id, the new actor appends a numbering scheme to the name that I don't understand. After three consecutive imports of the same DataTurbine actor, my tree now contains: DataTurbine DataTurbine_2_ DataTurbine_2__3_ DataTurbine_2__3__4_ I'm assuming that's wrong and should just be one incrementing number.			
Related issues: Blocks Kepler - Bug #4150: Save in Library broken			
			Resolved 06/12/2009

History

#1 - 05/04/2009 04:07 PM - Christopher Brooks

I'm not sure how to reproduce this.
Doing Tools -> Instantiate Component over and over again creates
Ramp
Ramp2
Ramp3

I must be missing something?

#2 - 05/04/2009 04:15 PM - Derik Barseghian

I should've been more clear, this is using the Kepler File => Import Archive (KAR) menu item and then looking at the resulting updated library tree, not Instantiate Components.

#3 - 05/04/2009 07:12 PM - Derik Barseghian

I've checked in a fix, but am leaving this open and changing to a feature request -- see if there's a better way to go about this. When a user instantiates multiple copies of the same actor, kepler increments the number on the name of the actor on the canvas. Things can get confusing when looking back and forth between the canvas and actor tree.

#4 - 05/20/2009 11:10 AM - Derik Barseghian

Soong notes that when dragging an actor with name ending in a digit(s) to canvas, the digits are dropped.

#5 - 05/20/2009 02:05 PM - Derik Barseghian

Re: comment#4, for example, if an actor RExpressionRain2001 is created and saved in the library, when dragged to canvas, it becomes RExpressionRain. One reason is that NamedObj's uniqueName method strips the numeric suffix. I've made a change to NamedObj locally to not do this, to instead optionally append an underscore followed by a unique number. If it's the first instance, it will be RExpressionRain2001. The second would be RExpressionRain2001_2. However, the first instance still gets its numeric suffix (2001) stripped by the time the actor hits the canvas, because MoMLParser's attribute calls CompositeEntity's uniqueName().

I see 5 declarations of uniqueName(), and many calls to them. I'm a little reluctant to make changes, it seems like there could be many ramifications...

One slightly hackish solution is to have the first instance of e.g. RExpressionRain2001 dragged to canvas be called RExpressionRain2001_. This seems to work, but isn't very satisfying.

#6 - 11/06/2012 08:33 PM - Christopher Brooks

See also https://chess.eecs.berkeley.edu/bugzilla/show_bug.cgi?id=61 and maybe http://bugzilla.ecoinformatics.org/show_bug.cgi?id=5726

#7 - 03/27/2013 02:25 PM - Redmine Admin

Original Bugzilla ID was 4049

#8 - 07/03/2013 04:56 PM - Daniel Crawl

To prevent the ending digits from being removed when dropping an actor on the canvas, set the display name and make the display name and name different.