

Kepler - Bug #4319

LibraryManager

08/17/2009 11:52 AM - Aaron Aaron

Status:	Resolved	Start date:	08/17/2009
Priority:	Immediate	Due date:	
Assignee:	Aaron Aaron	% Done:	0%
Category:	core	Estimated time:	0.00 hour
Target version:	2.X.Y	Spent time:	0.00 hour
Bugzilla-Id:	4319		

Description

The Component library was being built with an Index which made it difficult to dynamically add and remove components from the library. The Index does speed up Kepler startup by quite a bit though so we don't want to get rid of the Index (although it would greatly simplify the Library to not have to maintain the index). So the goal here was to split up what was previously just the LibraryIndex class into several classes that have different functions. These classes are now, SemanticTypeIndex, LibraryManager, LibraryIndex and LibraryIndexItem and it's subclasses. It may also be useful to further split up LibraryManager into two classes called LibraryIndexer and LibraryManager, where LibraryManager is responsible for dynamic changes to the library during runtime operations and the LibraryIndexer class is responsible for improving performance by syncing and building the Library on startup and shutdown of the application using the serialized LibraryIndex object.

History

#1 - 08/18/2009 07:47 AM - Christopher Brooks

This looks interesting, though it seems to be a feature request?
There could be a bug here, it should be stated more clearly if there is a bug.

#2 - 08/18/2009 12:22 PM - Aaron Aaron

The bug here was just poor API design (or more likely an unfinished API). The library was dependent solely on the library index for modification which meant that to change the library one needed to have intimate knowledge of how the index worked and had to include index specific code in their functions to get the library to do anything. A refactoring into multiple classes and the inclusion of methods in the LibraryManager for modification of the library that instead manages the index after the fact (instead of requiring the developer to do it before hand) is the main goal of this enhancement. To see this compare the two versions of the addKAR method in LibraryManager. One takes only a file and updates the index as it goes, whereas the other requires a pre-built hierarchical LibraryIndexItem to be passed to it. Separation of the code also gives the opportunity for substituting other mechanisms for managing the index (such as using sql or java serialization instead of xml).

#3 - 08/18/2009 04:17 PM - Christopher Brooks

I could easily be wrong about this and it could be a bug.
However, it is worded like an enhancement. To me, a bug is:
"When I do this, I expected some action, but got something else."

An enhancement is a new feature request that describes the requirement and may propose a possible solution. There is a grey area between bugs and enhancements.

Feel free to change it back to a bug if you'd like, I don't mind.

#4 - 11/24/2009 09:40 AM - Aaron Aaron

Building the library has been enhanced and now consists of 4 classes. LibraryManager, LibraryGenerator, LibIndex, and LibItem. These classes maintain an index in the SQL database. Closing this bug, new bugs can be opened for any specific bugs that crop up due to this change.

#5 - 03/27/2013 02:26 PM - Redmine Admin

Original Bugzilla ID was 4319