Morpho - Bug #1131

closing datapackage windows does not release memory

08/15/2003 07:51 AM - Dan Higgins

Status:	Resolved		**Start date:**	08/15/2003
Priority:	Normal		**Due date:**	
Assignee:	Dan Higgins		**% Done:**	0%
Category:	morpho - general		**Estimated time:**	0.00 hour
Target version:	1.5		**Spent time:**	0.00 hour
Bugzilla-Id:	1131			

Description

Creating a new datapackage display window can use a lot of memory. Observation of memory usage indicates that the JRE memory used when the window is created does not seem to be released when the window is closed. This may cause out-of-memory problems as many windows are opened and closed.

History

#1 - 08/16/2003 02:05 PM - Dan Higgins

Note that the Window.dispose() method was used when a window was closed to recover memory. However, the documentation for dispose says

"Releases all of the native screen resources used by this Window, its subcomponents, and all of its owned children. That is, the resources for these Components will be destroyed, any memory they consume will be returned to the OS, and they will be marked as undisplayable.

The Window and its subcomponents can be made displayable again by rebuilding the native resources with a subsequent call to pack or show. The states of the recreated Window and its subcomponents will be identical to the states of these objects at the point where the Window was disposed (not accounting for additional modifications between those actions). "

Note that only 'native screen resources' is recovered. It appears that various java objects associated with a window (e.g a treesub panels, trees, etc.) are NOT destroyed and their memory not automatically recovered! (or they could not be recovered with a call to pack or show).

#2 - 10/07/2003 11:45 AM - Dan Higgins

Several changes made to 'force' the release of memory when a window is closed. These seem to have solved the memory leak problems noted here.

#3 - 03/27/2013 02:16 PM - Redmine Admin

Original Bugzilla ID was 1131