Semtools - Bug #4836

Column Annotation page overwrites changes made in Full Annotation

02/24/2010 02:02 PM - ben leinfelder

Status:	Resolved		**Start date:**	02/24/2010
Priority:	Normal		**Due date:**	
Assignee:	ben leinfelder		**% Done:**	0%
Category:	Morpho Plugin		**Estimated time:**	0.00 hour
Target version:	Unspecified		**Spent time:**	0.00 hour
Bugzilla-Id:	4836			

Description

Because of how I've set up the listener on the madlib tab, if you make a change directly to the annotation table, then click on a column, you'll lose the change because the (hidden) madlib page doesn't have that change and it tries to save it's state whenever there's a selection change event (column or tab).

I knew it was less than ideal to trigger the annotation page to save it's state this way, but I'm not sure where exactly it should be. Perhaps I should be checking what tab is visible when the column selection changes and basing the save on this...

Related issues:

Blocked by Semtools - Bug #4819: "Column Annotation" view duplicates mappings	**Resolved**	**02/22/2010**

History

#1 - 02/24/2010 02:08 PM - Matt Jones

Alternatively, create StateChangeEvents that reflect the change regardless of where it happens, so that updates to the annotation model can be reflected in both the tabular and the madlib views.

#2 - 02/24/2010 03:16 PM - ben leinfelder

here's my thinking:
-there should be a [rather lightweight] class that holds the tabbed pane
-this AnnotationTabPane keeps track of which tab is currently active
-if the Column Annotation tab is active and a column selection change event
occurs, the current state of the Column Annotation is saved before anything
else happens.
-if the tab is not active, then nothing needs to be saved.
-if the tab was just active (and is being navigated away from) then the
annotation needs to be saved.

*I use "save" in the temporary sense, not in the "to disk" sense.

#3 - 02/25/2010 09:42 AM - ben leinfelder

this is working now. I've restricted how you can use the AnnotationPage's methods so as to avoid duplicating mappings.
The AnnotationTabPane is now responsible for keeping track of the selected state of the tabs and saving when need be. More importantly it keeps the tabs in synch with the latest state of the annotation as it is modified through various parts of the UI.

#4 - 02/25/2010 09:57 AM - ben leinfelder

and it's back

#5 - 02/25/2010 10:51 AM - ben leinfelder

-added logging
-use member variable for annotation page reference (used in event handling methods)
-reset the annotation page values when the appropriate column/tab selection events occur (this was not always happening, hence the duplication bug)

i'm pretty sure this won't be reopened...

#6 - 03/27/2013 02:28 PM - Redmine Admin

Original Bugzilla ID was 4836