

Semtools - Bug #4856

Ontology Browser as a non-modal palette

03/01/2010 09:42 AM - ben leinfelder

Status:	New	Start date:	03/01/2010
Priority:	Normal	Due date:	
Assignee:	ben leinfelder	% Done:	0%
Category:	Morpho Plugin	Estimated time:	0.00 hour
Target version:	morpho-plugin-0.9.0	Spent time:	0.00 hour
Bugzilla-Id:	4856		
Description			
From Shawn: --- 5. I still think we should display the current drop-down term selection as a panel in the right-hand side portion of the editor (where the plain old metadata is viewed now). There are multiple reasons for this: (a) currently when it is viewed as a drop down it hides the table (i.e., what we are trying to annotate); (b) it prohibits any kind of static interaction with the component, e.g., i can't search it or look through it without performing an annotation; and (c) I think we could add more features to the panel to make it easier to search and see search results if it were a static widget on the canvas, as opposed to a drop down. I think we could make this context sensitive, but not "focus stealing". So, e.g., if I were to click on the <characteristic> in the madlib, i'd only see active in the "ontology palette" the characteristic portion.			

History

#1 - 03/01/2010 02:36 PM - ben leinfelder

Use Proposals:

1. Drag and Drop
 - class can be selected from the tree and dragged to a:
 - madlib field (Column Annotation tab)
 - annotation table cell (Full Annotation tab)
 - class cannot be dropped into an incompatible field (Entity subclass into a MeasurementStandard subclass, for instance)
2. Select-then-browse
 - click on a [madlib/table] field to select it
 - browse the ontology tree to choose the class you want for that selected field
 - class in the field changes each time you choose a different class in the tree
 - navigating away from the [madlib/table] field saves whatever class was last chosen
3. Browse-then-select
 - browse to your heart's content in the tree
 - click on the [madlib/table] field you want to apply the currently selected class to
 - will only set the class if it makes sense (subclass of the proper OBOE class)
 - be careful where you click next?

#2 - 03/04/2010 03:26 PM - ben leinfelder

IRC conversation:

bowers: what if instead of or in addition to drag and drop, it worked more palette like
[12:27pm] bowers: in the sense that
[12:27pm] bowers: when you click in a box, e.g., in a mad lib
[12:28pm] bowers: when you select a concept in the tree, it fills in the box
[12:30pm] benMac: then you'd just have to be sure to "un click" on the madlib field when you were done so as not to inadvertently change the class as you navigate in the ontology browser
[12:31pm] benMac: there's no - and forgive the Harry Potter reference - "mischief managed" button
[12:35pm] bowers: yeah -- another way to do it would be to add a button in the palette called "set class" or "set term"
[12:36pm] bowers: then you click the in a box, select a term, then click "set term"
[12:36pm] bowers: this would seem to solve many of the issues you are raising (which are good ones)
[12:39pm] bowers: another extension to this would be to also have a simpler drop down box for the madlibs

[12:39pm] bowers: this one would react to typing in some terms

[12:39pm] bowers: like gmail style

[12:40pm] bowers: in fact, gmail has both: a way to drag and drop tags and a way to do tags drop-down style

[12:44pm] benMac: how much simpler of a drop down? flat?

[1:18pm] bowers: yeah, flat list w/ a scroll bar -- so it is of a fixed size

#3 - 04/19/2010 01:46 PM - ben leinfelder

We are newly-recommitted to this approach - it will replace the current "drop down" style ontology browser. The state of the palette will be determined by the focus on/off different class selection widgets. It will remain drag-able (the classes) and can be repositioned as needed. Initially it will be docked as the current browser is (in a tab), but will soon be free and floating about.

#4 - 04/19/2010 05:18 PM - ben leinfelder

Almost there with the floating dialog palette...looking good with preliminary work at least.

A few issues/todos:

- need to differentiate between the "last" position that the palette appeared because it was moved there by a user's action or simply because it popped up under a different selection field. We want to "know" when they've moved it vs. not.
- close operation on the dialog window should have the same behavior as hiding the popup with an ESC key press
- since this is a non-modal dialog it ends up going *behind* the modal search window and is completely inactive. So it appears that you can't have a non-modal dialog as a child of a modal dialog? Maybe I should leave the old drop-down style browser for the search window? Maybe there is a way to have the mixed modal windows play nice. More investigation/thought is needed. The search isn't the only place where these class selection widgets (and popups) are used. The "annotate current column..." interface also has this modal issue.

#5 - 04/20/2010 11:15 AM - ben leinfelder

1. modal issue resolved: made those modal dialogs non-modal so that the palette is not inactivated.
2. window closing event handled like ESC key
3. "remembering" last palette position if it is moved

#6 - 04/20/2010 11:22 AM - ben leinfelder

The only other thing I can think to add to this is that it not close unless you hit the close button on the window. Currently it will close when you double click a class to select it and set it in the field that opened it, or when you click on a field (focus change). It also never shows all the classes in the loaded ontologies (like the browser in the docked tab does).

#7 - 04/26/2010 08:10 AM - ben leinfelder

from shawn:

- do not close/reopen palette, just refresh the contents
- include search field in palette window
- include "Select" button in palette (same as double clicking class in field)

#8 - 04/26/2010 11:34 AM - ben leinfelder

Now the palette remains open. When a different field is clicked, the contents change to match (filter class and selected class). The search field is on the browser - it is not forward looking and works much better (conveniently responds to hitting "enter" to initiate the search). There's also a "Select" button on the bottom of the palette that will "move" the selected class to the field being edited (same as double clicking or clicking and dragging).

More enhancements?

#9 - 04/27/2010 11:59 AM - ben leinfelder

I've noticed it's quite easy to "forget" which field is driving the ontology palette - perhaps I'll had some clear visual clues (bold border?) to the field when it is "connected" to the palette.

#10 - 04/27/2010 02:15 PM - ben leinfelder

there's now a blue border around the "active" field that will disappear when a different field is selected or if the palette is closed

#11 - 03/27/2013 02:28 PM - Redmine Admin

Original Bugzilla ID was 4856

#12 - 11/26/2013 08:53 AM - ben leinfelder

- Target version changed from Unspecified to morpho-plugin-0.9.0