Setting up Change Management

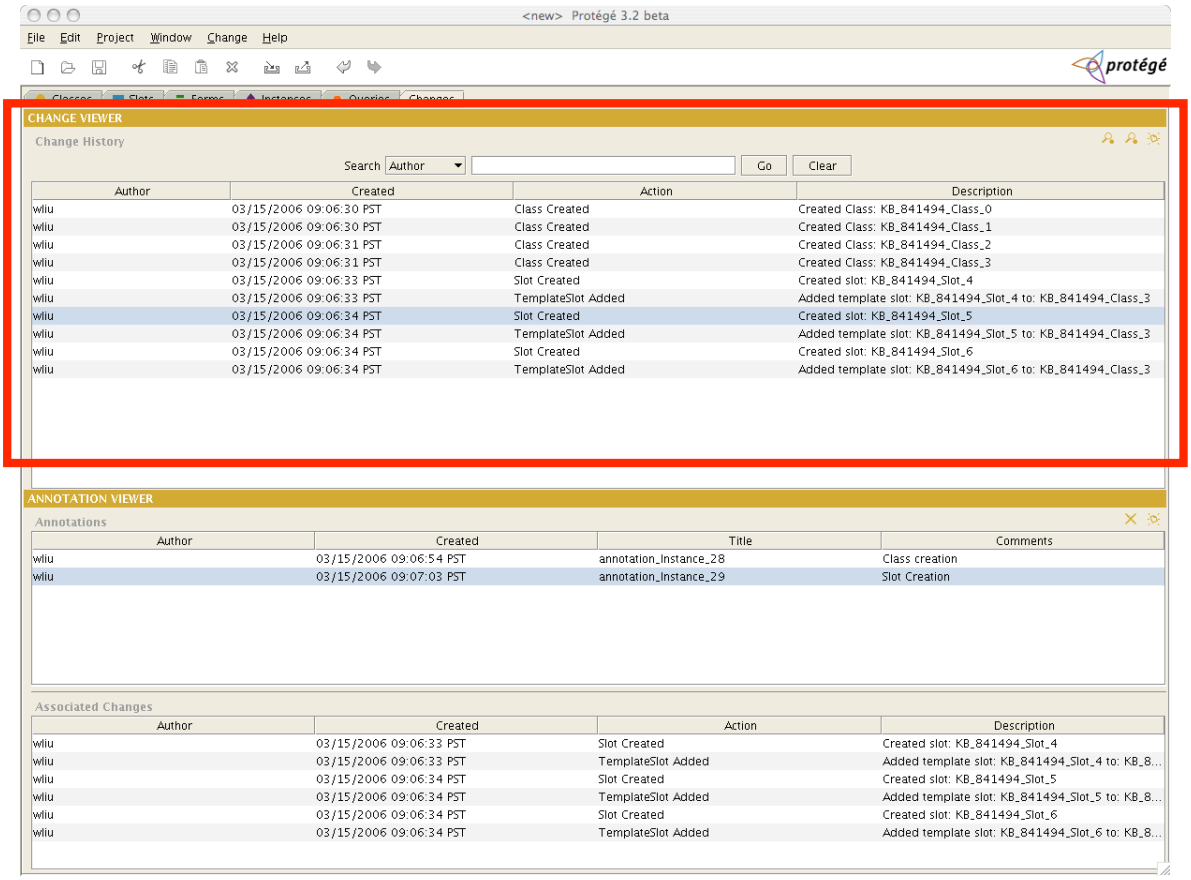
Activate:

To activate the Change Management tab for your given project, ensure you configure your project properly by doing the following in the menu bar:

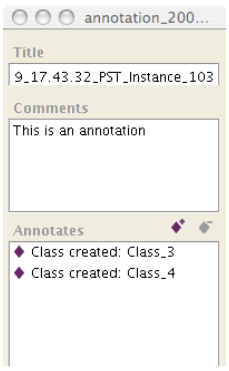
`Project > Configure >` Select the `'ChangesTab'` option.

Using Change Management

Change Viewer:



The change viewer shows what changes the user has made to the given project they are working on. Each entry in the table is a particular change. To create an annotation of a change (or a group of changes), simply select the given change(s) and select the `'Create Annotation'` icon:

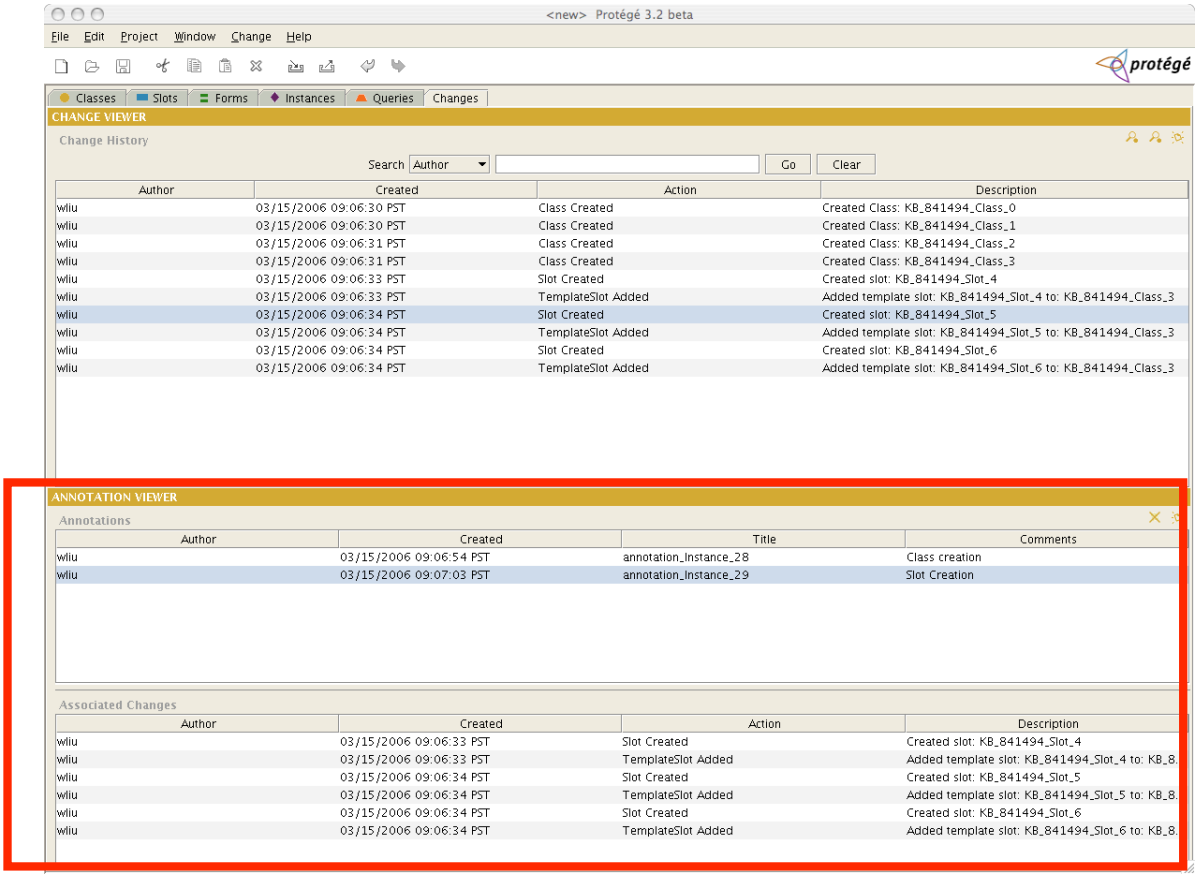


This will bring up a window where you can create the annotation for the given change(s). Simply type the contents of the annotation in the `'Comments'` text area and close the window.

The two other icons in the `'Change Viewer'` header are used to filter the changes that are displayed. There is a "Summary View" as well as a "Detailed View".

The search mechanism works like the other search mechanism built within Protégé. The keyword that is typed is in the form of a regular expression (please also note that the type case does matter).

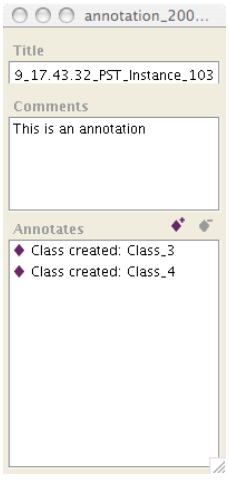
Annotation Viewer:



The 'Annotation Viewer' shows the user generated annotations. The upper table shows the actual annotations. The lower table shows the associated changes for a given annotation. To see the given changes for a given annotation, simply select the desired annotation.

The user can also edit a given annotation. Simply click on the 'Edit Annotation' icon:

This will bring up the corresponding annotation window (make the necessary changes within this window):



The user can also remove given annotations. To do this, simply select the annotation in the table and click the 'Remove Annotation' icon:

Miscellaneous:

Within Change Management, the user can also make quick changes. If they user is in another tab, once a change has been made, they can use the keystroke command: CTRL+A to bring up the annotation window, or they can go to the menu bar and select the Change item.

Also, the user can get Change or Annotation information about a particular class within the "Classes" tab by selecting the given class, then going to the Change menu (from the menu bar) and selecting: 'Selected Change Info'.

Server-Client Mode:

The changes tab can also be configured to work in server-client mode. In order to do this two things must be configured:

- The class path must be set properly on the client. In an upcoming release, this change to the class path will already be done.
- The project being monitored for changes must have the changes tab enabled and the changes annotation project saved.
- The server metaproject file needs to be configured properly to indicate that the server is providing the changes annotation project to the clients and to define the relationship between the project being edited and the annotation project.

If the class path is not correctly set then the client will see a StreamCorruptedException when trying to access the server. To avoid this exception the user needs to add the changes_model.jar in the change management plugin directory to the class path.

Setting the classpath is done differently on different platforms. On a windows machine, the user can simply edit the Protégé.lax file and add the

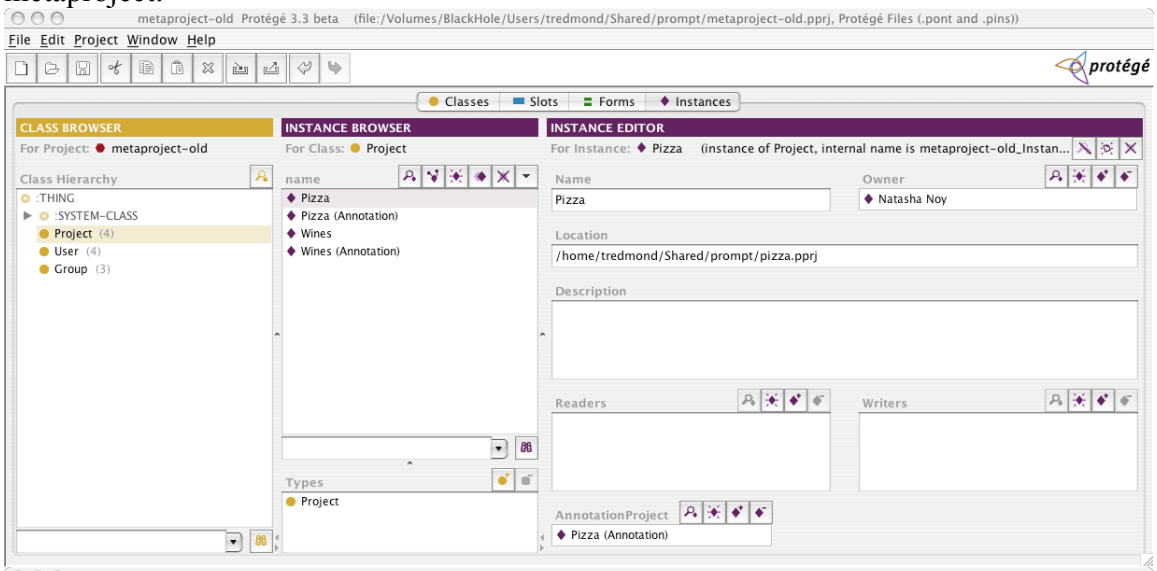
"plugins/edu.stanford.smi.protegex.changes/changes_model.jar"

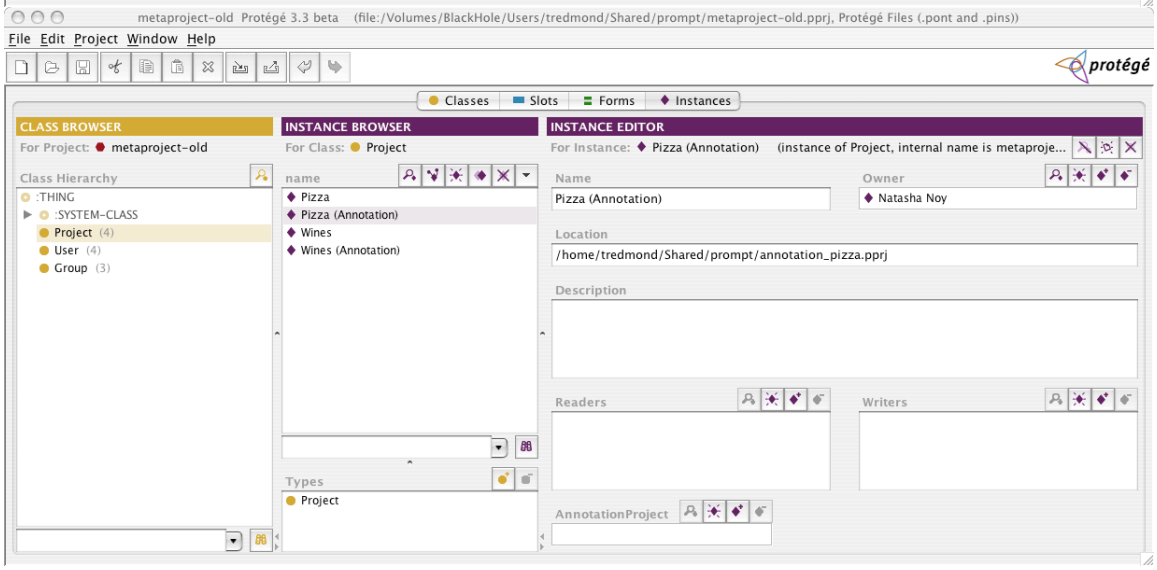
to the classpath. On an "OS X" machine, the user right clicks on the Protégé executable and then double clicks "Show Package Contents". A new finder window will open up and click on Contents and then edit the Info.plist file in that directory. Preferably edit the Info.plist file with the Property list editor but a text editor will also work if you don't have access to the Property list editor.

The metaproject must be set to provide both the project being monitored for changes and the changes annotation project. So for instance, suppose that the original project is the pizza project. If

1. the pizza project is edited in standalone mode,
2. the changes tab is configured and
3. the pizza project is saved

then a new project called annotation_pizza will be created in the same directory as the pizza project. The metaproject that serves the pizza project should also serve the annotation pizza project. In addition the metaproject must have a slot annotationProject, from Project to Project, which takes the pizza project to the annotation pizza project. As a result of these steps you should see the following displays when editing the server metaproject.





Now running the server and connecting with the client will result in a client with access to the changes tab.