

Viewing and Editing the Ontology in Protege

This section contains specific instructions for eagle-i editors who are working with the ontology in Protégé, as well as an overview of the general workflow for various common editing tasks.

Protégé is a free, open source ontology editor and knowledge-base framework. While this document contains a few general instructions for using Protégé, it is not meant to serve as a comprehensive help document for that program. Consult the Protégé user documentation for more details: http://protegewiki.stanford.edu/wiki/Main_Page.

Setting up Protégé

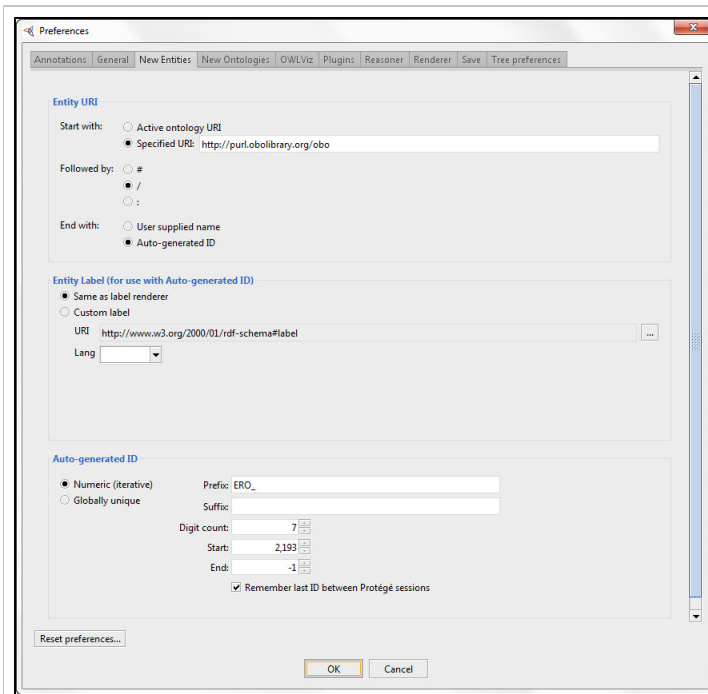
Step 1: Download Protégé

- Download OBI-friendly Protégé 4.3 or 5: <http://protege.stanford.edu/>
- If prompted, do not install updates for the plugins, as the plugin versions contained in the download work specifically with OBI.

Step 2: Configure Protégé

If you are editing for the first time in Protégé, you will need to adjust the settings (should only need to be done once).

- Start Protégé and open an ontology. Any ontology will work. A dialog may appear asking you about importing OWL files; if so, click Cancel.
- Open the **Save** tab under **Preferences** and ensure 'Use XML Entities' is **NOT** checked.
- Go to: **Preferences > New Entities** and set up your new entity URI settings as shown below:



Entity URI

- Start with:** select 'Specified URL' and enter '<http://purl.obolibrary.org/obo>'
- Followed by:** select '/'
- End with:** select 'Auto-generated ID'

Entity Label (for use with Auto-generated ID)

- Select 'Same as label renderer'

Auto-generated ID

- Select 'Numeric (iterative)'
- Prefix:** enter 'ERO_'
- Digit count:** enter '7'
- Start:** enter a four digit number one higher than that of the last ID created. To determine correct number:
 - consult the **IDs.xlsx** spreadsheet, located in **trunk > docs** to find the last ID number used
 - search for the next highest ID number in Protégé to double check no additional IDs have been created
- End:** -1

Step 4: Reasoning

The reasoner tests the logic and consistency of the asserted axioms in the ontology and computes the inferred hierarchy. The suggested reasoner for eagle-i is FaCT++. It is good practice to run the reasoner periodically before committing, to ensure nothing has been accidentally moved. The 'Class hierarchy' tab shows only the asserted hierarchy; the 'Class hierarchy (inferred)' tab shows the inferred hierarchy relationships (i.e. the hierarchy seen in the software). The inferred hierarchy is only visible after the reasoner has been run in that file.

Viewing the eagle-i ontology

Before viewing or editing the ontology, make sure you're looking at **the most current version**:

- Update the ontology trunk in your Subversion repository to get the latest changes: <https://open.med.harvard.edu/svn/eagle-i-dev/datamodel/trunk>.
- Use the module scripts to generate the local ontology
 - Right click on the `src\isf\module-scripts` folder and select 'Open in Terminal' to open a command window. (Note: these instructions are specific to the SmartSVN client, but a variation on this workflow should be possible from other clients. See [Ontology File Structure > src/isf/module-scripts](#) for more details.)

- b. Type `ei-all.bat` (for Windows) and hit enter. The script is done when the command prompt reappears. This process will take some time and potentially slow your computer down.
3. Depending on what part of the ontology you wish to view, open one of the following OWL files in Protege:
 - a. `eaglei-modules-top.owl` — imports the source (the ISF), the module configuration, and the exclude and include files.
 - b. `eaglei-module-configuration.owl` — shows only the source (ISF) and the module configuration.
 - c. `eagle-i-extended-app.owl` — shows both the eagle-i ontology generated and application OWL files.

Reminder: **do not edit the generated files**. Any direct edits of the generated files will not be committed to SVN (these files are SVN ignored) and they will be lost the next time the scripts are ran. Only view these files in Protégé:

- Select **Show all loaded ontologies** under View settings to show terms from all files. Terms in **bold** are contained in the current active file. Hover over the name of unbolded terms to see their actual location.
- **Show the imports closure of the active ontology** shows terms both directly and indirectly imported into that file.
- **Show only the active ontology** shows only terms contained in the active file; no imports.

Next: [General Editing Instructions](#)