OBO Explorer: an editor for open biomedical ontologies in OWL
Stuart Aitken1,*, Yin Chen1 and Jonathan Bard2
1School of Informatics, University of Edinburgh, 11 Crichton Street, Edinburgh EH8 9LE and 2School of Biomedical Sciences, University of Edinburgh, 1 George Square, Edinburgh EH8 9JZ, UK
Received on August 20, 2007; revised on November 12, 2007; accepted on November 27, 2007
Advance Access publication December 1, 2007
Associate Editor: Dmitrij Frishman

ABSTRACT
Motivation: To clarify the semantics, and take advantage of tools and algorithms developed for the Semantic Web, a mapping from the Open Biomedical Ontologies (OBO) format to the Web Ontology Language (OWL) has been established. We present an ontology editor that allows end users to work directly with this OWL representation of OBO format ontologies.
Availability: http://www.aiai.ed.ac.uk/project/cobra-ct
Contact: stuart@aiai.ed.ac.uk

1 INTRODUCTION
Inquires into new subject domains, into the conceptual basis and into the curation of biomedical ontologies have led to the evolution of modelling and language standards: The original Gene Ontology flat-file format has been replaced by the Open Biomedical Ontologies (OBO) format, itself having several versions, and now a mapping from OBO to the Web Ontology Language (OWL) has been established. The formality of OWL brings numerous benefits, including access to the logics, reasoning algorithms and tools of the Semantic Web, but raises the issue of correctly and adequately capturing the content of a concept definition. In many cases, the current OBO definitions of biological concepts lack information essential to a formal OWL definition. To remedy this problem by enabling end users to work with bio-ontologies in OWL, new tools are needed. We present the OBO Explorer, an editor for OBO ontologies in OWL.

A representation for OBO ontologies in OWL has been agreed (NCBO, 2007) and a tool for automatically converting ontologies from OBO to OWL and for reading OBO ontologies into the Protégé 3 ontology editor (http://protege.stanford.edu) has recently been developed (Moreira and Musen, 2007). OBO-in-OWL succeeds in capturing all of the content of OBO ontologies, both the logical structure and the informal annotations, e.g. synonyms and database cross-references (DbXRefs). The annotations are important as they capture all the information that cannot be stated in the logical structure of the ontology, and, of course, we want users to be able to edit all aspects of a term’s definition. However, Protégé 3 is unable to display the annotations associated with OBO terms using the default interface configuration, and therefore the user cannot edit this information. Hence, we implemented the OBO Explorer. This tool is distributed as a Protégé tab, contributing to the large user community that supports Protégé development.

2 THE OBO EXPLORER PROTÉGÉ TAB
The OBO Explorer interface is implemented as a tab that presents the class hierarchy on the left hand panel, and term annotations on the right, as shown in Figure 1. The titles of the boxes within the main panel correspond to the names of the corresponding OBO annotations. Where possible, the user interface components are present on the main panel, and immediately update the underlying OWL model. The synonym, subset and DbXRef information is displayed in list form in a concise manner to enable users to see all this information in one place. These data are edited by calling up dialogs that allow new items to be added and existing items to be deleted from the lists. A second tab, of similar design, allows the user to edit the metadata associated with the OBO ontology. For example, new subsets and synonym types can be created.

When an OWL ontology is newly created in Protégé, it will lack the agreed OBO-in-OWL classes and relationships needed to represent OBO terms. In this case, the tab creates the appropriate definitions for these elements—hiding the underlying details of the OWL representation from the user.

It is important to investigate whether the translation from OBO to OWL causes confusion to users, for example, between OBO term names/IDs and OWL URIRefs. On translation to OWL, the unique term ID (for example, GO:0002516) becomes the local name in the URI, and will be used as the label for the concept in Protégé’s display (in the default configuration). However, the user will expect to see the term name ‘B cell deletion’. The OBO Explorer has a feature to cause the name to be displayed with the term ID as a postfix (shown in Figure 1), and we are interested to investigate any problems in the use of URIs and the usability of the features provided.

Additionally, as part-of relations no longer appear as edges in the ontology graph, but now serve to specify the term as a subclass of an anonymous class (e.g. as a subclass of the set of things that are part-of some B cell tolerance induction) it is necessary to show this information to the user. The insert in Figure 1 allows the user to see and to edit part-of restrictions...
that define the currently-selected class. In the lower panel, the user can see a list of classes that are defined in terms of the currently-selected class B cell deletion (these do not define B cell deletion, but are references to it).

A potentially more significant change that the user will observe is the displacement of all terms that have no is-a definition (in the original OBO) to the top level of the OWL ontology. For ontologies that are is-a complete such as the Gene Ontology (which completed the process of assigning is-a parents to all terms in January 2007), there will be no change in the taxonomic structure. But for other ontologies, and for the OBO anatomies in particular, the user will see that the taxonomy is deficient. The combination of Protégé and the plug-in described here allows this problem to be solved, but, in certain cases, significant manual effort will be required to complete the transition of an ontology from OBO to the more formal OWL representation.

A simple task-based evaluation of the OBO Explorer is underway. Users are asked to install the tool and perform a number of searching and editing operations. The trial addresses the installation and configuration tasks as these involve navigating the numerous dialogs that Protégé users must complete when opening an ontology and adding tabs to the interface. The results will indicate whether the OBO Explorer should be packaged such that these steps are avoided. We do not compare the new tab with the existing UI as the tasks could not be performed with these components by the user group.

Initial results suggest that configuring Protégé to show the required tabs is time consuming for users (and barrier for some). The OBO Explorer tab follows the Protégé interface style where changes to text fields are confirmed by typing return, however, this was noted as being inconvenient. The procedure for generating new term IDs was not sufficiently clear. Overall, users found the tab’s documentation clear, and completed the tasks successfully.

3 RELATED WORK

OBO-Edit (Day-Richter, 2007), a bio-ontology editor supported by the Gene Ontology consortium, now has an OWL import/export function but does not provide access to an underlying OWL representation. Protégé and Swoop (http://code.google.com/p/swoop) are generic OWL editors that can read, write and edit OWL ontologies but do not support OBO in OWL specifically and so leave the user to specify all the representational details.

4 CONCLUSIONS

We have described a graphical editor for bio-ontologies that makes use of the Protégé platform and the OBO to OWL Converter tab (Moreira and Musen, 2007). Together, these components allow the user to work with Open Biomedical Ontologies in OWL, and to access many other tools developed for Protégé, for example, graph viewers and Description Logic reasoners. This tool is intended to be used with an ontology management server (Aitken et al., 2007) that supports version management and publication.

ACKNOWLEDGEMENT

This work was funded by BBSRC grant BB/D006473/1.

Conflict of Interest: none declared.

REFERENCES