Evaluating Protégé OWL as an Editing Environment for NCI Thesaurus

Robert Lintern*, Ray Fergerson#, Natasha Noy#, Holger Knublauch#, Tracy Safran^, Gilberto Fragoso^, Sherri de Coronado^.

***The CHISEL Group, University of Victoria; #Stanford Medical Informatics, Stanford University; ^NCI Center for Bioinformatics, National Cancer Institute.**

The National Cancer Institute has developed the NCI Thesaurus, a biomedical vocabulary that provides consistent, unambiguous codes and definitions for concepts used in cancer research. It currently has about 39,000 concepts, in 20 domain hierarchies. A lexical component provides human usable definition and other information. Description logic is used to ensure complete, consistent and non-redundant definitions in the formal sense. NCI Thesaurus enables retrieval of information across a wide range of domains related to cancer research, with one goal being facilitating translational research. This vocabulary was originally built using Ontylog, a description logic developed explicitly for building large complex terminologies by Apelon, and implemented in the Apelon Terminology Development Environment (TDE). We have converted the Ontylog version of NCI Thesaurus to OWL, and a team from the University of Victoria and NCI have been testing the features and performance of Protege OWL in a multi-user environment with a database backend to determine whether it could be used to support the operational needs of NCI Thesaurus development and maintenance. In response to this testing, the Protégé team has been making changes (largely to the multiuser server version) to enhance performance.

NCI Thesaurus Operational Requirements:

The NCI Thesaurus provides vocabulary support to the NCICB bioinformatics infrastructure that includes objects and data standards as well as vocabulary, and is used directly and indirectly by a number of applications within and outside of NCI. It operates on a monthly publication schedule: editing of distinct schemas by multiple domain experts is consolidated in one database, published to a test database and then promoted to production. The vocabulary is edited full time by a group of editors primarily at NCI, but also around the country. Currently, each editor has a separate schema that resides on a central server. Edits are made to the individual copies. Each editor performs classification, and then exports a change set. The Work Flow Manager combines the changes from the separate schemas with the assistance of a work flow tool, deals with conflicts, and produces a new baseline. The editing must be halted from export of the change sets until the new baseline is released.

Potential Advantages of Protégé OWL Multiuser Version:

Using the multiuser version of Protégé OWL as an editing environment has the potential to provide a number of advantages over our current environment. First, the semantics of OWL are publicly known and the software is open source, so researchers could contribute to the NCI Thesaurus more readily; similarly, the NCI Thesaurus could refer to other OWL reference terminologies. Second, it would provide additional semantics not available in our current environment. Third, it would enable our editors to work simultaneously on the same database, without having to wait for weekly cycles of editing, facilitating real time collaboration. Finally, a workflow system that better integrates concept history and production releases would be a big plus.

Summary of Protégé Enhancements Identified from Pilot Editing Tests

We are conducting a series of pilot projects with the goal of identifying and solving performance and user issues for Protege to become a suitable tool in the NCI EVS production environment. The NCI Thesaurus OWL editing project is a) the largest OWL project using Protégé, b) the only OWL project using multi-user Protégé, and c) the first (or at least the largest) OWL project seriously using the database backend. These pilots are conducted in the multi-user server

version with a database backend, with testers located in the U.S. and Canada and the server located in Maryland. Among the items that have been solved or identified so far:

- Editors inside and outside firewalls had required different Protege client configurations; a single configuration has now been identified that works in both cases
- Performance in general has been improved by retrieving the minimal set of OWL classes at startup required to populate various widgets; additional classes are retrieved on demand
- Performance of navigation in the Protege class tree browser has been improved by eliminating lookups of subclasses when rendering the nodes in the tree widget
- Frame caching observed when navigating the class hierarchy in the OWL plugin (and under other routine editing actions) has been minimized.
- The number of queries sent to the database by Protégé/OWL during classification by Racer has been reduced; updates to the database are done in a single batch statement
- Enhancement of Prompt allowing users to declare specific properties for comparisons utilizing codes or identifiers
- Generation and publication of concept history following baseline comparisons by Prompt
- The utilization of Protégé's journaling files by Prompt to identify specific editing changes and ascribe these changes to editors during history processing
- User-defined enable/disable configurability of drag-and-drop actions
- Provision for additional configurability of allowed/disallowed editing actions in the OWL plugin, e.g. disallow enumerating classes, and for finer grained control over editor's privileges
- Simplification of the Prompt comparison routines when it is utilized for change and conflict detection in an evolving baseline (in progress)
- Improvement of the Protégé search facilities; e.g. soundex and other lexical techniques (in progress)

These small and large-scale pilots have helped identify the major functional enhancements required by the tool in order to serve NCI's needs. A major issue has been the performance of Protégé in multi-user mode over the internet, which has been improved dramatically over the testing period. The workflow issues are currently being examined, and a potential solution is being evaluated in additional pilot tests. These future tests will include an examination of new functionality introduced in the NCIOWLClasses tab, which enables and tracks merges, splits, and retirement of classes, provides for code generation, and adds utilities for batch loading and concept reports.