Alternative Interfaces for OWL Ontologies
Alan L Rector, Nick Drummond, Matthew Horridge, Hai H. Wang and Julian Seidenberg
Department of Computer Science, University of Manchester, M13 9PL UK

Abstract

This poster presents a selection of ontology editing tools that have been developed for Protégé-OWL. The tools were developed in the context of the CO-ODE project\(^1\). Although Protégé-OWL\(^2\) is arguably the most widely used tool for editing OWL ontologies, the design and implementation of the standard Protégé-OWL user interface has been focused around satisfying the needs of logicians and knowledge modellers. The tools that are presented in this poster are designed to be used by domain experts who are generally not logic savvy. It is hoped that these tools will encourage e-Scientists to take up OWL for use in their service architectures.

Ontologies and metadata are key to knowledge management and service architectures for e-Science. They are critical for annotating resources so that they can be discovered and used by semantic middleware in service oriented architectures and workflows. In particular, the Web Ontology Language, OWL, is on the verge of gaining acceptance from a wide range of e-Science groups and organisations.

Despite the fact that OWL is generally seen as a solution to a number of design and implementation areas in e-Science, the complexity of the language, and understanding its logical underpinnings can be a barrier to its correct use. To address this situation, various ontology editors such as Protégé-OWL [1] and SWOOP [2] have been developed. However, the default interfaces in both of these tools are somewhat geared towards the tastes of logicians. These kinds of users are typically seduced by the ability to take advantage of the full expressivity of OWL. In contrast, for many other types of users, it is frequently the case that a particular application or style dictates that only a subset of the full expressive power of a language such as OWL is required. This is particularly true when the aim is to extend a simple terminology or taxonomy. In addition, most users need an environment that makes ontology development a fast and pain free process. They also require tools support for frequently used ontology design patterns, so that such patterns can be applied quickly and without error.

Using real requirements from groups, and exposure to difficulties experienced by beginners during tutorials, we have developed and are experimenting with several different user interfaces. These user interfaces attempt to address the following points:

1) Hiding the logical constructs and complex notation from the user.
2) Selecting appropriate expressiveness for a given group.
3) Abstracting away from the language by simulating some higher-level constructs.

\(^1\) http://www.co-ode.org
\(^2\) http://protege.stanford.edu/plugins/owl
We have developed a range of plugins for Protege-OWL that help achieve the above goals. A selection of these plugins are described below.

We have developed a new syntax - the Manchester OWL Syntax - for the representation of ontology classes. We have found this to be the syntax of choice for domain experts and non-logicians. In addition to the development of this syntax, a special editor has been developed (Figure 1). As well as the customary auto completion, keyword highlighting, error highlighting and pretty printing, the editor provides "macros" or shortcuts for common patterns. For example, multiple existential or universal restrictions along a particular property, coupled with closure, can be created in a very compact form. We have found that the editor results in a significant increase in the speed at which complex or long class descriptions can be constructed.

Finally, we have developed a suite of wizards (Figure 3) that allow users to quickly and accurately complete time consuming and error prone tasks. Such tasks range from quickly inputting/sketching a class hierarchy, to filling in restrictions for multiple classes. The wizards also support several of the W3C Semantic Web Best Practices Working Group recommended ontology design patterns.

Work with the International Organisation for Terminology in Anaesthesia (IOTA) group, who are developing a medical terminology, has produced a simplified class editing interface. The interface (Figure 2) trades expressivity for simplicity. Despite this seeming reduction in expressivity, we have found it to be very popular as it allows users to enter class descriptions in a “property oriented” or UML like manner.
Conclusion

It is hoped that the availability of the above tools will aid ontology development in e-Science projects. We are interested in using this opportunity to showcase the tools, and discuss the difficulties that users have had with developing ontologies. In particular, we would like to obtain tools requirements from researchers involved with e-Science projects.

Acknowledgements

The CO-ODE project is funded by JISC.