

Ontology Management with the PROMPT Pplugin

Natalya F. Noy
noy@smi.stanford.edu

The study of ontologies and their use is no longer just one of the fields in the Artificial Intelligence literature. Ontologies are now ubiquitous in many information-systems enterprises: they constitute the backbone for the Semantic Web, they are used in E-commerce, and in various application fields such as bioinformatics and medicine. As a result, developers are designing a large number of ontologies using different tools and different languages. These ontologies cover unrelated or overlapping domains, at different levels of detail and granularity. Such wide-spread use of ontologies inevitably produces an *ontology-management problem*: ontology developers and users need to be able to find and compare existing ontologies, reuse complete ontologies or their parts, maintain different versions, and so on. In other words, ontology developers face problems that are very similar to the ones that software engineers have been facing for many years.

The following are some of the tasks in managing multiple ontologies.

Maintain libraries of ontologies Allow uniform access to ontologies in a library, provide pertinent information about each ontology, such as its authors, domain, documentation, etc., provide search capabilities across all the ontologies in a library, allow browsing of the ontologies themselves.

Provide support for ontology versioning Provide mechanisms for storage and identification of different versions of the same ontology and for highlighting differences between versions.

Merge ontologies Given source ontologies, create a new ontology that incorporates information from all the sources.

Align and map between ontologies Define correspondences between concepts and relations in different ontologies.

Extract semantically independent parts of an ontology Analyze dependencies and allow users to extract sets of concepts and relations as a sub-ontology.

Many of the tasks in multiple-ontology management are closely interrelated, have common elements and subtasks, and tools for supporting some of the tasks can benefit greatly from their integration with others. For example, the methods that we develop to help users find overlap between ontologies for the tasks of ontology merging can be used successfully in finding differences between ontology versions. In both cases, we have two overlapping ontologies and we need to determine a mapping between their elements. When we compare ontologies from different sources, we concentrate on *similarities*, whereas in version comparison we need to highlight the *differences*, which can be a complementary process.

We get significant advantages from looking at these tasks together rather than completely independently. These advantages include:

- We can leverage *algorithms* for finding similarities between overlapping ontologies coming from *different* sources to find differences between versions of the *same* ontology. In the case of versioning, we use different thresholds to decide whether two frames are similar, but underlying analysis can be the same. Similarly, we can use some of the heuristics that we discovered when comparing ontology versions to compare different ontologies.
- All of the ontology-management tasks involve comparison of several ontologies. Analyzing and understanding semantic relations between elements of different ontologies is a cognitively difficult task. Regardless of whether the ontologies are headed for merging or alignment, or whether the user simply wants to compare them, having a *uniform user interface* for showing similarities and differences between ontologies, suggestions for integrating them, visualization of large-scale ontologies and relations among them, will greatly reduce the cognitive load for a user.

Figure 1 presents the PROMPT ontology-management framework, which we developed in our laboratory. All the components of this framework are plugins or extensions to the Protégé ontology-development environment.

The PROMPT framework brings together different ontology-management tools and provides an infrastructure for other related tools. The key components of the framework are:

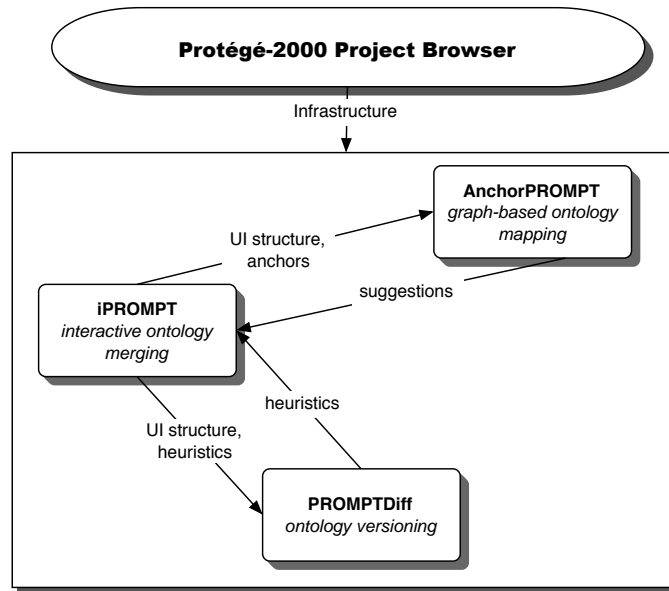


Figure 1: The PROMPT infrastructure

iPROMPT—an interactive ontology-merging tool, which assists users in merging ontologies by providing suggestions, analyzing conflicts, and suggesting conflict-resolution strategies [2]

ANCHORPROMPT—a graph-based tool for finding related concepts in different ontologies [1]. ANCHORPROMPT takes as input pairs of related terms in the source ontologies and analyzes the graph structure of the ontologies to find new pairs of related terms.

PROMPTDIFF—an ontology-versioning tools that determines what has changed between two versions [3].

Protégé project browser—a tool that provides access to a library of ontologies, giving users meta-information about an ontology (authors, documentation, modification date, etc.), snapshots of the top level of ontologies, search through classes and slots in all ontologies

All of these tools benefit from one another. For example, iPROMPT, as the first tool in the framework, provided user-interface components for other tools. These components allow users to browse two ontologies side-by-side, use of different colors for concepts from different ontologies, lists of pairs of

related terms, and so on. In addition, IPROMPT provides pairs of related terms to ANCHORPROMPT. Analysis in ANCHORPROMPT in turn provides additional suggestions that IPROMPT can present to the user. PROMPTDIFF uses some of the heuristics that we developed in IPROMPT in its comparison of ontology version.

In this talk, we will describe the various components of the PROMPT plugin to Protégé, their functionalities and relations between them.

References

- [1] N. F. Noy and M. A. Musen. Anchor-PROMPT: Using non-local context for semantic matching. In *Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001)*, Seattle, WA, 2001.
- [2] N. F. Noy and M. A. Musen. The PROMPT suite: Interactive tools for ontology merging and mapping. *International Journal of Human-Computer Studies*, 59(6):983–1024, 2003.
- [3] N. F. Noy and M. A. Musen. Ontology versioning in an ontology-management framework. *IEEE Intelligent Systems*, page in press, 2004.