Generating a Document-Oriented View of a Protégé Knowledge Base

Samson W. Tu, Shantha Condamoor, Mark Musen

Stanford Medical Informatics, Stanford University School of Medicine
Stanford, CA 94305-5479, USA

Abstract

A frame-based knowledge base can be a very large network whose information content is difficult for developers and users to comprehend and manipulate. In the SAGE consortium that is creating the technological infrastructure required in developing enterprise clinical decision-support systems for guideline-based care, knowledge engineers and clinicians must collaborate to develop knowledge bases of computable clinical practice guidelines. To facilitate communication among project members, we created a method to annotate a Protégé ontology and use the annotations to export an XML version of the Protégé knowledge base that can be transformed into a readable document. We demonstrated the generality of the method by applying it to other Protégé knowledge bases such as those developed by the PRODIGY guideline projects.

Introduction

A frame-based knowledge base is a network of classes, instances, and relationships among them. A large knowledge base may contain thousands or even millions of frames. The default Protégé graphical user interface provides a view of the knowledge base in terms of class hierarchies and direct instances of classes. Over the years, specialized views, such as the diagram/graph widgets, instance-tree tab, Ontoviz, and Jambalaya, have added additional ways of viewing and visualizing the content of Protégé knowledge bases. Most of these tools expose a relatively small amount of information at any given time. Protégé backends and export facilities such as the "html export," on the other hand, dump out all or large portions of Protégé knowledge bases, but only in terms of Protégé's class, instance, slot, and facet modeling constructs.

For projects that involve collaborative development of knowledge bases by computer scientists and domain experts, there is a pressing need to make the content of knowledge bases intelligible to those who are not users of sophisticated knowledge-engineering tools. The format should abstract away from the knowledge representation formalism, but at the same time presents an accurate view of the knowledge content. For the purpose of *reviewing* the knowledge content, the format should also expose large amount of information. Having such a review format not only benefits collaboration with domain experts, but also provides knowledge engineers with an efficient method for reviewing a knowledge base systematically.

The SAGE (Standards-Based **S**harable **A**ctive **G**uideline **E**nvironment) project, a collaboration among research groups at IDX Systems Corporation, the University of Nebraska Medical Center, Intermountain Health Care (IHC), Apelon, Inc., Stanford University, and the Mayo Clinic, seeks to create the technology for integrating guideline-based decision support into enterprise clinical information systems. The project is creating an ontology of computable clinical practice guidelines (CPG) and is validating this ontology by formalizing a number of CPGs in terms of this ontology and by simulating the management of patient cases according to these formalized CPGs. The knowledge-base development process involves close collaboration among computer scientist, knowledge engineers, and expert clinicians (Tu, Musen et al., 2004). Clinicians who are trained in the use of Protégé tools generally do the bulk of the encoding work. Each of the CPG knowledge bases typically includes several thousand frames. Even expert users of Protégé have difficulty drilling down to the depth of the knowledge base and understand the modeling decisions made by the encoders. The project decided to develop a tool to generate a document-oriented view of these Protégé knowledge bases so that colleagues in the project can review them and identify and correct both content and modeling problems.

Method

A fully developed CPG knowledge base contains not only the content of a computable guideline, but also formal terminologies, organization model, and patient data model that are necessary for the guideline to be implemented in

the electronic medical record of an enterprise. We decided that the scope of the document-oriented view should consist of frames that can be reached directly and indirectly from a particular top-level *Guideline* instance. In Protégé term, that means exposing the content of an instance tree. The tool should traverse the instance tree, and for each frame in the tree, generate XML output according to templates associated with classes that are direct types of these frames. Furthermore, we decided that we should leverage the Protégé graphs that we use to model guideline activities and decisions to organize the document-oriented view. Thus, the tool should create clickable images of these graphs to allow a user to navigate to different parts of the document. In the final step, we use Extensible Stylesheet Language Transformation to transform image files and XML file into an html file. The overall architecture is shown in Figure 1.

We modeled templates for generating XML fragments as a Protégé knowledge base itself. For each Protégé class (or metaclass) whose instances we want to include as part of the XML output, we enumerate, as part of the template associated with the class, an ordered list of slots whose values we want to include in the output. The default XML output use the names of the class and of the slots as XML tags. Thus, for an instance of the class *Presence_Criterion*

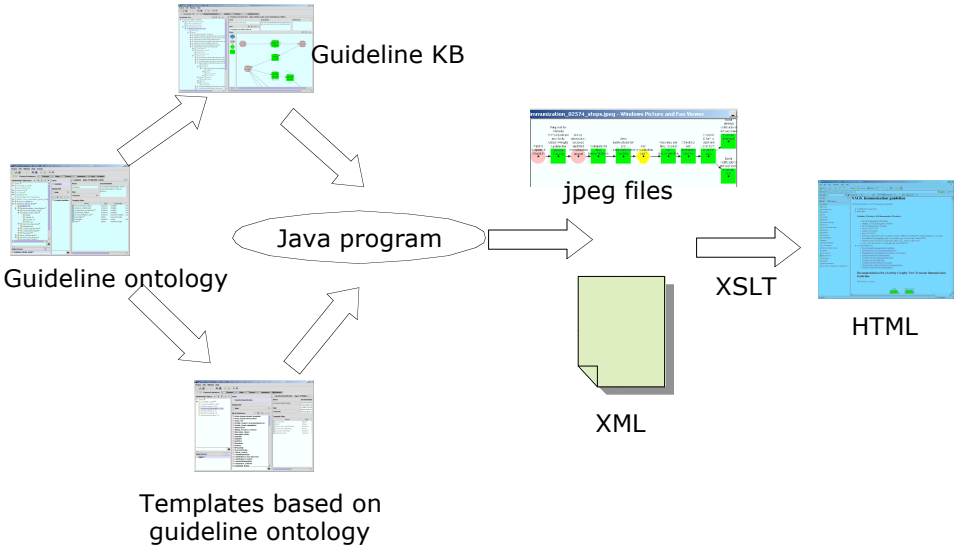

Figure 1. The process to create a document-oriented view from a Protégé knowledge base.

that has slots *code, presence, valid_window, and vmr_class,* the XML fragment looks like the following:

```
<Presence_Criterion p_id= "sageimmunization_02486">
    <code>MMR vaccine</code>
    <presence>true</presence>
    <valid_window>
       <Relatve_TimeInteval p_id="sageimmunization_42">
                             28.0 day before NOW
       </Relative_TimeInterval>
    </valid_window>
    <vmr_class>SubstanceAdministration</vmr_class>
```

In order to provide greater user-level control of the output format, we defined alternative ways of specifying textual patterns for writing instance. In Figure 2, we show an alternative template for generating text for instances of Presence_Criterion. The pattern "`{presence} of {vmr_class}{code}{valid_window}`" specifies how values of slots should be substituted into the pattern to generate a string. Furthermore, for selected slots, we specify how text may be generated based on presence or absence of slot values. Thus, for the slot *valid_window*, the slot template indicates that, the slot value should be preceded by "and time is within." The *Presence_Criterion* instance shown earlier becomes the string "`presence of SubstanceAdministration MMR vaccine and time is within 28.0 day before NOW.`"

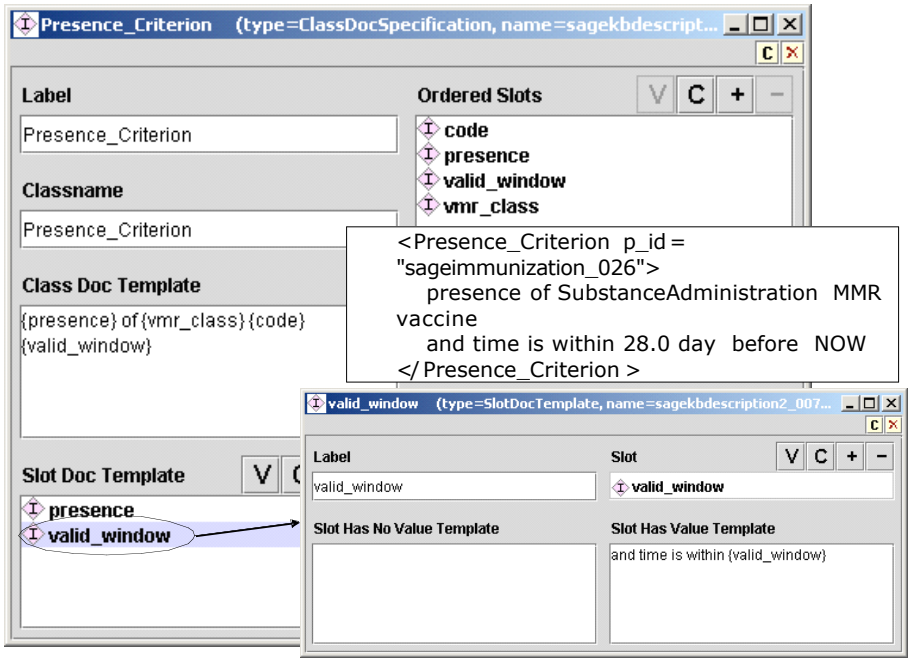

Figure 2 Alternative template for specifying output format of an instance of Presence_Criterion

Results

Using the template knowledge base and XSLT transforms, we successfully generated html views of guideline knowledge bases encoded in Protégé. The tool has been incorporated into the SAGE guideline workbench (Shankar, Tu et al., 2003) so that the document-oriented view can be generated at any stage of the knowledge development process. We further demonstrated the generality of this document-generation technology by applying it to the PRODIGY guideline model (Johnson, Tu et al., 2000) and generated html pages showing the details of a PRODIGY guideline for managing patients with prior myocardial infarction.

We are exploring the possibility of generating XML schemas for the XML format used in this export technology. With such a schema, we may be able to use the same format for *importing* guideline knowledge bases developed in alternative tools, as long as they conform to the SAGE guideline ontology.

References

Johnson, P. D., S. W. Tu, et al. (2000). Using Scenarios in Chronic Disease Management Guidelines for Primary Care. Proc AMIA Symp, Los Angeles, USA.

Shankar, R. D., S. W. Tu, et al. (2003). A Knowledge-Acquisition Wizard to Encode Guidelines. Proc AMIA Symp, Washington DC.

Tu, S. W., M. A. Musen, et al. (2004). Modeling Guidelines for Integration into Clinical Workflow. Medinfo 2004, San Francisco, CA, USA.