

Reuse of JessTab Rule Sets in the Protégé Environment

David Corsar, Derek Sleeman

{dcorsar, sleeman}@csd.abdn.ac.uk

Computing Science Department, University of Aberdeen, Aberdeen, UK

The Problem

Due to their ability to provide an explicit definition of a domain, ontologies are one of the most commonly used forms of domain knowledge capture. The popular Protégé environment provides an extensive ontology and knowledge base authoring and maintenance environment in one tool. One way to reason over this knowledge base is provided by the JessTab Protégé plugin, which provides a bridge between the Java Expert System Shell (JESS) rule engine and the Protégé knowledge base. However these rules are explicitly tied to the knowledge base for which they were developed as they are required to name *particular* classes and slots. Although an unavoidable requirement, this can greatly hinder reusing a set of JessTab rules developed for one ontology/knowledge base with additional knowledge bases. In particular, the developer could manually define mappings between the class and slot names in the JessTab rules and those in the second and subsequent knowledge bases. However this would be a tedious and very error prone process.

Supporting and Automating the Rule Set Reuse Process

We have developed a Protégé plugin which supports the user in achieving reuse of a JessTab rule set. By automating many of the procedures the user would otherwise be required to perform manually, our plugin simplifies and accelerates the process of modifying a JessTab rule set for reuse.

The reuse process supported by our plugin, as illustrated in Figure 1, consists of two phases. In the first phase a set of JessTab rules are modified to remove all references to the ontology for which they were developed. This process involves extracting, from the ontology, all the class names along with all the slot names and each slot's data type. All occurrences of a class name in the rules are then replaced by an abstract class name consisting of a prefix and the original class name. Similarly all occurrences of a slot name are replaced by an abstract slot name consisting of a prefix, the original slot name and the slot data type to produce a set of abstract, ontology independent JessTab rules.

The second phase consists of mapping between the abstract rules and the concepts defined within the ontology they are to be (re)used with. The abstract references in the abstract rules are used to create a list of the classes (and their respective slots) that were used by the rules. Our tool supports a mapping process between the *extracted* classes and those of the second ontology. In particular, our tool supports this by providing an automatic mapping feature which suggests mappings based on lexical analysis and comparisons of the classes and slot names while also ensuring slot data type compatibility. The lexical analysis and comparisons utilize checks ranging from simple string comparison, minor spelling variation and WordNet based synonym look up. Additionally the tool provides a clear user interface which allows the user to correct and/or complete the mapping process. Having defined and performed mappings for all the concepts extracted from the rules, the rules are then ready to be executed against the target ontology.

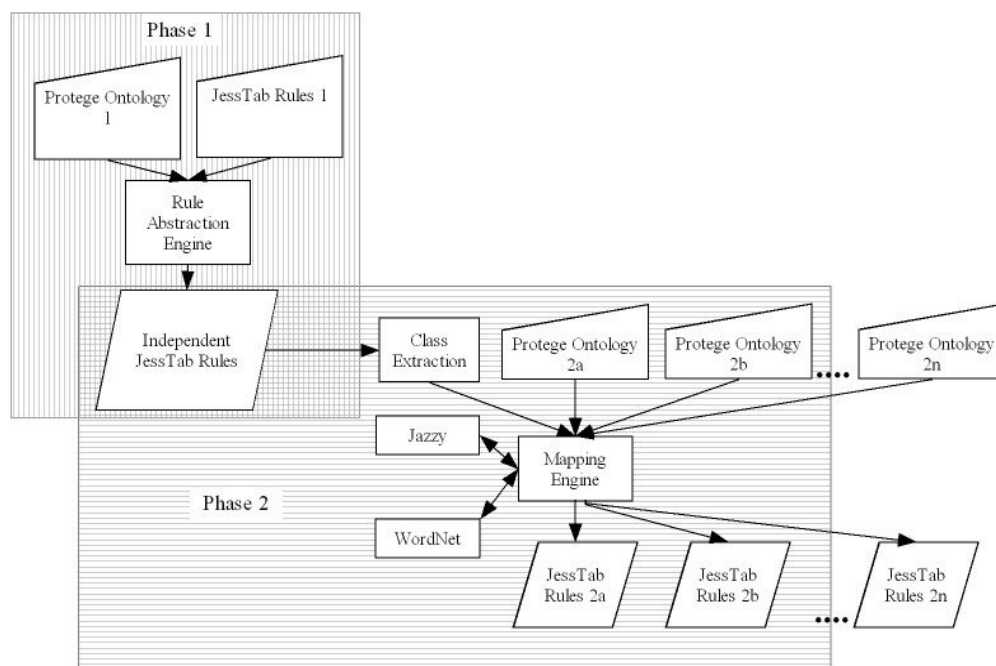


Figure 1: Illustration of our automated JessTab rule reuse process.

Sample Mappings

Table 1 provides a selection of results from our experiments with the Document ontology downloaded from the Stanford KSL OKBC server with the Protégé OKBC Tab plugin.

Concept Name	Data Type	Abstract Name	Mapping in to			
			Application A		Application B	
			Name	Mapping Type	Name	Mapping Type
Document	Class	XX_Document	Document	Automatic	Docment	Automatic
Thesis	Class	XX_Thesis	Thesis	Automatic	Dissertation	Automatic
Artwork	Class	XX_Artwork	Art	Automatic	Artistry	Manual
Has-Author	String	XX_Has-Author_String	Author	Automatic	HasWriter	Automatic
Title-Of	String	XX_Title-Of_String	Title	Automatic	TitleOf	Automatic
Publisher_Of	String	XX_Publisher-Of_String	PublicationHouse	Manual	PublisherOf	Automatic

Table 1: Mappings for selected classes and slots of the OKBC Document Ontology.

Demonstration

We plan to demonstrate the reuse of single set of JessTab rules with ontologies from several domains for example planning, resource management and assembly. In fact, we claim to have gone some way to implementing the vision, developed by John Park and Mark Musen of creating ad-hoc knowledge bases systems from existing problem solving methods (here JessTab rule sets) and knowledge bases (here Protégé OKBC/OWL ontologies).