

The SWRLTab: An Extensible Environment for working with SWRL Rules in Protégé-OWL

Martin J. O'Connor, Amar Das
Stanford Medical Informatics, Stanford, CA 94305-5479
martin.oconnor@stanford.edu

Abstract

The SWRLTab is an extension to Protégé-OWL that supports editing and execution of SWRL rules. It provides a graphical editor for creating and modifying SWRL rules. It also provides extension mechanisms to support the execution of SWRL rules with a variety of rule engines as well as mechanisms to implement user-defined SWRL built-ins and execute them. At present, the Jess rule engine is supported and implementations have been defined for several built-in libraries including the core SWRL built-ins.

1. System Description

The SWRLTab [1] is an extension to the open-source ontology development toolkit in Protégé-OWL [2]. It provides a set of APIs that support the building of tools that work with SWRL [3] rules.

The SWRLTab has four main software components: (1) a graphical editor; (2) a Java API for working with SWRL rules; (3) a rule engine bridge that interoperates with rule engines to execute SWRL rules; and (4) a built-in bridge for defining and executing implementations of SWRL built-ins.

1.1 SWRL Editor

The Protégé-OWL SWRL Editor [2] is an extension to Protégé-OWL that permits interactive editing of SWRL rules. Users can create, edit, and read/write SWRL rules. With the exception of arbitrary OWL expressions, this editor supports the full set of language features outlined in the current SWRL submission. It is tightly integrated with Protégé-OWL and is primarily accessible through a tab within it. When editing rules, users can directly refer to OWL classes, properties, and individuals within an OWL knowledge base. They also have direct access to a full set of built-ins described in the SWRL built-in specification and to all of the XML Schema data types.

Figure 1 shows a screenshot of the Protégé-OWL SWRL Editor.

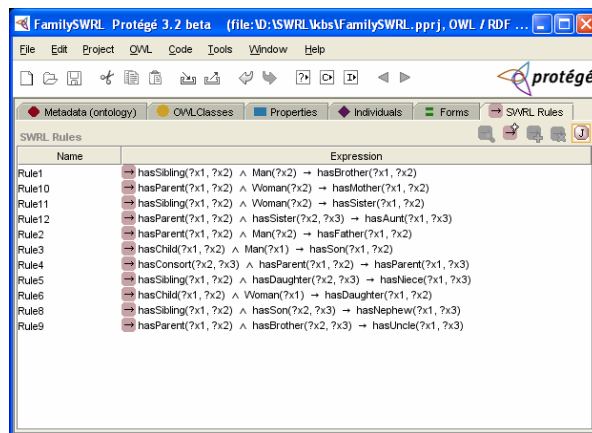


Figure 1. The SWRL Editor tab in Protégé-OWL. This tab provides a tabular listing of all SWRL rules in an OWL knowledge base.

1.2 SWRL Factory

The SWRLTab comes packaged with a Java API called the SWRL Factory, which allows developers to directly manipulate SWRL rules in an OWL knowledge base. This API can be used by developers wishing to work with SWRL rules in their applications. The SWRL Factory provides a mapping from OWL individuals representing SWRL rules to analogous Java instances. It also provides Java classes representing SWRL classes in the SWRL ontology [5] and mechanisms to create run-time instances of classes that mirror individuals in an OWL knowledge base. It is used internally by the SWRL Editor. However, it is accessible to all Protégé-OWL developers. SWRL plugin developers can base their work directly on the classes created by this factory and can, for example, use it to integrate existing rule engines with Protégé-OWL. This API could also be used to create new SWRL rule editors.

1.3 SWRL Rule Engine Bridge

The SWRL Rule Engine Bridge is a subcomponent of the SWRLTab that provides a bridge between an OWL model with SWRL rules and a rule engine. Its goal is to provide the infrastructure necessary to incorporate rule engines into Protégé-OWL to execute SWRL rules.

The bridge provides mechanisms to (1) import SWRL rules and relevant OWL classes, individuals, properties and restrictions from an OWL model; (2) write that knowledge to a rule engine; (3) allow the rule engine to perform inference and to assert its new knowledge back to the bridge; and (4) insert that asserted knowledge into an OWL model. The bridge also provides mechanisms to dynamically add graphical user interfaces to the SWRLTab to allow interaction between a particular rule engine implementation and a user.

A bridge for the Jess rule engine [6] called the SWRLJessTab [7] is provided in the Protégé-OWL distribution. Figure 2 shows a screenshot of the SWRLJessTab.

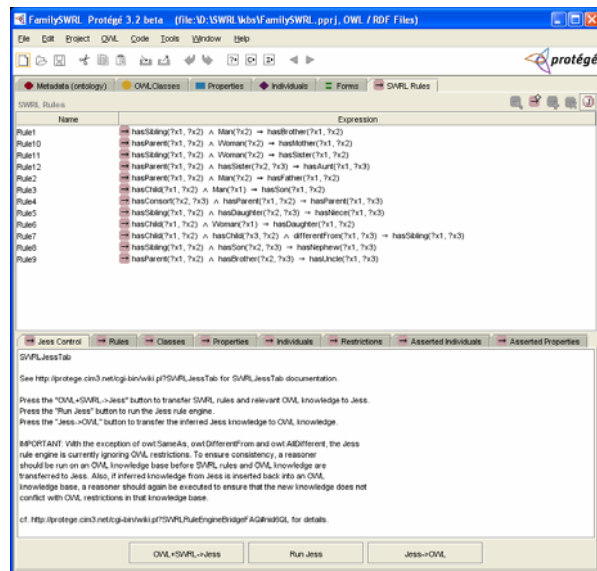


Figure 2 SWRLJessTab. This tab provides a graphical means to control Jess inference with SWRL rules.

1.4 SWRL Built-in Bridge

SWRL provides a very powerful extension mechanism that allows the use of user-defined methods in rules. These methods are called built-ins and are predicates that accept one or more arguments. Built-ins are analogous to functions in production rule systems. A number of core built-ins are defined in the SWRL specification. This core set includes basic mathematical operators and built-ins for string and date manipulations. SWRL users can also define their own built-in libraries. Example libraries could include built-ins for currency conversion, or for statistical, temporal or spatial operations. Again, these user-defined built-ins can be used directly in SWRL rules.

We have developed an extension to the Protégé-OWL SWRLTab called the SWRL Built-in Bridge. This extension provides support for defining built-in implementations written in Java and dynamically loading them. Users wishing to provide implementations for a library of built-in methods can define a Java class that contains definitions for all the built-ins in their library. The bridge has a dynamic loading mechanism to import these built-in definitions and provides an invocation mechanism to execute these loaded definitions from rule engines.

We have defined an implementation library for most of the core SWRL built-ins. We have also developed a library of temporal built-ins that allows SWRL to be used to perform temporal reasoning.

2. References

- [1] <http://protege.cim3.net/cgi-bin/wiki.pl?SWRLTab>
- [2] Knublauch, H. R.W. Ferguson, N.F. Noy, and M.A. Musen, "The Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications", *Proc Third ISWC (ISWC 2004)*, Hiroshima, Japan, pp. 229-243, 2004.
- [3] <http://www.w3.org/Submission/SWRL/>
- [4] O'Connor M.J., et al., "Supporting Rule System Interoperability on the Semantic Web with SWRL", *Fourth International Semantic Web Conference (ISWC2005)*, Galway, Ireland pp. 974-986, 2005.
- [5] <http://www.daml.org/2003/11/swrl/swrl.owl>
- [6] <http://www.jessrules.com/>
- [7] <http://protege.cim3.net/cgi-bin/wiki.pl?SWRLJessTab>
- [8] <http://www.daml.org/2004/04/swrl/builtins.html>