

# **Visualizing SWRL Rules Using Grailog Frame Formulas**

## **Semantic Web Techniques**

Bo Yan,

Junyan Zhang,

Ismail Akbari

*Instructor: Harold Boley*

Fall 2012

## Table of Contents

Introduction.....	3
Objectives .....	3
Methodology.....	3
Project tools .....	4
SWRLTab.....	4
Jambalaya.....	5
OWLViz .....	5
Axiomé.....	6
References.....	6

## Introduction

SWRL (Semantic Web Rule Language) is an implemented Semantic Web rule language, combining the sublanguages Web Ontology Language, Description logic (OWL DL) and Rule Markup Language (Unary/Binary Datalog). [1]. SWRL has become one of the most popular Web rule languages, possibly because of its use as SWRLTab in the very wide-spread Protégé ontology editor and knowledge-base framework. SWRL's employs Unary/Binary Datalog formulas, which can also be seen as slotted formulas in F-logic, W3C RIF-BLD, or PSOA RuleML terms. This allows two alternate Grailog visualization methods. Graph inscribed logic (Grailog) invokes imagery for logic Proposed cognitively motivated systematic graph standard for visual-logic knowledge. Graphs should make it easier for humans to read and write logic constructs by exploiting a 2-dimensional representation with shorthand & normal forms, from Controlled English to logic [2].

## Objectives

SWRL's employs Unary/Binary Datalog formulas, which can also be seen as slotted formulas in F-logic, W3C RIF-BLD, or PSOA RuleML terms. This allows two alternate Grailog visualization methods.

Both will be explored and compared in the project. One of the two Grailog visualization methods will be chosen. The implementation approach for it can be chosen among the open-source Axiom's further Protégé visualizers (e.g., Jambalaya/OntoGraf or OWLViz for OWL ontologies), or other approaches. In this project we will use the Graphviz [3] to perform the visualization. Also OWLViz uses the Graphviz to do the visualization. SWRL rules will be received as input to the project and their slotted syntax and graphical presentation based on Grailog will be shown as output. The Grailog visualization will be exemplified with a SWRL rule set from the Protégé site. The results will be made available on the Web, open source, so they can be pointed to from the RuleML and Grailog sites.

## Methodology

- Describe SWRL's Unary/Binary Datalog terms and how they can be represented as slotted terms. Also their logical syntax will be presented in brief.
- These two alternate methods of SWRL give two methods for Grailog visualization. We compare these methods and show pros and cons of each method. However, the concentration of this project is on Grailog visualization using slotted formulas.
- How to visualize SWRL's slotted formulas in Grailog. To do so, some tools as mentioned in next section can be applied for the visualization purpose. However, the main tool which is used in this project is Graphviz, because of its simplicity and ease of use. For achieving our purpose, a tool will be developed based on Java which is able to show SWRL rules and their equivalents in Grailog (in both logical slotted syntax and visual graph syntax).

- Some SWRL rules will be used as input and their Grailog slotted formulas and graphs as output will be generated. We use the following sub steps to visualize SWRL rules:
  - Convert SWRL human readable syntax to OWL syntax.
  - Convert OWL syntax to Grailog's logical syntax.
  - Apply Graphviz to visualize the rules based on Grailog format (special shapes and arrows is used in Grailog). The output visual graph can be available in different formats such as png, ps, imap, svg, gif and some other formats.
  - Test both logical syntax and visualization method of the SWRL rules using some examples.
- Conclusion

## Project tools

### SWRLTab

The SWRLTab is a development environment for working with SWRL rules in Protege-OWL. It supports the editing and execution of SWRL rules and includes a set of libraries that can be used in rules, including libraries to interoperate with XML documents, and spreadsheets, and libraries with mathematical, string, RDFS, and temporal operators. A SWRL-based OWL query language called SQWRL is also provided. The SWRLTab is designed to work in Protege-OWL only. However, partial interoperation with Jena is also supported. The SWRLTab has several software components [4]:

- **SWRL Editor:** The editor supports editing of SWRL rules in an OWL ontology.
- **SWRL APIs:** The SWRLTab provides a collection of Java APIs to work with SWRL rules.
- **SWRL Built-in Libraries:** A number of built-in libraries are provided by the SWRLTab. These include an implementation of the core SWRL built-ins defined in the SWRL Submission and built-ins for querying OWL ontologies.
- **SQWRL Query Tab:** The query tab provides a graphical interface to display the results of SQWRL queries.
- **SQWRL Query API:** This API provides a JDBC-like Java interface to retrieve the result of SQWRL queries.
- **SWRL Temporal Ontology and Library:** The SWRLTab provides an array of mechanisms for modeling, reasoning with, and querying temporal information.
- **SWRL Built-in Bridge:** SWRL built-ins are user-defined predicates that can be used in SWRL rules. The SWRLTab has a subcomponent called the built-in bridge that provides a mechanism to define Java implementations of SWRL built-ins. These implementations can then be dynamically loaded by the bridge and invoked from a rule engine.
- **SWRL Bridge:** The bridge provides the infrastructure necessary to incorporate rule engines into Protege-OWL to execute SWRL rules.

- **SWRL Jess Bridge:** A bridge for the Jess rule engine is provided in the Protege-OWL distribution. A user interface called the SWRLJessTab is also provided to interact with this bridge.
- **SWRL Factory:** The factory provides high-level Java APIs that support the creation and modification of SWRL rules in an OWL ontology. This API can be used by developers who wish to work with SWRL rules in their applications.

## Jambalaya

Jambalaya is a plug-in created for Protégé which uses Shrimp to visualize the knowledge bases the user has created. Protégé is a “an ontology editor and a knowledge-base editor”, developed at Stanford University, which allows domain experts to build knowledge-based systems by creating and modifying reusable ontologies and problem-solving methods. Jambalaya uses the Piccolo zooming library developed by the Human-Computer Interaction Lab (HCIL) at the University of Maryland [5].

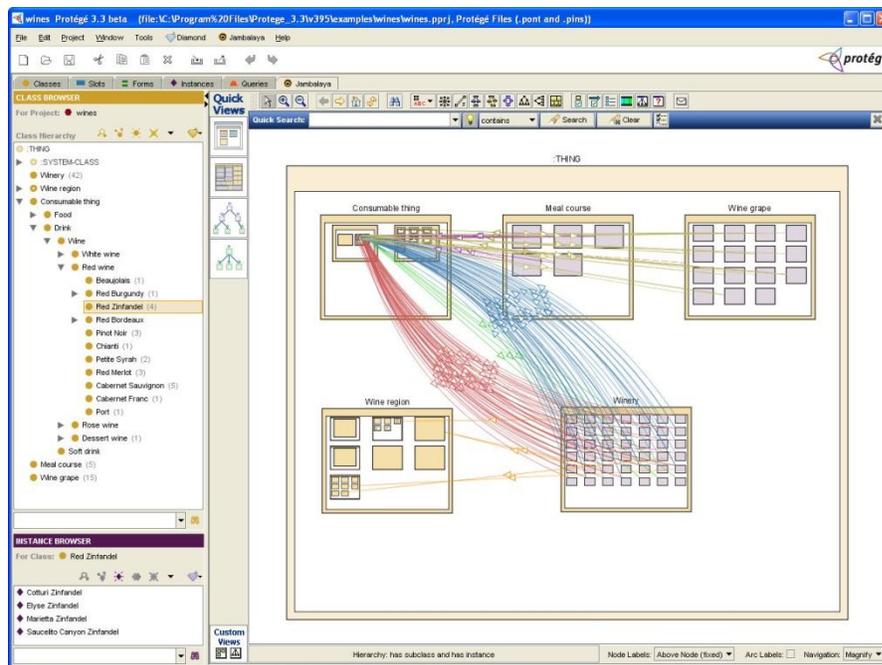


Figure 1. Jambalaya screenshot

## OWL Viz

OWL Viz is designed to be used with the Protege OWL plugin. It enables the class hierarchies in an OWL Ontology to be viewed and incrementally navigated, allowing comparison of the *asserted* class hierarchy and the *inferred* class hierarchy. OWL Viz integrates with the

Protege-OWL plugin, using the same colour scheme so that primitive and defined classes can be distinguished, computed changes to the class hierarchy may be clearly seen, and inconsistent concepts are highlighted in red. OWLViz has the facility to save both the asserted and inferred views of the class hierarchy to various concrete graphics formats including png, jpeg and svg [6].

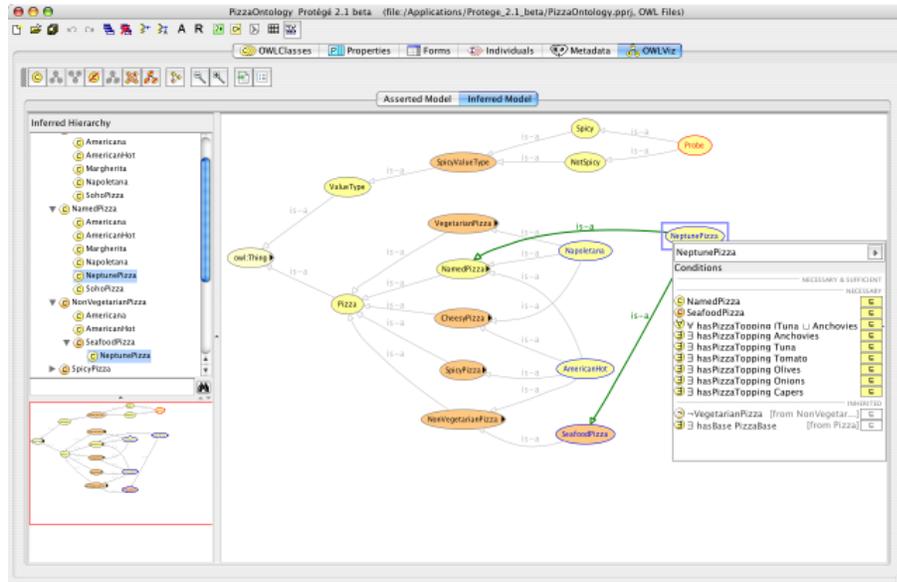


Figure 2. OWLViz screenshot

## Axiom é

Axiom é is developed as a Prot é é-OWL plug-in and has five main functional areas [7]. These are available as sub-tabs within the plug-in:

- **Rule Graph** tab that provides a graph structure to browse and explore the SWRL rule base
- **Rule Visualization** tab to visualize individual rules
- **Rule Paraphrasing** tab that displays an English-like text explanation for each rule
- **Rule Categorization** tab to automatically categorize rules in a rule base
- **Rule Elicitation** tab that provides a graphical template to acquire new rules based on analysis of existing rules in a rule base.

A Rule Browser component is permanently displayed to show a tree-table representation of the SWRL rules in an ontology. This tree-table enables users to explore the rule base and launch any of five sub-tabs for the rule or group being explored [7, 8].

## References

- [1] [http://en.wikipedia.org/wiki/Semantic\\_Web\\_Rule\\_Language](http://en.wikipedia.org/wiki/Semantic_Web_Rule_Language).

[2] The Grailog Systematics for Visual-Logic Knowledge Representation with Generalized Graphs. Harold Boley, <http://www.cs.unb.ca/~boley/talks/RuleMLGrailog.pdf>.

[3] [www.graphviz.org/](http://www.graphviz.org/)

[4] <http://protege.cim3.net/cgi-bin/wiki.pl?SWRLTab>.

[5] <http://thechiselgroup.org/2004/07/06/jambalaya>.

[6] <http://www.co-ode.org/downloads/owlviz>.

[7] <http://protegewiki.stanford.edu/wiki/Axiom%C3%A9>.

[8] Saeed Hassanpour, Martin J. O'Connor, Amar K. Das, "Axiomé: a Tool for the Elicitation and Management of SWRL Rules", OWL: Experiences and Directions (OWLED), Chantilly, VA, 2009.