R-DEVICE: An Object-Oriented Knowledge Base for RDF Metadata

Nick Bassiliades, Aristotle University of Thessaloniki, Greece

Ioannis Vlahavas, Aristotle University of Thessaloniki, Greece

ABSTRACT

In this paper we present R-DEVICE, a deductive object-oriented knowledge base system for reasoning over RDF metadata. R-DEVICE imports RDF documents into the CLIPS production rule system by transforming RDF triples into COOL objects and uses a deductive rule language for reasoning about them. R-DEVICE is based on an OO RDF data model, different than the established triple-based model, which maps resources to objects and encapsulates properties inside resource objects, as traditional OO attributes. In this way, fewer joins are required to access the properties of a single resource resulting in better inferencing/querying performance, as it is experimentally shown in the paper. Furthermore, RDF can interoperate seamlessly with other Web data models and languages. The descriptive semantics of RDF may call for dynamic redefinitions of resource classes, which are handled by R-DEVICE effectively. Furthermore, R-DEVICE features a powerful deductive rule language for reasoning on top of RDF metadata. The rule language includes features such as normal and generalized path expressions, stratified negation, aggregate, grouping, and sorting functions. The rule language supports a second-order syntax, which is efficiently translated into sets of first-order logic rules using metadata, where variables can range over classes and properties, so that reasoning over the RDF schema can be made. Users can define views that are materialized and incrementally maintained by translating deductive rules into CLIPS production rules that preserve truth. Users can choose between an OPS5/CLIPS-like and a RuleML-like syntax. Finally, users can define and use functions through the CLIPS host language.

Keywords: aggregation; CLIPS; deductive rules; descriptive semantics; generalized path expressions; materialized views; object data model; production rules; RDF; RuleML

INTRODUCTION

The Semantic Web is the next step of evolution for the World Wide Web (Berners-Lee, Hendler, & Lassila, 2001), where information is given well-defined meaning, enabling computers and people to work in better cooperation. Currently, information found on the Web is mainly for human consumption and is not machine understandable. It is quite difficult to automate things on the Web, and because of
the volume of information the Web contains, it is even more difficult to manage it manually. The solution proposed by the WWW Consortium is to use metadata to describe the data contained on the Web (Berners-Lee, 1997). The Resource Description Framework (RDF) is a foundation for processing metadata; it provides interoperability between applications that exchange machine-understandable information on the Web (Manola & Miller, 2004).

RDF is actually a general-purpose language for representing information in the World Wide Web. However, it is particularly intended for representing metadata about Web resources, such as the title, author, and so forth. RDF generalizes the concept of a “Web resource,” so it can be used to represent information about anything that can be identified on the Web. The RDF model is based on sets of statements or triples, each of which can be thought of as a directed labelled graph in which nodes are called resources (or literals), and edges are called properties. The source node of an edge of the directed graph is the subject of the statement, and the target is the object. The edge is labeled with the predicate. Furthermore, RDF has a schema definition language (RDFS) (Brickley & Guha, 2004), for creating types for graph nodes (called classes) and edges (called properties). Finally, RDF has an XML syntax for expressing metadata and schemas in a form that is both human readable and machine understandable.

Conveying the content of documents is just a first step for achieving the full potential of the Semantic Web. Additionally, it is very important to be able to reason with and about information spread across the WWW, so that intelligent agents can automatically perform complicated tasks on the Web, on a user’s behalf. Rules provide the natural and widely-accepted mechanism to perform automated reasoning, with mature and available theory and technology. This has been identified as a Design Issue for the Semantic Web, as clearly stated in Berners-Lee (1997).

Rules and rule mark-up languages, such as RuleML (Boley, Tabet, & Wagner, 2001), will play an important role in the success of the Semantic Web. Rules will act as a means to draw inferences, to express constraints, to specify policies, to react to events/changes, to transform data, and so forth. Rule mark-up languages will allow enriching Web ontologies by adding definitions of derived concepts, to publish rules on the Web, to exchange rules between different systems and tools, and so forth. The applications include electronic commerce, data integration and sharing, information gathering, security access and control, law, diagnosis, B2B, and of course, to modelling of business rules and processes.

It seems natural to add rules “on top” of Web ontologies. However, as it is argued in Antoniou and Wagner (2003), putting rules and description logics together poses many problems, and may be overkill, both computationally and linguistically. Another possibility is to start with RDF/RDFS, and extend it by adding rules.

One solution to implement such a rule system is to start from scratch and build inference engines that draw conclusions directly on the RDF data model. However, such an approach tends to throw away decades of research and development of efficient and robust rule engines. In this paper we follow a different approach: we re-use an existing rule system (CLIPS [CLIPS Basic Programming Guide, 2003]) for reasoning on top of RDF data. However, before an existing rule system is used, careful consideration must be given to how RDF data and semantics are going to be treated in the host system.

The semantics of the RDF data model differ from those of traditional data structures, such as the object data model, that are used in existing rule systems. Specifically, RDF is an assertional language, according to its semantics (Hayes, 2004), that is, each assertion declares that certain information about resources is true, including schema information, and its meaning is not changed by future assertions. This kind of semantics is called descriptive and is based on the Open-World Assumption of the Semantic Web. Traditional data models define certain constraints in their schema definitions.
Related Content

Discovering and Characterizing Places of Interest Using Flickr and Twitter
Steven Van Canneyt, Steven Schockaert and Bart Dhoedt (2013). International Journal on Semantic Web and Information Systems (pp. 77-104).
www.igi-global.com/article/discovering-and-characterizing-places-of-interest-using-flickr-and-twitter/97654?camid=4v1a

Inductive Classification of Semantically Annotated Resources through Reduced Coulomb Energy Networks
www.igi-global.com/article/inductive-classification-semantically-annotated-resources/41735?camid=4v1a

A Core Ontological Model for Semantic Sensor Web Infrastructures
www.igi-global.com/article/core-ontological-model-semantic-sensor/70585?camid=4v1a
Knowledge Protocols
www.igi-global.com/chapter/knowledge-protocols/35734?camid=4v1a