Extending SWRL to Enhance Mathematical Support

Alfonso Sánchez-Macián[1,⋆], Encarna Pastor[1], Jorge E. López de Vergara[2], and David López[3]

[1] Dept. Ingeniería de Sistemas Telemáticos, Univ. Politécnica de Madrid
{aasmp, encarna}@dit.upm.es
[2] Dept. Ingeniería Informática, Univ. Autónoma de Madrid
jorge.lopez_vergara@uam.es
[3] Dept. Ingeniería Eléctrica y Electrónica, Univ. de León
david.lopez@unileon.es

Abstract. This paper presents an extension to the Semantic Web Rule Language and a methodology to enable advanced mathematical support in SWRL rules. This solution separates mathematical and problem semantics allowing the inclusion of integration, differentiation and other operations not built-in to SWRL. Using this approach, it is possible to create rules to cope with complex scenarios that include mathematical relationships and formulas that exceed the SWRL capabilities.

1 Introduction

Current Semantic Web languages provide a way to represent knowledge formally and exchange information. Some of these languages introduce Horn-like rules or First-order-logic support, so they enable declarative programming. One example is the Semantic Web Rule Language (SWRL) [2].

Developing complex systems will require the use of mathematical functions that are not currently supported in the Semantic Web languages. An example can be the implementation of real-time systems for emergency care units or vehicular control. Semantic Web languages do not provide the tools to cope with these scenarios. They usually include some mathematical built-ins to perform simple operations such as addition or subtraction. However, they are not designed to work with complex formulas.

This paper presents a methodology and a practical approach to add the required functionality to SWRL. The strategy is based on the separation between mathematical and problem semantics.

2 SWRL and Mathematical Semantics in the Web

The OWL Web Ontology Language [4] provides the base for knowledge representation by means of the definition of classes, properties and individuals. SWRL

⋆ Now at University of Southampton IT Innovation Centre, UK.

M. Marchiori, J.Z. Pan, and C. de Sainte Marie (Eds.): RR 2007, LNCS 4524, pp. 358–360, 2007.
© Springer-Verlag Berlin Heidelberg 2007

is based on OWL, adding high level abstract syntax to support Horn-like rules that can be used to deduce new facts in the knowledge-base. SWRL built-ins are used to perform specific mathematical computation (e.g. add, subtract, round, sin), comparisons (e.g. equal) and operations on different types of data value (e.g. strings).

Mathematical built-ins are useful, but the problem arises when the relationship to be presented implies unsupported operators (e.g summations). An example could be the calculation of a cumulative value of a Gaussian distribution as it requires the use of integration operators. There is also a problem of clarity due to mixing mathematical and problem semantics in the same rule. Furthermore, when the formula implies a high number of operations, the memory requirements for the reasoner increase as it has to store many temporary variables. The reason is that SWRL built-ins were designed as predicates. However, the capabilities of the existing reasoners allow us to use the built-ins as functions.

It becomes necessary to find a different way to represent mathematical formulas in SWRL. One of the main efforts to represent mathematical equations and formulas on the Web is OpenMath [1]. OpenMath is formed by a set of tags and content dictionaries. The tags allow the definition of primitive types, variables, symbols (e.g. π) and operators. Content dictionaries (CD) group mathematical symbols and define their semantics including arithmetic functions, transcendental functions, polynomials, differentiation and integration.

3 Working with SWRL and OpenMath

In order to overcome the issues pointed out in Section 2, we propose a combination of SWRL and OpenMath.

The solution uses SWRL to select the information to be included in the formulas and the formula itself. OpenMath is used to represent the formula and pass the information to a mathematical software tool. It is necessary to perform a binding between both languages in order to relate the information located by SWRL and the variables included in the OpenMath expression.

A new class is created called $Formula$ with a datatype property (*hasOMExpression*) to hold the OpenMath expression in XML. Additionally, a new built-in (*mathext*) is defined in a new namespace (*swrlbext*). This built-in has a minimum of three arguments. The first one is the result of the formula. The second one is the OpenMath formula. The rest of the arguments are the values that correspond to the variables in the formula. The reason for having a Formula instance rather than just passing the OpenMath expression to mathext is to allow reuse of the formula.

The classes, rules and formulas are written using an OWL editor. The formula, written in OpenMath, is set as the value of the hasOMExpression property of an instance of the Formula class.

When the reasoner uses a rule (Figure 1), it selects the appropriate values, placing them as values of the variables used in the rule. When it finds a customized SWRL built-in, it calls a built-in handler function that delegates

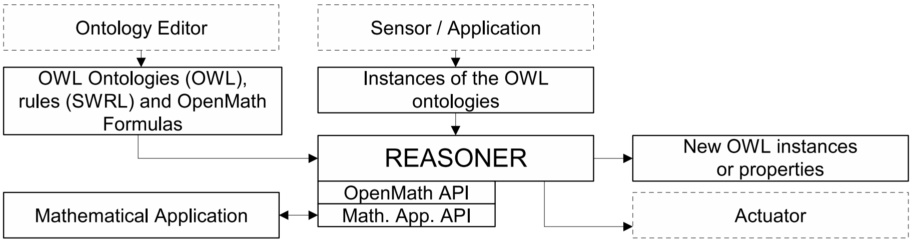

Fig. 1. SWRL-OpenMath Architecture

mathematical computation to programs specially prepared for it. It links to a mathematical application through existing tools (OpenMath and syntax translation APIs), gets the result and returns it to the reasoner. The reasoner resumes the rule execution and generates new statements depending on the values computed using the formula. The proposed architecture has been implemented using Bossam [3] and Mathematica[5].

4 Conclusions

This paper has identified the limitations of Semantic Web Languages to cope with complex scenarios that require the use of advanced mathematical expressions and conditions to deduce new facts. A new practical approach and architecture has been presented to add the required functionality to SWRL. The strategy is based on the separation between mathematical (OpenMath) and problem semantics (SWRL).

Acknowledgement. This work has been partially supported by the Spanish National Plan of Research, Development and Innovation (Ministry of Education and Science) under grants TIC2003-04406 and TSI2005-07306-C02-01.

References

1. Buswell, S., Caprotti, O., Carlisle, D.P., Dewar, M.C., Gaëtano, M., Kohlhase, M.: The OpenMath Standard version 2.0., Technical Report, The OpenMath Society, (June 2004) http://om-candidate.activemath.org/standard/om20-2004-06-30/
2. Horrocks, I., Patel-Schneider, P., Boley, H., Tabet, S., Grosof, B., Dean, M.: SWRL: A Semantic Web Rule Language Combining OWL and RuleML, W3C Member Submission, (May 2004) http://www.w3.org/Submission/SWRL
3. Jang, M., Sohn, J.: Bossam: an extended rule engine for the web. In: Antoniou, G., Boley, H. (eds.) RuleML 2004. LNCS, vol. 3323, pp. 128–138. Springer, Heidelberg (2004)
4. McGuinness, D.L., van Harmelen, F.: OWL Web Ontology Language Overview, W3C Recommendation, (February 2004) http://www.w3.org/TR/owl-features
5. Wolfram, S.: The Mathematica Book, 5th edn. Wolfram Media, Champaign, Illinois, US, (2003) http://documents.wolfram.com/mathematica