Embeddings of Simple Modular Extended RDF

Carlos Viegas Damásio¹, Anastasia Analyti², and Grigoris Antoniou³

1 CENTRIA, Departamento de Informática da Faculdade de Ciências e Tecnologia,
Universidade Nova de Lisboa, 2829-516 Caparica, Portugal
cd@di.fct.unl.pt
2 Institute of Computer Science, FORTH-ICS, Crete, Greece
analyti@ics.forth.gr
3 Institute of Computer Science, FORTH-ICS, and Department of Computer Science,
University of Crete, Crete, Greece
antoniou@ics.forth.gr

Abstract. The Extended Resource Description Framework has been proposed to equip RDF graphs with weak and strong negation, as well as derivation rules, increasing the expressiveness of ordinary RDF graphs. In parallel, the Modular Web framework enables collaborative and controlled reasoning in the Semantic Web. In this paper we exploit the use of the Modular Web framework to specify the modular semantics for Extended Resource Description Framework ontologies.

1 Introduction

The Extended Resource Description Framework [5] (ERDF) provides a model theoretical semantics for RDF graphs allowing negative triples, and ontologies defined by first-order rules including the two forms of negation, weak and strong. However, no means of combining different ontologies is specified. The necessity of mechanisms to encapsulate and organize knowledge in the Semantic Web is essential [13,6,8,10], and the ERDF framework has been extended to allow the specification of import and export declarations of classes and properties, resulting in the Modular ERDF framework [3]. The semantics of the modular ERDF framework has also been defined model theoretically, but it was lacking a declarative rule-based semantics for implementing the system.

In parallel, the Modular Web Framework (MWeb) is a proposal to address the issues of programming-in-the-wide faced by the new Semantic Web rule-engines [24]. MWeb defines general constructs to allow sharing of knowledge in the Semantic Web provided by logic based knowledge bases, including scoped open and closed world assumptions with contextualized and global interpretation of predicates. The MWeb framework is constructed, compatible and based on Rule Interchange Format (RIF) guidelines fostering immediate integration with RDF [12]. MWeb provides two semantics designated MWebWFS and MWebAS with a solid theory based on the two major semantics of extended logic programming, respectively, Well-Founded Semantics with Explicit Negation [1] and Answer Sets [9]. A compiler of MWeb into XSB Prolog is available-making use of the tabling features to guarantee termination of recursive rules with negation. It provides separate interface and implementation of rulebases with modular and independent compilation.

1 The system can be downloaded at http://centria.di.fct.unl.pt/~cd/mweb/
The major contribution of the paper is the specification of the semantics of ERDF reasoning entirely in the MWeb framework, including alignment with RIF, support of RDF and RDFS entailment, as well extensions to the original ERDF semantics for dealing with closed classes and properties. These results complement the mapping of simple modular ERDF ontologies into MWeb rulebases defined in [7]. Thus reasoning on simple modular ERDF ontologies can be achieved through our MWeb implementation ², and in particular supporting modular reasoning over RDF(S) ontologies.

The paper is organized as follows. In Section 2 we illustrate how simple modular ERDF ontologies are mapped into MWeb rulebases. Next Section 3 specifies the support of ERDF reasoning by MWeb logic rules instead of the formal model-theoretical presentation of [3]. The paper finishes with some conclusions.

2 The MWeb system is available at http://centria.di.fct.unl.pt/~cd/mweb

2 The MWeb Embedding of ERDF Ontologies

Simple modular ERDF ontologies [3] allow the combination of knowledge in different ontologies. Specifically, a simple modular ERDF ontology (SMEO) is a set of simple r-ERDF ontologies. The language of simple r-ERDF ontologies allows the use of ordinary triples \( s.p.o \) and negated triples \( \text{neg } s.p.o \) in the ERDF graph, where \( s \), \( p \) and \( o \) are respectively the subject, predicate and object of the statement. Additionally, it allows to construct programs using deductive rules to derive new (extended) triples by rules having bodies formed by combining the connectives \( \text{naf} \) (weak negation), \( \text{neg} \) (strong negation), and conjunction. Moreover, provides mechanisms to define modules of knowledge, which are described by an interface and formed by an ERDF graph and a program. Finally, it provides a means to query other rulebases via qualified literals of the form \( \text{Lit@URI} \) in rules. Details can be found in [3,7].

The MWeb framework requires for each rulebase (module of knowledge) the definition of an interface document and of the corresponding rulebase (logic) document. The MWeb interface is formed by a sequence of declarations. First, the name of the rulebase is stated via a rulebase declaration followed by an IRI. Optional base IRI and prefixes can be declared for simplifying writing of classes and property names, via a base and prefix declarations. Other interfaces may be recursively included via a special import declaration. This mechanism will be used to import the interfaces declaring the classes and properties defined by RIF, RDF, RDFS and ERDF. An optional vocabulary declaration can be used to list the vocabulary of the rulebase. Next, follow two blocks of declarations. The first block defines the predicates being defined in the MWeb rulebase, and correspond to a generalization of export declarations found in logic programming based languages. The second block correspond to generalization of import declarations. The interesting feature of the MWeb framework is that besides scope (i.e. internal, local, or global), different reasoning modes can be associated to predicates (i.e. definite, open, closed, or normal). This allows control of monotonicity of reasoning by the producer and consumer of the knowledge. In this work, all properties and classes are defined global (meaning that it can be defined in multiple rulebases) and normal (meaning that weak negation can be used). The semantics of all MWeb constructs can be found in [4] as well as additional motivation. In [7] it is defined the