Translating expressive ontology mappings into rewriting rules to implement query rewriting

Gianluca Correndo and Nigel Shadbolt

ECS Department, University of Southampton, SO17 1BJ, Southampton, UK
{gc3,nrs}@ecs.soton.ac.uk,
http://users.ecs.soton.ac.uk/{gc3,nrs}

Abstract. The semantics of ontology alignments, often defined over a logical framework, implies a reasoning step over huge amounts of data. This is often hard to implement and rarely scales on Web dimensions. This paper presents our approach for translating DL-like ontology alignments into graph patterns that can be used to implement ontological mediation in the form of SPARQL query rewriting and generation.

1 Introduction

In spite of the high expressive power of the languages used to define ontologies (e.g. RDFS, OWL, and SKOS), the wide range of vocabularies within the data cloud restrains the realisation of a machine-processable Semantic Web. Ontology matching is an important task within the data integration work flow and Semantic Web community provided automated tools for mining and describing correspondences between data vocabularies [4]. Among other tools, the Alignment API [2] provides a rich language to describe ontology alignments called *Expressive and Declarative Ontology Alignment Language* or EDOAL for short.

Our approach to implement data integration is based on the rewriting of SPARQL queries applying syntactic rules that modify their basic graph pattern in order to rework a given source query to fit a target data set.

2 SPARQL Query Rewriting

The approach adopted here is similar to the one used in peer data management systems [3] where queries can be rewritten multiple times, depending on where the query will be executed. The full description of the algorithm that rewrites SPARQL queries can be retrieved from a previous publication [1] while in this paper we will focus on how to translate EDOAL expressions into our internal representation.

An entity alignment EA codifies how to rewrite a triple for fitting a new ontology, it defines therefore a pattern rewriting and eventually a set of constraints over variables present in the alignment itself. The alignments so defined are directional (i.e. not symmetric). An entity alignment EA is defined as a triple $EA = \langle LHS, RHS, FD \rangle$. LHS is an atomic formula that contains no functional symbols. RHS is a conjunctive formula that contain no functional symbols. Finally FD is a set of functional dependencies that must hold in the rewriting process.

3 Support for EDOAL Alignments

The EDOAL language is meant to describe correspondences between entities of different ontologies and it uses DL-like primitives to describe those entities. The subset of EDOAL translated in our approach includes only those primitives which would affect the BGP section of a query leaving out: **concepts** (or **properties**) **disjunction** that would require an OPTIONAL statement; **attributes value restrictions** different from EQUAL that would require a FILTER statement; and **attribute occurrence restrictions** for similar reasons. The translation of EDOAL primitives into rewriting rules pattern can be provided as a denotational semantics over a simplified subset of the grammar that generates the entities' description in EDOAL (i.e. the expressions that are the subjects of the alignments). The semantics will define how rewriting patterns are created from inspecting the parsing tree of the entity description.

In Figure 1 it is described the minimal annotated grammar for the EDOAL language primitives (pseudo code) alongside with the translation in terms of triples patterns used by our approach. Every grammar rule is decorated with its denotational semantics v and numbered for later reference.

$$
\begin{aligned}
CE &::= class\ URI & v(CE) &= (Triple(?s, rdf:type, URI)) & (1)\\
CC &::= CE\ and\ CC' & v(CC) &= v(CE) + v(CC') & (2)\\
AEXP &::= pr\ URI & v(AEXP) &= (Triple(?s, URI, ?o)) & (3)\\
&\ |pr\ URI\ and\ AEXP' & v(AEXP) &= Triple(?s, URI, ?o) + v(AEXP') & (4)\\
&\ |pr\ URI\ comp\ AEXP' & v(AEXP) &= Triple(?s, URI, ?o) \circ v(AEXP') & (5)
\end{aligned}
$$

Fig. 1: EDOAL denotational semantics

4 Acknowledgements

This research has been supported by the EnAKTing project funded by the Engineering and Physical Sciences Research Council (EPSRC) under the contract EP/G008493/1. The information provided is the sole responsibility of the authors and does not reflect the Council's opinion.

References

1. G. Correndo, M. Salvadores, I. Millard, H. Glaser, and N. Shadbolt. SPARQL query rewriting for implementing data integration over linked data. In *1st International Workshop on Data Semantics (DataSem 2010)*, March 2010.
2. J. David, J. Euzenat, F. Scharffe, and C. T. dos Santos. The Alignment API 4.0. *Semantic Web*, 2(1):3–10, 2011.
3. A. Y. Halevy, Z. G. Ives, D. Suciu, and I. Tatarinov. Schema mediation in peer data management systems. In *Proceedings of the* 19^{th} *International Conference on Data Engineering*, pages 505–516, 2003.
4. Y. Kalfoglou and M. Schorlemmer. Ontology mapping: The state of the art. In Y. Kalfoglou, M. Schorlemmer, A. Sheth, S. Staab, and M. Uschold, editors, *Semantic Interoperability and Integration*, number 04391, 2005.