Chapter XLVII
Matching Relational Schemata to Semantic Web Ontologies

Polyxeni Katsiouli
University of Athens, Greece

Petros Papapanagiotou
University of Athens, Greece

Vassileios Tsetsos
University of Athens, Greece

Christos Anagnostopoulos
University of Athens, Greece

Stathes Hadjiefthymiades
University of Athens, Greece

INTRODUCTION

The Semantic Web (SW; Berners-Lee, Hendler, & Lassila, 2001) is already in its implementation phase and an indication of this is the intense research and development activity in the area of SW tools and languages. SW is based on metadata, which describe the semantics of the Web content. SW envisages the enrichment of data with semantics in order to be machine understandable and enable knowledge reasoning. The core element for achieving such Web evolution is ontology: "Ontology is an explicit specification of a conceptualization" (Gruber).

However, ontologies are just knowledge representation schemata and unless they are populated with real instances, they are of little value. A main problem of the SW is, currently, the lack of ontology instances. What is more challenging is the fact that the ontology elicitation process should be as automatic as possible in order to be effective and provide usable results in the short term. During the last decade, there have been proposed many approaches for producing such ontology instances (Staab & Studer, 2004). Some of them rely on controlled vocabularies and ontologies (e.g., WordNet) in order to create semantics metadata. Others try to provide mappings between existing information sources, thus adopting an information integration approach. The methodology that will be presented in the rest of this article is more close to the latter approach. Specifically, it is based on matching relational schemata to ontologies.

This seems a quite promising (semi-)automatic ontology population method since it is well known that a lot of information in the Web is stored in relational databases and forms the so-called "Deep Web." In order to manage such information with the SW technologies, it is very important to develop a methodology that performs the migration of the relational data to ontology instances. Data migration relies on a schema matching process between the relational schema and the target ontology. Schema matching is considered a task based on the fact that both schemata (relational and ontological) differ in structure, expressiveness, and reasoning capability. In this article, we propose a methodology for schema matching and present a tool, called RONTO (relational to ontology), which deals with the semantic mapping between the elements of a relational schema to the elements of an ontological schema.

BACKGROUND

Regarding the source schema, we assume a relational database (RDB) schema deployed on a typical relational database management system (RDBMS).

The conceptual schema (CS) of the target ontology (ONT) is expressed in a description logic (DL) language, due to the popularity of DLs in the SW community. DLs are knowledge representation languages (subsets of first-order logic) that express knowledge about concepts and conceptual hierarchies. An ontology expressed in DL language consists of concepts (classes) that can be described by various constructs and may have several restrictions (axioms). Concepts are categorized to primitive and defined concepts. Roles define binary relationships between concepts or between a concept and a datatype. Concepts and roles of an ontology can be both organized in hierarchical structures through the inclusion relation $\sqsubseteq$ (i.e., is-a, generalization, or subsumption). In OWL-DL, which is a sublanguage of the Web ontology language (OWL; Antoniou & van Harmelen, 2001), the term role is referred to as property. A property has a domain and range. When the domain and the range of a property are (primitive or defined) concepts, the property is called object-property. In case the range of a property is a literal (e.g., integer, string), the property is called datatype-property.

In order to better describe the proposed methodology, several intermediate modeling elements are introduced.

Definition 1. A candidate concept for an ontology concept *c*, CC_c , can be (a) an RDB relation, (b) an RDB view, or (c) a combination of them, which is structurally and semantically similar to the concept *c* of the target ontology.

Definition 2. A candidate datatype-property for a datatype-property *p*, CDP_p, is an attribute of an RDB relation,[1] which has the same (or a compatible) datatype, and is semantically similar to the datatype-property *p* of the target ontology.

Definition 3. A candidate object-property for an object-property *p*, COP_p, is (a) a referential constraint, (b) an RDB relation representing an N:M relationship between relations, or (c) an RDB attribute, which is structurally and semantically similar to the object-property *p* of the target ontology.

Definition 4. A candidate concept set (CCS_C) for an ontology concept c is the set of all CCs that can be computed for the concept *c*. Hence, CCS_c is defined as follows.

$$CCS_c \equiv \bigcup_{i=1}^{n} \{CC_{Ci}\} \, , \tag{1}$$

where n is the number of CCs for concept c. In case $CCS_c = \varnothing$, then no mapping exists for the concept c. Similarly, the candidate datatype-property set ($CDPS_p$) and the candidate object-property set

Related Content