Augmenting Linked (RDF) Data with Relational Data

Souripriya Das, Seema Sundara, Jagannathan Srinivasan
Oracle
One Oracle Drive, Nashua, NH 03062

ABSTRACT

Under *Linked Open Data* project, many RDF datasets are becoming available, to meet the objective of facilitating connecting related data. However, a vast majority of data, often referred to as the deep web, is not available as they reside in backend databases. Our position is that relevant portions of deep web data should also be made available as RDF datasets and OWL ontologies to enterprise level semantic web applications.

1. INTRODUCTION

Linking Open Data project [1] aims to "Create Knowledge out of Interlinked Data." It already hosts over 256 RDF datasets, with over 30 Billion triples describing resources from domains ranging from media, geography, life sciences, publications, and government [2].

However, a vast majority (estimated 99%) of data, often referred to as the *deep web* [3], is beyond the reach of state-of-the-art web-crawlers as they belong to dynamically generated web pages, where data resides in backend databases.

Although getting access to deep web data has been pursued by search engines, it also becomes important for semantic web applications.

Our contention is that as a next step, standards and mechanisms should be introduced that facilitate publishing relevant portions of relational data using RDF/OWL preferably without having to make a copy of data. The overall RDB2RDF architecture is shown in Figure 1.

This will facilitate integration of data across multiple schemas or databases and also between data stored in relational databases with RDF data.

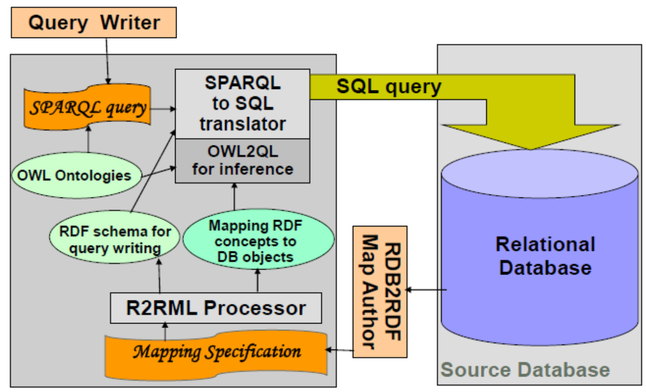

Figure 1. RDB2RDF Architecture

2. PUBLISHING RELATIONAL DATA AS RDF

The W3C RDB2RDF Working Group has been working towards standardizing publishing of relational data and relational schema into RDF datasets and OWL ontologies [4].

RBD2RDF Mapping

A simple direct mapping from a relational database schema to an RDF schema can be illustrated with a simple example as shown in Figure 2.

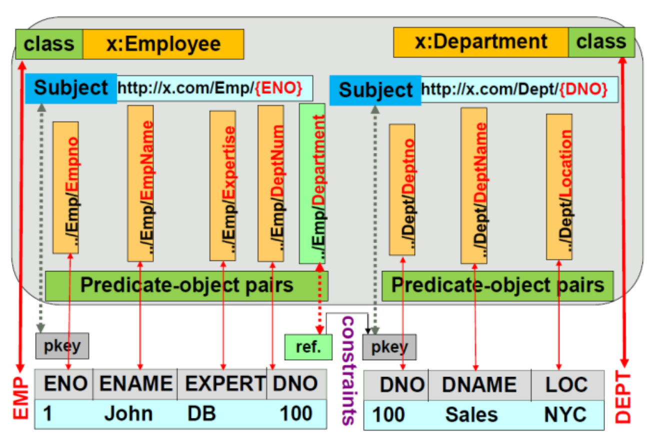

Figure 2. A sample RDB2RDF mapping.

In this example, EMP and DEPT tables store information about employees and departments. The EMP.DNO references DEPT.DNO column. Each table is mapped to a corresponding RDF/OWL class (`x:Employee` and `x:Department`) and each column is mapped to a predicate (e.g., column ENAME maps to `http://x.com/Emp/EmpName`). The referential constraint is mapped to an object property: `http://x.com/Emp/Department`.

The primary keys of the tables are used to create the subject IRIs (e.g., `http:/x.com/Emp/1` for the only row in EMP table). Each subject is asserted to be an instance of the class: for example, `<http:/x.com/Emp/1> rdf:type x:Employee`.

The value in each cell is mapped to an RDF term that make up the object for the corresponding subject and predicate thus generating the triples (e.g `<http:/x.com/Emp/1> <http://x.com/Emp/EmpName> "John"`).

Customized mappings provide flexibility of using table or view name or a SQL query as a logical table, and allow naming of predicates, subjects, and objects via templates. The upcoming W3C R2RML [5] Working Draft describes a mapping language that will allow such customization.

For cases, when mapping relational columns values to IRIs, the context (such as the column name) needs to be taken into account. For example, `'R'` in *color* column may map to `'Red'`, whereas `'R'` in *media* column may map to `'Rewritable'`.

Linking and Inferencing

Linking the generated virtual RDF datasets to other RDF datasets and OWL ontologies can be handled by using `owl:equivalentProperty`, `owl:equivalentClass`, and `owl:sameAs` assertions.

After linking, inference benefits can be availed by using backward-chaining reasoners such as for OWL 2 QL profile [6]. Note that one could use forward-chaining reasoners as well. However, that will involve materializing the inferred triples, which is may not be preferred as it can potentially generate huge volume of RDF triples.

SPARQL (against RDB + RDB2RDF mapping) to SQL (against RDB)

Assuming the RDB2RDF mapping is available, we need a SPARQL to SQL translator where the SPARQL is issued against the virtual RDF dataset defined by the RDB2RDF mapping on the relational database, and the translator converts it to a query against the underlying relational database. Note that this scheme would avoid materializing the RDF form of relational data. Also, the query against virtual RDF datatset and referenced OWL ontologies entailed using OWL 2 QL profile can be handled by use of backward chaining reasoner at query time.

CONCLUSIONS

We have participated actively in the RDB2RDF Working Group and would like to see the application of the R2RML mapping approach of viewing relational data as RDF for data integration purposes. The semantic web applications that can benefit by publishing relational data as RDF include web content providers, social networking, and homeland security applications.

REFERENCES

1. Linked Data: Connecting Distributed Data Across the Web, http://linkeddata.org/
2. The State of LOD Cloud. http://www4.wiwiss.fu-berlin.de/lodcloud/state/
3. Bergman, M., Michael K (August 2001). "The Deep Web: Surfacing Hidden Value". The Journal of Electronic Publishing 7 (1)
4. RDB2RDF Working Group. http://www.w3.org/2009/08/rdb2rdf-charter.html
5. R2RML Mapping Language. http://www.w3.org/2001/sw/rdb2rdf/r2rml/
6. OWL 2 Web Ontology Language Profiles http://www.w3.org/TR/owl2-profiles/