Benchmarking Data Exchange System Adopting Semantic Web Ontology

Harshalkumar Eknath Bhagat
Chetan Wasudeo Rane
Mayur Rajesh Gujarathi
Akshay Dipendra Pawar

Prof. Abhijit Chandrasen Manepatil

BE IT, Department of Information Technology,
SKN Sinhgad Institute of Technology & Science, Kusgaon (Bk) Lonavala, Maharashtra, India

ABSTRACT

The increasing popularity of the Web of Data is motivation for integrating semantic-web ontologies. Data exchange is one integration approach that aims to populate a target ontology by using data coming from one or more source ontologies. Currently, we have different systems that are suitable to perform data exchange amongst these ontologies; unfortunately, they have uneven performance, which makes it appealing assessing and ranking them from an empirical point of view. In the bibliography, there are variety of benchmarks, but they cannot be applied to this context because they are not suitable for testing semantic-web ontologies. In this article, we are representing MostoBM which is also a benchmark for testing data exchange systems in the context of such ontologies. It provides us three real-world and seven synthetic data exchange patterns, which can be used into a variety of scenarios using some parameters. These scenarios help to analyze how the performance of data exchange systems improves the exchanging ontologies are scaled in structured and/or data. we are providing an evaluation methodology to compare data exchange systems side by side and to make informed and statistically-sound decisions regarding:
1) which data exchange system performs better; and
2) how the Performance of a system is influenced by the parameters of our benchmark.

Keyword: - *Semantic web data, Data exchange, Resource Description Framework (RDF)*

---

1. INTRODUCTION

The goal of the Semantic Web is to give the current Web with metadata, means it is giving web of data. Now a days increasing popularity of semantic-web ontologies, in the context of Linked Open Data, and government, life sciences, geographic, media, or publications. Semantic-web ontologies build on the RDF, RDFS and OWL ontology languages for modeling structure and data, and the SPARQL query language to query them. Here, we are focusing on data exchange, which aims to populate a target ontology using data that come from one or more source ontologies. When using SPARQL queries, data exchange is performed by executing a number of CONSTRUCT queries that extract data from the source ontologies, transform them, and load the results into the target ontology.
Our benchmark provides a catalogue of three real-world and seven synthetic data exchange patterns; seven parameters to construct scenarios that are instantiations of the patterns; and a publicly available tool that facilitates the instantiation of the patterns, and the gathering of data about the performance of systems. The three real-world patterns are relevant data exchange problems in the context of Linked Open Data, whereas the seven synthetic ones are common integration problems that are based on current proposals in the ontology evolution context and on our experience regarding real-world information integration problems Furthermore, the parameters allow to scale the

structure of source and target ontologies, the data of the source ontology, and the SPARQL queries to perform data exchange for the synthetic patterns.

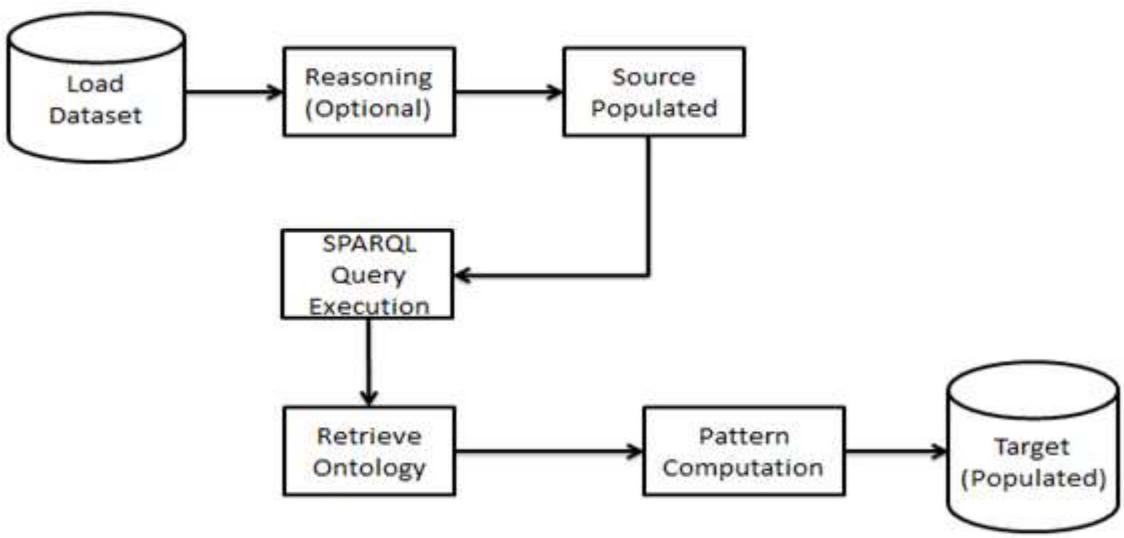
2. EXISTING SYSTEM

Benchmark provides a catalogue of several synthetic data exchange patterns, such as ontology evolution, which can be seen as a data exchange problem in which the source is the original ontology, and the target is an evolution of the source ontology. For example, adding or removing classes, subclasses, properties, sub properties, property domains, or property ranges. These changes can be seen as the simplest operations building on which the evolution of ontology may be specified. Some combinations of changes are very frequent in practice, which motivated the authors to devise a catalogue of common composite changes. Unfortunately, the specification of the evolution of an ontology does not take how data are exchanged into account. Our catalogue of synthetic patterns summarizes common changes we have found in real-world information integration problems: not only specify they how the structure of the source ontology evolves, but also how source data must be exchanged by means of queries of the Construct type.

3. PROPOSED SYSTEM

In this project, we present a benchmark for testing data exchange systems in the context of ontologies and query engines as well as its graphical representation. Our benchmark provides a catalogue of three real-world and seven synthetic data exchange patterns; seven parameters to construct scenarios that are instantiations of the patterns; and a publicly available tool that facilitates the instantiation of the patterns and the gathering of data about the performance of systems. In addition, we provide an evaluation methodology that allows comparing data exchange systems side by side. Regarding the catalogue, the three real-world patterns are relevant data exchange problems in the context of Linked Open Data, whereas the seven synthetic patterns are common integration problems that are based on current approaches in the ontology evolution context, and our experience regarding real-world information integration problems. This catalogue is not meant to be exhaustive: the patterns described in this project are the starting point to a community effort that is expected to extend them. It provide user the exact data the user looking for, in minimum amount of time.

4. ARCHITECTURE



The architecture is consist of following parameters

4.1 Load Dataset

Load dataset represents loading of source and target ontologies. It also represents set of SPARQL queries from a persistent storage into appropriate internal data structure.

4.2 Reasoning over Source

Reasoning over Source shows that how to make knowledge explicit and this step is not necessary. If the knowledge is explicit the no need to perform this step.

4.3 SPAQL Query execution

Here SPARQL query execution is done on source ontology to create instance of target ontology. There is no order for query execution.

4.4 Reasoning over target

Reasoning over target is also same as reasoning over source. This is used to make target ontology knowledge explicit. If it is already explicit then no need to perform this step.

4.5 Unloading:

Unloading represents saving of target ontology to persistent storage

5. THREE REALWORLD PATTERN

We have selected this pattern as our benchmark's part because they used to represent the integration problem in the context of Linked Open Data.

There are three real world patterns:

- Evolution of an ontology
- Vocabulary adaption
- Publication of Linked Open Data

5.1 Evolution of an ontology

This context defines the Source ontology and Target Ontology.

Source Ontology: The ontology before changes are applied.

Target Ontology: The ontology after changes are applied.

5.2 Vocabulary Adaption

In this context it is necessary that vocabulary of Target Ontology should be same as the Source ontology vocabulary.

5.3 Publication of linked Open Data

Many existing ontologies are not following the principles of Linked Open Data initiative and usually we have to transform them to comply with those principles.

6. SEVEN SYNTHETIC PATTERN

Synthetic pattern represents a common and relevant integration problem. Our benchmark provides seven synthetic data exchange pattern. Each pattern represents intention of change.

6.1 Lift Properties

The intention of the change is that the user wishes to extract common properties to a super-class in hierarchy. Therefore, the data properties of set of subclasses are moved to a common superclass.

6.2 Sink Properties

The intention of the change is that the user wishes to narrow the domain of properties. Therefore the data properties of the superclass are moved to a number of subclasses.

6.3 Extract Subclasses

The intention of the change is that the user wishes to specialize a class. Therefore, a subclass is split into several subclasses and the domain of the target data properties is selected amongst the subclasses.

6.4 Extract Superclasses

The intention of the change is that user wishes to generalize a class. So, a class is split into several superclasses, and data properties are distributed amongst them.

6.5 Extract Related Classes

The intention of the change is that the user wishes to extract a number of classes building on a single class.Therefore,the data properties that have single class as domain change their domain by new classes which are related to the original one by a number of object properties.

6.6 Simplify Specialization

The intention of the change is that user wishes to flat a hierarchy of classes.Thus a set of specialized classes are flattened into a single class.

6.7 Simplify Related Class

The intention behind the change Is that the user wishes to join a set of classes that are related by object properties. Therefore, several classes are transformed into one class that aggregates them all.

7. RESOURCE DESCRIPTION FRAMEWORK (PROPOSED OUTPUT)

RDF is nothing but graphical representation of outcome information. It helps user to understand outcome information easier manner without time consuming. E.g. If user want to search information about person, he will get result as

8. CONCLUSIONS

In this project, our benchmark provides a catalogue of three real-world and seven synthetic data exchange patterns. The former are relevant data exchange problems in the context of Linked Open Data. The latter are common integration problems based on current approaches in the context of ontology evolution, and on our experience in information integration. This catalogue of patterns is not meant to be exhaustive: we expect a community effort to extend them. These patterns can be instantiated into synthetic scenarios by means of seven parameters that allow controlling the construction of both the structure and/or data of ontologies. The scaling of the patterns helps analyze the performance of systems when data exchange problems increase their scale in structure data. Finally, our benchmark provides an evaluation methodology that allows comparing systems side by side, and to make informed and statistically sound decisions about their performance. It is applied to a number of patterns and systems, and allows ranking which system performs better or how the performance of a system is influenced by the parameters.

9. REFERENCES

[1] L. Stojanovic, A. Maedche, B. Motik, and N. Stojanovic. User-driven ontology evolution management. In EKAW, pages 285–300, 20

[2] Y. Guo, Z. Pan, and J. Heflin. LUBM: A benchmark for OWL knowledge base systems. J. Web Sem., 3(2-3):158–182, 2005

[3] B. Alexe, W. C. Tan, and Y. Velegrakis. STBenchmark: Towards a benchmark for mapping systems. PVLDB, 1(1):230–244, 2008

[4] M.Klusch,B.Fries,andK.P.Sycara,OWLS-MX:AHybridSemanticWebService Matchmaker for OWL-S Services, J. Web Semantics, vol. 7, no. 2, pp. 121-133, 2009.

[5] C.R. Rivero, I. Hernandez, D. Ruiz, and R. Corchuelo, On Benchmarking Data Translation Systems for Semantic-Web Ontologies, Proc. Conf. Information and Knowledge Management (CIKM), pp. 1613-1618, 2011.

[6] C.R. Rivero, D. Ruiz, and R. Corchuelo, On Benchmarking Data Translation Systems for Semantic-Web Ontologies, technical report, Univ. of Sevilla, http://tdgseville.info/Download.ashx?id=205, 2012.

[7] C.R. Rivero, A. Schultz, C. Bizer, and D. Ruiz, Benchmarking the Performance of Linked-Data Translation Systems, Proc. Intl Workshop Linked Data on the Web (LDOW), 2012