Supporting Geosciences Web Services Metadata Management and Discovery

Pearl Brazier, Artem Chebotko †, Eric Gonzalez, Andrey Kashlev, Anthony Piazza
Department of Computer Science
The University of Texas-Pan American
1201 West University Drive, Edinburg, TX 78539-2999, USA
† Corresponding author. Email: artem@cs.panam.edu

***Abstract* — Geosciences Web portals are becoming increasingly important for supporting geoscientists in their research. The GEO-SEED portal is a repository of geosciences web services metadata, represented in Resource Description Framework (RDF), which supports management and discovery by machines and automated agents. This project uses SPARQL, the W3C standard for querying RDF, to support discovery of web services. SPARQL query performance becomes more critical as the amount of RDF metadata in the repository increases. Most existing RDF storage systems are based on relational database technology. The problem with most of these systems is that their query performance is limited by the use of fixed schema mapping strategies. In this paper, we present our system, called S2ST, which addresses this issue in the context of geosciences web services metadata management. Our experiments show that S2ST provides better SPARQL performance than existing relational RDF storage systems.**

Keywords-component; web services; metadata management; RDF; SPARQL

I. INTRODUCTION

The GEO-SEED [1] project collects metadata on geosciences web services and stores it as RDF. The GEO-SEED team evaluated many RDF storage systems and ultimately decided to use S2ST because of the unique benefits that it provides. S2ST is able to efficiently store and query large Semantic Web datasets in a relational database and contains algorithms that provide mappings between RDF and relational data models. While several relational RDF storage systems have been developed worldwide, S2ST is the first and only one that supports nearly 20 RDBMS backends, user-customizable schema mappings, schema-independent data mappings, and semantics-preserving query translation. The data loading and query performance of existing systems is limited due to fixed database schemas. S2ST provides substantial improvements via schema customization, data redundancy techniques and query optimizations. The use of generic schema mapping, data mapping, and query mapping algorithms means that S2ST is able to subsume numerous existing and even future database schema generation approaches including schema-oblivious, schema-aware, data-driven, and hybrid schema mappings. While S2ST is based on a formal mathematical model realized as computer data structures and algorithms, its complexity is hidden from end users via an intuitive Web-based interface. In this paper, we demonstrate S2ST's functionality by storing and querying RDF datasets from GEO-SEED and evaluate different approaches for improving the performance of geosciences metadata management and discovery.

II. WEB SERVICES DISCOVERY ONTOLOGY (WSDO)

GEO-SEED's ontology, called Web Services Discovery Ontology, provides a vocabulary encoded in standard Semantic Web languages OWL and RDF for metadata acquisition. OWL and RDF allow Web services descriptions to be readily interpreted by machines or automated agents to enable automatic Web services discovery. The descriptors of a web service are organized into six categories shown in Figure 1.

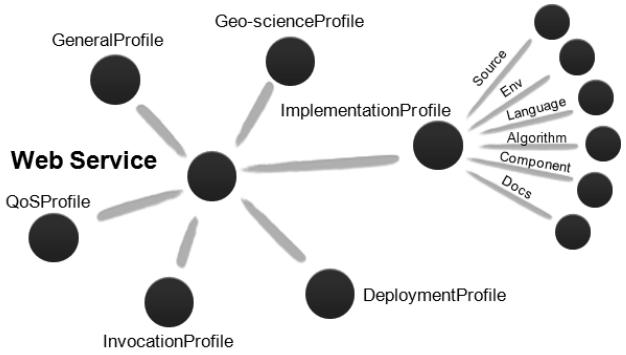

Figure 1. GEO-SEED Web Services Ontology

III. DATABASE SCHEMA GENERATION

Mapping RDF data to the relational model can be accomplished using a number of different strategies. Determining which schema mapping strategy is appropriate for a dataset depends largely on the type of queries one intends to execute. S2ST provides support for all known schema mapping strategies, unlike existing relational RDF storage systems that rely on fixed mappings and support only one or two schema mapping strategies. S2ST is the first relational RDF storage system that allows users to define his or her mappings, called user-customizable schema mappings, visually through a web interface. S2ST can use the following five approaches to generate database schemas for storing GEO-SEED datasets.

- **Schema-Oblivious**: A single general-purpose relation is used to store all RDF triples.

- **Schema-Aware**: An ontology, such as WSDO, is used to generate database schemas with relations that correspond to classes and properties in the ontology.
- **Data Driven**: Patterns in RDF data are used to generate relations on the fly.
- **User-Customizable:** Relations are defined by a user via generic mechanism (triple patterns) that describes what triples can be stored in these relations.
- **Hybrid**: A database schema is generated by any combination of the previous approaches.

S2ST has two notions of schema: a logical schema that determines database schema for a particular approach and a physical schema that realizes the logical schema in a particular SQL dialect. S2ST can generate DDL (Data Definition Language) and DML (Data Manipulation Language) for nearly twenty different database backends including Oracle, MySQL, SQL Server, and DB2, enabling the user to define multiple physical schemas based on a single logical schema.

IV. Data Loading

S2ST supports three different strategies for populating a physical schema with RDF data.

- **Flat File**: Generates a single file containing SQL INSERT statements for each tuple.
- **JDBC Batch**: Caches tuples in main memory and inserts them in large chunks allowing efficient operations on relations and indices.
- **Bulk**: Generates one or more files that can be inserted using database-specific bulk loading facility.

The Bulk loading strategy usually results in the best performance for data loading operations on large datasets.

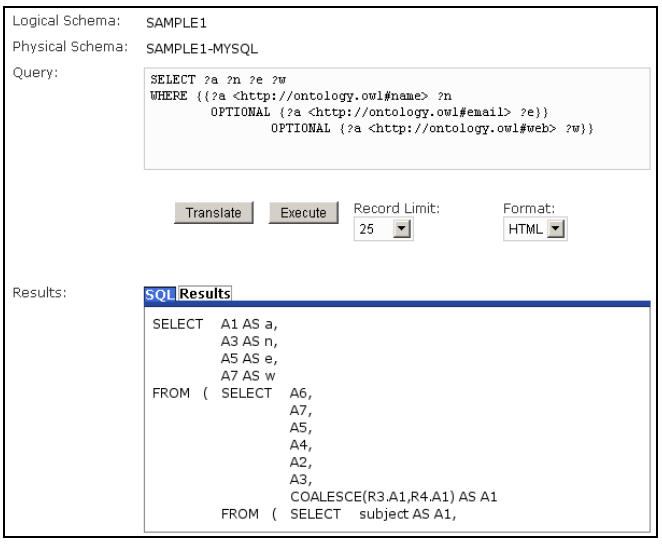

Figure 2. Query Translation in S2ST

V. Query Translation

RDF queries are expressed in the SPARQL query language. S2ST translates SPARQL queries into SQL queries and executes them over a relational database that stores the RDF data. S2ST's SPARQL-to-SQL translation algorithm is mathematically proven to be semantics preserving [2], thus ensuring the equivalence of SPARQL and SQL query results. Figure 2 contains a screenshot of S2ST after it has translated a SPARQL query to SQL.

VI. Performance Study

We benchmarked S2ST on a sample GEO-SEED dataset that describes 10,000 web services using six test queries commonly executed by geoscientists. In S2ST, we deployed four physical schemas in MySQL 5.1 based on schema-oblivious, schema-aware, data-driven, and user-customizable schema mapping approaches. Our experiments show that the user-customizable schema mapping strategy (only supported in S2ST) provided the best performance for all test queries. Figure 3 shows that schema customization can significantly improve query performance. The experiments were conducted on a PC with 3.00 GHz Intel Core 2 CPU, 4 GB RAM, and 750 GB disk space running MS Windows XP Professional.

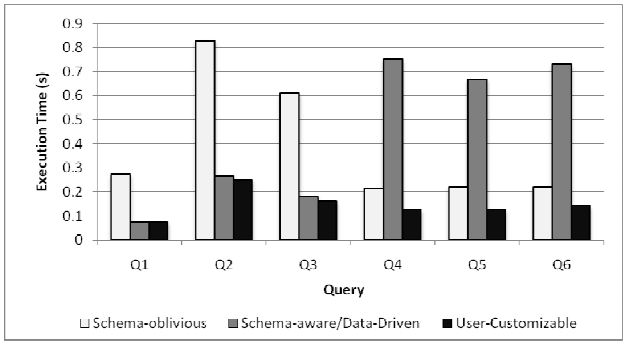

Figure 3. Performance comparison of schema mapping strategies

VII. Conclusion

We demonstrated S2ST's novel schema generation, data loading and query translation techniques based on the sample data from the GEO-SEED project. We used WSDO to generate database schema in S2ST, loaded 10,000 web services RDF descriptions, and queried this dataset using the SPARQL query language. Our performance study showed that unique customizations that are only available in S2ST can significantly improve query execution times.

References

[1] Pearl Brazier, Artem Chebotko, Anthony Piazza, Ann Q. Gates, and Leonardo Salayandia, "Web 2.0 and Semantic Web Portal for Annotation and Discovery of Web Services in Geosciences", in *Proc. of the International Conference on Semantic Web and Web Services (SWWS)*, 2009.

[2] Artem Chebotko, Shiyong Lu, and Farshad Fotouhi, "Semantics Preserving SPARQL-to-SQL Translation", *Data & Knowledge Engineering*, 68(10), pp. 973-1000, 2009.