The ICS-FORTH RDFSuite: High-level Scalable Tools for the Semantic Web

Sofia Alexaki Nikos Athanasis Vassilis Christophides
Greg Karvounarakis Aimilia Maganaraki Dimitris Plexousakis
Institute of Computer Science, FORTH, Heraklion, Greece
{alexaki, athanasi, christop, gregkar, aimilia, dp}@ics.forth.gr
Karsten Tolle
Johann Wolfgang Goethe-University, Frankfurt/Main, Germany
tolle@dbis.informatik.uni-frankfurt.de

Abstract

We present an overview of RDFSuite, a suite of high-level, scalable tools for validating, storing and querying RDF schemas and resource descriptions. RDFSuite addresses the need for effective and efficient management of large volumes of RDF metadata as required by real-scale Semantic Web applications. The development of RDFSuite has been supported in part by the EU projects C-Web (IST-1999-13479) and MesMuses (IST-2001- 26074) and it is available for download under an Open Source Software License from the Web site http://139.91.183.30:9090/RDF/.

Keywords: Metadata on the Web, Languages and Standards, Storage and Querying, Semantic Web

1. Introduction and Overview

In the next evolution step of the Web, termed the Semantic Web, vast amounts of information resources (data, documents, programs) will be made available along with various kinds of descriptive information, i.e., metadata. Better knowledge about the meaning, usage, accessibility, quality or validity of web resources will considerably facilitate automated processing of available Web content/services. The Resource Description Framework (RDF) [1] enables the creation and exchange of resource metadata as normal Web data. To interpret these metadata within or across user communities, RDF allows the definition of appropriate schema vocabularies (RDFS) [2]. Managing voluminous RDF description bases and schemas with existing low-level APIs and file-based implementations does not ensure fast deployment and easy maintenance of real-scale Semantic Web applications such as **Knowledge Portals** and **E-Marketplaces**. Still, we want to benefit from database technology in order to support **declarative access** and **logical and physical RDF data independence**. This is the main design choice of ICS-FORTH RDFSuite, which comprises: the Validating RDF Parser (VRP), the Schema-Specific Data Base (RSSDB) and an interpreter for the RDF Query Language (RQL). In this way, Semantic Web applications have to specify in a high-level language - like RQL - only which resources need to be accessed, leaving the task of determining how to efficiently store or access their descriptions to RSSDB. Figure 1 depicts the architecture of RDFSuite.

2. The Validating RDF Parser (VRP)

The ICS-FORTH Validating RDF Parser (V2.1) is a tool for analyzing, validating and processing RDF schemas and resource descriptions. The Parser analyses syntactically the statements of a given RDF/XML file according to the RDF Model & Syntax Specification (revised syntax of December 18, 2001). The Validator checks whether the statements contained in both RDF schemas and resource descriptions satisfy the semantic constraints derived by the RDF Schema Specification (RDFS). Unlike other available RDF parsers, VRP is based on standard compiler generator tools for Java, namely CUP (0.10j) and JFlex (1.3.5) similar to YACC/LEX. The stream-based parsing support of JFlex and the quick LALR grammar parsing of CUP ensure good performance, when processing large volumes of RDF descriptions. For this purpose, the VRP validation module relies on an original object representation, separating RDF schemas from their instances. VRP has been successfully used with a number of RDF schemas from existing Semantic Web applications, hosted by the ICS-FORTH Schema Registry at http://139.91.183.30:9090/RDF/Examples.html.

VRP supports embedded RDF in XML or HTML documents, XML Schema Data Types and Unicode character set. Users are provided with the ability to activate or deactivate semantic constraints against which validation is performed and hence, to configure the parser according to their needs. An additional substantial feature is the fetching of remote namespaces and their inclusion and integration in VRP's internal RDF model. VRP provides various options for **debugging**, **serialization** under the form of triples or graphs, as well as complete **statistics** (validation / parsing time, characteristics) of validated schemas and resource descriptions. It is easy to use as a standalone application (GUI provided) and to integrate with other applications (RDF Model Construction and Validation Java APIs).

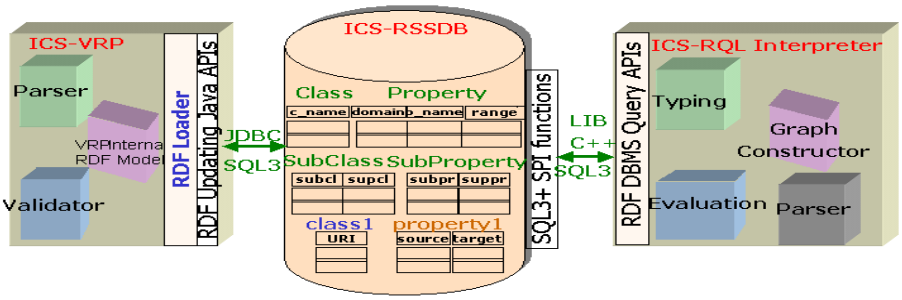

Figure 1: The ICS-FORTH RDFSuite Architecture

3. The RDF Schema-Specific Data Base (RSSDB)

RSSDB (V1.5) is a persistent RDF Store for loading resource descriptions in an object-relational DBMS (SQL-3 compliant ORDBMS) by exploiting the available RDF schema knowledge. It preserves the flexibility of RDF in refining schemas and enriching descriptions at any time, whilst it can store resource descriptions created according to one or more associated RDF schemas. The main goal of the design of RSSDB is the separation of the RDF schema from data information, as well, as the distinction between unary and binary relations holding the instances of classes and properties respectively. The experiments we have carried out (using as testbed the Open Directory RDF dump comprising about 6 million of triples), illustrate that RSSDB yields considerable performance gains in query processing and storage volumes as compared to other triple-based RDF Stores [3].

RSSDB has been implemented on top of the PostgreSql (v7.1.3) ORDBMS. It comprises a Loading and an Update module, both implemented in Java using a number of primitive methods (APIs) for inserting, deleting, and modifying RDF triples. Access to the ORDBMS relies on the JDBC interface (v2.0) for interoperating with various commercial or public domain ORDBMS. The most distinctive RSSDB feature is the **customization** of the database representation according to the employed meta-schemas (RDF/S, DAML-OIL), the peculiarities of RDF schemas and description bases as well as the target query functionality. In addition, the RSSDB Loader supports **incremental loading** of distributed namespaces by automatically detecting changes in already stored RDF schemas or data.

4. The RDF Query Language Interpreter (RQL)

RQL [4] is a typed language, following a functional approach (as in ODMG OQL or W3C XQuery). RQL relies on a formal graph model (as opposed to other triple-based RDF query languages) that captures the RDF modeling primitives and permits the interpretation of superimposed resource descriptions by means of one or more schemas in a variety of application contexts (e.g., advertisements, recommendations, copyrights, content ratings, push channels, etc.). The novelty of RQL lies in its ability to **seamlessly combine schema and data querying**.

The RQL Interpreter (V1.5) has been implemented in C++ on top of PostgreSql v7.1.3 using a standard client-server architecture (XDR-based) for Solaris and Linux platforms. It consists of three modules (a) the Parser, analyzing the syntax of queries; (b) the Graph Constructor, capturing the semantics of queries in terms of typing and interdependencies of involved expressions; and (c) the Evaluation Engine, accessing RDF descriptions from the underlying database via SQL3 queries. It fully supports (i) **XML Schema data types** (for filtering literal values) (ii) **grouping** primitives (for constructing nested XML results) (iii) **aggregate** functions (for extracting statistics) and (iv) **recursive traversal** of class and property hierarchies (for advanced matchmaking of descriptions). It pushes as much as possible query evaluation to the underlying DBMS (in order to benefit from robust SQL3 query engines and DB indices) while it is easy to couple with commercial ORDBMSs (using SQL3/C++ RDF Querying APIs) as well as to integrate with various Web Application Servers (using C++ or Java RQL clients). Finally, it provides a generic RDF/XML form of query results that can be processed by standard XSL/XSLT scripts for customized rendering. An online demo with several representative RQL queries is available at http://139.91.183.30:8999/RQLdemo/.

5. REFERENCES

1. O. Lassila and R. Swick. RDF Model and Syntax Specification. W3C Recommendation, 1999.
2. D. Brickley and R.V. Guha. RDF Schema Specification 1.0. W3C Candidate Recommendation, 2000.
3. S. Alexaki, G. Karvounarakis, V. Christophides, D. Plexousakis and K. Tolle. The ICS-FORTH RDFSuite: Managing Voluminous RDF Description Bases. 2nd International Workshop on the Semantic Web, Hong-Kong, pp. 1-13, 2001.
4. G. Karvounarakis, V. Christophides, D. Plexousakis, S. Alexaki and M. Scholl. RQL: A Declarative Query Language for RDF. to appear in Proceedings of WWW2002, 11th International Conference on the WWW, Hawaii, 2002.