# Querying Semantic Web Data Sets by Using SPARQL

Necula, Sabina-Cristiana

Alexandru Ioan Cuza University of Iasi

May 2011

# Querying Semantic Web Data Sets by Using SPARQL

[1]Sabina-Cristiana NECULA

[1] Alexandru Ioan Cuza University of Iaşi

**Abstract:** *This paper presents with examples some queries made on data sets using SPARQL. We treat the problem of available standards and tools. We show how we configure a SPARQL endpoint. Also the article treats the problem of describing data by using Resource Description Format.*

**Keywords:** Semantic web technologies, semantic search, SPARQL, RDF
**JEL Codes:** L86, D83

## 1. INTRODUCTION

The Semantic Web is the extension of the World Wide Web that enables people to share content beyond the boundaries of applications and websites. It has been described in rather different ways: as a utopic vision, as a web of data, or merely as a natural paradigm shift in our daily use of the Web. Most of all, the Semantic Web has inspired and engaged many people to create innovative semantic technologies and applications (Wikipedia). The core technological building blocks are now in place and widely available: ontology languages, flexible storage and querying facilities, reasoning engines, etc. Standards and guidelines for best practice are being formulated and disseminated by the World Wide Web Consortium (W3C) - Bishr (1998).

We address in this paper the problem of semantic search. The field of economy and finance is a conceptually rich domain where information is complex, huge in volume and a highly valuable business product by itself. The Security Exchange Commission developed a vocabulary useful for financial reporting (2000).

This paper has 4 Sections. Section 1 presents an introduction. Section 2 contains some aspects related to the current Semantic Web standards addressed by our paper. Section 3 presents with examples the uses of vocabularies, Resource Description Format (RDF), and SPARQL Protocol and RDF Query Language (SPARQL) for querying data sets. Section 4 treats the main conclusions.

## 2. SEMANTIC WEB STANDARDS

TheW3C has defined two languages for the Semantic Web: RDF and OWL. The Resource Description Framework (RDF) as IBM quotes (2000) plays a basic role by allowing the expression of statements, in the form of subject-predicate-object triples. The Web Ontology Language (OWL) as Olsen quote (2002) allows the expression of ontologies, which define the meaning of terms used in RDF statements. Simple ontologies can already be expressed using the RDF Schema (RDFS) vocabulary as IBM quotes (2000).

Although the standard syntax for RDF and OWL uses XML, it should be noted that the meaning of RDF and OWL knowledge bases is independent of XML and abstracts from the XML serialization used. Here the notion of RDF graph as Pickett and Hamre quote (2002) plays a role.

In the case of dealing with multiple ontologies, applications also require to integrate such ontologies.

A Resource Description Foundation (RDF) vocabulary is a defined set of predicates that can be used in an application. One can define a vocabulary for an application by creating an ontology file, which is an RDF document that contains all possible predicates for an application. Ontology not only defines the predicates themselves, but defines the data type of each predicate and the relationship, if any, of one predicate to another.

RDF vocabularies can describe relationships between vocabulary items from multiple vocabularies that have been developed independently.

Some analysis has been done on the topic of RDF stores which can handle large datasets. (A large dataset in this context is usually considered one on the order of tens or hundreds of millions of triples). The W3C ESW wiki contains information on a variety of RDF stores which can scale to large numbers of triples, but does not speak specifically to the performance of SPARQL queries against these stores.

The performance of a SPARQL query against any particular dataset depends not only upon the size of the dataset but also on the nature of the dataset's storage (a relational store, a native triple store, LDAP, etc.), the complexity of the query itself, optimizations in use by the SPARQL engine, the distribution of the data, and other environmental factors. To date, little work has been done in analyzing SPARQL query performance in particular, and the field of SPARQL query optimization is relatively inchoate.

SQWRL (Semantic Query-Enhanced Web Rule Language) is a SWRL-based language for querying OWL ontologies. It provides SQL-like operations to retrieve knowledge from OWL.

## 3. AVAILABLE SEMANTIC WEB TOOLS

There are a variety of tools available for producing, manipulating and exploring linked data. Here, we provide an overview of tools available, categorized by function.

Linked data can either be created from scratch or via conversion from a legacy format e.g. relational data, XML data, Microsoft Excel spreadsheets, or text files. Conversion can be done either by hand or using a tool. Conversion tools exist to both convert data prior to its publication or to convert it when it is needed.

Convertor tools and related resources include:

- B2RDF (http://sourceforge.net/projects/db2rdf/, 17/07/09, GPL). A tool that converts relational data to RDF. It also supports a SPARQL endpoint for querying the data.

- GRDDL (http://www.w3.org/TR/grddl/, 11/09/07, W3C document licence). A W3C specification that defines a method for exposing XML as RDF via XSLT (a technology for mapping XML-to-XML).
- RDFTEF (http://rdftef.sourceforge.net/, 04/10/05, GPL). A tool that converts XML documents consisting of a subset of TEI (Text Encoding Initiative) XML into RDF. TEI (http://www.tei-c.org/index.xml) is a standard for representing texts in digital form and has been used for epigraphic data which is the application area for SPQR (e.g. the IAphrodisias dataset)
- Krextor (http://trac.kwarc.info/krextor, 12/08, Lesser-GPL). An XSLT framework for XML-to-RDF conversion which can be invoked via shell scripts or Java.

For storing linked data, a number of established products exist, including:

- Virtuoso (http://virtuoso.openlinksw.com/, 22/09/10, versions available licensed under commercial license or GPL (OpenLink Virtuoso)). A data server that supports various data representations including relational, XML and RDF. It provides an RDF triple-store and supports SPARQL endpoints and Sesame and Jena APIs allowing it to be used with those products.
- Sesame (http://www.openrdf.org/, 12/09, BSD-style Sesame licence). An RDF framework supporting SPARQL and other query languages.
- Jena (http://jena.sourceforge.net/, 18/02/11, BSD). A semantic web framework with Java APIs for RDF manipulation and serialization to a relational database. Unlike Sesame it has support for OWL.
- Talis (http://www.talis.com, 19/01/11, custom pricing model, free up to certain data volumes). A semantic web application platform offered as a service. Talis will host open source linked data and provide a SPARQL endpoint, content negotiation and access control.
- AllegroGraph (http://www.franz.com/agraph/allegrograph/, 17/01/11, free and commercial licences, closed source - an RDF database with support for SPARQL queries and Prolog reasoning.
- Mulgara (http://www.mulgara.org/, 01/10/10, Open Software Licence). A 100% Java RDF database which supports REST interfaces for SPARQL and also to insert update or delete triples.
- Cliopatria (http://cliopatria.swi-prolog.org/home, 27/01/11, free and open source, license unknown). An RDF database with web server, user management, SPARQL query support and Prolog reasoning.

## 4. A PRACTICAL EXAMPLE IN REALIZING SPARQL QUERIES

We start by describing using RDF graphs a datasets that we want to query (Figure 2).

We used Security Exchange Commission data sets available in n3 format at http://www.rdfabout.com/demo/sec/. Besides this data about companies we used some financial data available in Excel files at http://pages.stern.nyu.edu/~adamodar/New_Home_Page/data.html.

RDF triples can be described using turtle syntax. We present a simple example in Figure 1.

```
:company1     ns:name    "China Infrastructure
Invsmt"
```

Fig. 1. An RDF triple in Turtle syntax

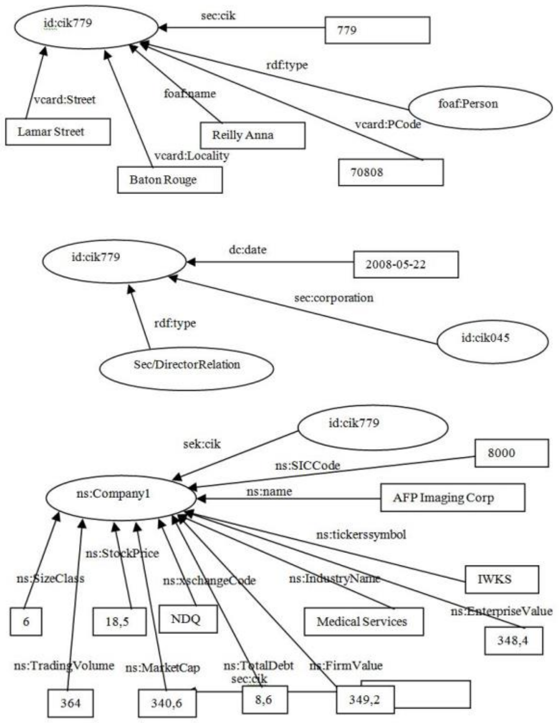

Fig. 2. RDF classes, instances and properties

Writing queries by making use of SPARQL doesn't mean anything else than asking values for objects from subject-predicate—object triple. An example is given in Figure 3.

```
ns:company        foaf:name ?name.
```

Fig. 3. An SPARQL triple pattern, with a single variable

When writing queries all parts of a triple can be requested. An example is given in Figure 4.

```
?company foaf:name ?name.
```

Fig. 4. An SPARQL triple pattern

On our datasets if we want to retrieve all variables from sec vocabulary that are of the Directorrelation kind of type we will write a query that looks like the one from Figure 5.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-
syntax-ns#>
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
prefix dc:          <http://purl.org/dc/elements/1.1/>
prefix vcard:       <http://www.w3.org/2001/vcard-
rdf/3.0#>
prefix :            <http://example.org/company/>
prefix ns:
<http://sandbox.metadataregistry.org/uri/schema/fin>
SELECT *
WHERE {

  ?subject sec:cik ?cik;
```

```
               rdf:type sec:DirectorRelation.
}
```

Fig. 5. A SPARQL query retrieving all variables

If we want to query what are the uri and the StockPrice of highest SockPriced companies we will write a query that look like query depicted in Figure 6.

```
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
prefix dc:         <http://purl.org/dc/elements/1.1/>
prefix vcard:      <http://www.w3.org/2001/vcard-
rdf/3.0#>
prefix :           <http://example.org/company/>
prefix ns:
<http://sandbox.metadataregistry.org/uri/schema/fin>

SELECT ?company ?StockPrice
WHERE {
         ?company ns:StockPrice ?StockPrice.
}
ORDER BY DESC(?StockPrice)
LIMIT 10
```

Fig. 6. A SPARQL query that returns uri and StockPrice of the ten companies that have the highest Stock Price

In the next example we will query what are the companies that have created their uri between May 1st 2008 and December 12th 2011.

```
PREFIX xsd: http://www.w3.org/2001/XMLSchema#
prefix dc:         <http://purl.org/dc/elements/1.1/>
prefix vcard:      <http://www.w3.org/2001/vcard-
rdf/3.0#>
prefix :           <http://example.org/company/>
prefix ns:
<http://sandbox.metadataregistry.org/uri/schema/fin>

SELECT ?name
WHERE {
  ?id dc:date ?date;
          sec:corporation ?corporation.
  ?company ns:name ?name.
  FILTER (?date > "2008-05-01"^^xsd:date &&
          ?date < "2011-12-12"^^xsd:date)
```

Fig. 7. A SPARQL query that returns the name of the companies that created their uri between May 1st 2008 and December 12th 2011.

In the next example we will query what are the uri and the StockPrice of the companies that have a SockPrice below 9000.

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
prefix dc:         <http://purl.org/dc/elements/1.1/>
prefix vcard:      <http://www.w3.org/2001/vcard-
rdf/3.0#>
prefix :           <http://example.org/company/>
prefix ns:
<http://sandbox.metadataregistry.org/uri/schema/fin>
SELECT ?company ?StockPrice
WHERE {
         ?company ns:StockPrice ?StockPrice.
FILTER( xsd:double(?StockPrice) < 9000.0 )
}
```

Fig. 8. A SPARQL query that returns the uri and the StockPrice of the companies that have s StockPrice below 9000.

If we want to find out what are the names of the companies that have a name like “ollo” we will address a query that looks like the one depicted in Figure 9.

```
PREFIX xsd: http://www.w3.org/2001/XMLSchema#
prefix dc:         <http://purl.org/dc/elements/1.1/>
prefix vcard:      <http://www.w3.org/2001/vcard-
rdf/3.0#>
prefix :           <http://example.org/company/>
prefix ns:
<http://sandbox.metadataregistry.org/uri/schema/fin>
SELECT ?name
WHERE {
   ?company ns:name ?name.
     FILTER( regex(?name, “ollo”, “i” ) )
}
```

Fig. 9. A SPARQL query that returns the names of the companies that have a name like “ollo”

We wanted to show how we can integrate financial data by making use of Semantic Web technologies.

We developed a vocabulary/ontology for merging data from the two sources. The vocabulary and its namespaces is available at http://sandbox.metadataregistry.org/schemaprop/list/schema_id/49.html.

We created a turtle file in order to represent data from the Excel file. Sample content is presented in Figure 10.

```
@prefix dc:
<http://purl.org/dc/elements/1.1/> .
@prefix vcard:     <http://www.w3.org/2001/vcard-
rdf/3.0#> .
@prefix ns:
<http://sandbox.metadataregistry.org/uri/schema/fin>
.

:company1
    ns:name     "China Infrastructure Invsmt" ;
    ns:tickersymbol   "CIIC" ;
         ns:industryname "Diversified Co" ;
         ns:SICCode "9913" ;
         ns:xschangeCode "NDQ" ;
         ns:SizeClass "4" ;
         ns:StockPrice "0.69" ;
         ns:TradingVolume "241738" ;
         ns:MarketCap "45.6" ;
         ns:TotalDebt "473" ;
         ns:FirmValue "519.1" ;
         ns:EnterpriseValue "517.8".

:company2
    ns:name     "AFP Imaging Corp" ;
    ns:tickersymbol   "IWKS" ;
         ns:industryname "Medical Services" ;
         ns:SICCode "8000" ;
         ns:xschangeCode "NDQ" ;
         ns:SizeClass "6" ;
         ns:StockPrice "18.5" ;
         ns:TradingVolume "364" ;
         ns:MarketCap "340.6" ;
         ns:TotalDebt "8.6" ;
         ns:FirmValue "349.2" ;
         ns:EnterpriseValue "348.4".
```

Fig. 10. Financial data sets available in Turtle format

We configured Joseki in order to query data. Services and datasets configuration are presented in Figure 11.

```
<#service2>
    rdf:type            joseki:Service ;
    rdfs:label          "SPARQL on the company
model" ;
    joseki:serviceRef   "company" ;
    joseki:dataset      <#company> ;
    joseki:processor
joseki:ProcessorSPARQL_FixedDS ;
    .
<#service3>
    rdf:type            joseki:Service ;
    rdfs:label          "sec" ;
    joseki:serviceRef   "sec" ;
    joseki:dataset      <#sec> ;
    joseki:processor
joseki:ProcessorSPARQL_FixedDS ;
    .
## Datasets

<#company>   rdf:type ja:RDFDataset ;
    rdfs:label "company" ;
    ja:defaultGraph
      [ rdfs:label "company.ttl" ;
        a ja:MemoryModel ;
        ja:content [ja:externalContent
<file:Data/company.ttl> ] ;
      ] ;
    .
<#sec>   rdf:type ja:RDFDataset ;
    rdfs:label "sec" ;
    ja:defaultGraph
      [ rdfs:label "sec.n3" ;
        a ja:MemoryModel ;
        ja:content [ja:externalContent
<file:Data/sec.n3> ] ;
      ] ;
    .
```

Fig. 11. Services and datasets configuration in Joseki

## 5. CONCLUSIONS

This paper presents with examples querying data sets by using Semantic Web technologies. Although not sufficiently treated by semantic web developers there are a lot of opportunities for those who intend to query public data sets like governments or companies or financial regulatory bodies.

We present in this paper the necessary SPARQL examples queries in order to observe what the potential for semantic search is. Our future work will refer to scalability in order that web applications may work on big data sets. We mention that we used for our examples 89523 triples stored in our triple store and that working with bigger data sets represents a problem in accessing data for Joseki server.

## ACKNOWLEDGMENTS

This work was supported by CNCSIS-UEFISCSU, project number PN II-RU code 188/2010.

## REFERENCES

**Bishr, Y.** (1998) *Overcoming the semantic and other barriers to GIS interoperability*, International Journal of Geographical Information Science, vol. 12(4), pp. 229–314

**Palmer, S.B.** (2001) *The Semantic Web: An Introduction*, http://infomesh.net/2001/swintro/

**Herman, I.** (2008) *W3C Semantic Web Activity*, W3C. http://www.w3.org/2001/sw

**Olsen, F.** (2002) *The Power of Portals*, Chronicle of Higher Education, vol 48, pp. A32-A34

**Pickett, R.A., Hamre, W.B.** (2002) *Building Portals for Higher Education*, New Directions for Institutional Research, Vol. 113, pp.37-55.

*** Securities Exchange Commission RDF Data, http://www.rdfabout.com/demo/sec/

*** W3C SemanticWeb, http://www.w3.org/2001/sw/

*** W3C Semantic Web Frequently Asked Questions, http://www.w3.org/2001/sw/SW-FAQ

*** Wikipedia, search terms “semantic web”

*** IBM Global Education Industry, Higher Education Portals: Presenting Your Institution to the World, 2000

**Correspondence to:**
**Sabina-Cristiana NECULA**
sabina.mihalache@gmail.com, Alexandru Ioan Cuza University of Iasi,