

ROLES AND USAGES OF RDF, SPARQL, ONTOLOGY TOOLS IN SEMANTIC WEB

Rupal Gupta¹, Ruchika Gupta², Ashish Bishnoi³

¹CCSIT, Teerthanker Mahaveer University, Moradabad

²Bansal Institute, Rohini

³CCSIT, Teerthanker Mahaveer University, Moradabad

Abstract

Semantic web has become as a new version of machine-understandable web including with intelligent search results in order to improve search along with data usage on the global scale. Semantic web vision is to allow intelligent description and interchange of integrated data from various distributed and heterogeneous web resources. A structure for this data on web is known as Resource Description Framework (RDF) where data is stored in the form of XML (Extended Markup Language). For effective and efficient Retrieval of the data from such large RDF, a query language SPARQL (Standard Protocol and RDF Query Language) is used.

In this paper, the usages and roles of RDF, SPARQL and Web ontology is shown with illustrations of SPARQL queries along with some optimization heuristics.

Keywords: Semantic web, Ontology, TWINKLE, RDF, SPARQL, JENA ARQ,

1. INTRODUCTION

1.1 Semantic Web

The semantic web is a potential web which provides a common framework to allow data to be shared and utilized across different web applications, enterprises and communities across world [1]. Semantic web is a broad area which consists of several components or technologies like URI, XML, RDF, RDF Schema, OWL, Unifying Logic, Proof, Trust, User interface and applications and various tools to support them. It is based on the interlinked ontology through which machine can understand the data and also it can be annotated semantically [2]. Therefore, information on web should be expressed in a meaningful way accessible to machines which may be achieved by Resource Description Framework (RDF) as a basic data format which represents resources information on the web. [3]

1.2 RDF

RDF is a basic format of representing data for Semantic Web. RDF is a framework which uses metadata and URIs to identify web resources and a graph model that describes relationship among resources. RDF describes web resources in terms of (S-P-O) subject-predicate-object triples where the subject indicates the resource, and the predicate indicates different aspects of resource and show a relationship between subject and object.[4,5] RDF graphs are also represented using these RDF triples with S-P-O. [11]

The RDF specification has its own semantics and keywords known as RDFS (Resource Description Framework Semantics), which is used to represent some special

relationships among resources such as inheritance of classes and properties [5].

RDF Tools- It Stores and manages any kind of RDF data. Some of the tools are listed below with the purpose and features.

Table 1- RDF Tools

| Name of tool | Version | Purpose | Features |
|--------------|------------|--------------------------------------|--|
| ARC | 2 | RDF Store & semantic web development | Flexible RDF storage, SPARQL query |
| Bigdata | 0.81b | RDF store | It supports RDF Schema and OWL Lite reasoning along with support to high level query |
| Sesame | 3.0-alpha1 | RDF store, Querying | It is an Open source RDF framework that support RDFS inferencing and query |
| 4 store | 0.9.2 | RDF store & query | Efficient, scalable & stable RDF database |

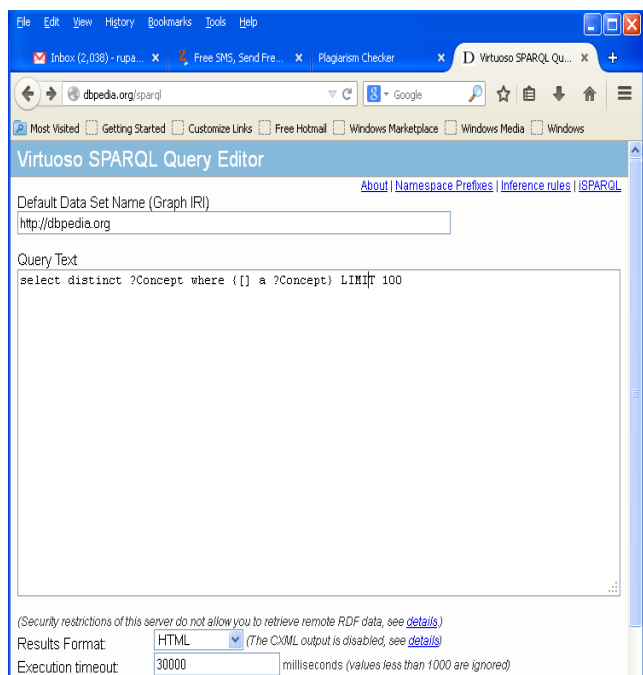
1.3 SPARQL

SPARQL stands for "Simple protocol and RDF query Language", which is a protocol and standard used to access

data from RDF, RDFS and OWL.[8] W3C SPARQL working group has defined a new version of SPARQL known as SPARQL 1.1 which supports additional features of SPARQL which was not supported earlier.[12] SPARQL query language constructs is much similar to SQL constructs and there are some tools available as open source like TWINKLE 2.0, Jena Framework with ARQ [6,7] processor on which SPARQL can be executed and tested.

For the assistance and execution of SPARQL various tools are available like Twinkle, Jena with ARQ Processor even SPARQL can also be executed using Java Program. SPARQL can also be executed online using dbpedia.

Snapshot of SPARQL query using DBpedia Tool-



1.4 Web Ontology

The Web Ontology Language (OWL) is a standard which is used for exchanging and encoding Ontology. Web Ontology provides the basic building blocks for expressing semantics in a well defined manner in semantic web. The Semantic Web gives the explicit meaning to the information so that machines can process it in intelligent manner. It will help machines to take their decisions by their own. Semantic Web also allows to give formal definitions for the standard terms created by the users.[9,10]

The Semantic Web infrastructure depends on specifications which express ontology in web compatible format. Web Ontology can also be manipulated to support Semantic web, using tool like Protégé 4.0.

There are ontology editing and development tools through which we can create, edit and manipulate the OWL, through OWL documentation along with export and import features. Some tools are shown with features and purpose.

Table 2- Web Ontology Tools

| Name of Tool | Version | Purpose | Features |
|---------------------|-------------|--|---|
| Protégé | 3.4.3 | Ontology editor | It is a Free Open Source editor and a knowledge base framework. |
| Semantic Works 2010 | v2010r2 | Used as RDF Editor along with the utility that supports Ontology creation along with Semantic Markup | Graphical RDF,RDFS,OWL editor |
| NeON toolkit | v2.3 | Ontology editor | Performs ontology engineering activities |
| SWeDE | 2.0.2 | OWL editor | Ontology editor with additional features like auto completion, syntax highlighting and error detection. |
| SWOOP | 2.3 Beta 4 | OWL editor | Tool for creating, editing, and debugging OWL ontologies. |
| Top Braid composer | 2.6.2 | Ontology editor | Powerful graphical development environment for modeling data |
| ROO | 1.0.1 | Visual RDF & OWL editor | OWL ontology construction tool |
| Ontostudio | | Ontology Engineering Tool | An ontology engineering tool for the development of semantic applications, with emphasis on rule-based modeling |
| HOZO | Ver. 5.2.14 | Ontology editor | It is an Ontology development GUI tool that support ontology development |
| COE | V5.0.3 | Ontology editor | It is an integrated software tool that construct, share and view web ontology |

1.5 Other Semantic Web Tools

There are various others tools also which helps in developing web applications, framework, micro-blogging, Machine learning in OWL. Some are listed below with the purpose and features.

Table 3- Other Tools Semantic Web

| Name of tool | Version | Purpose | Features |
|--------------|--------------------|--|---|
| CubicWeb | 3.6.11 | Semantic Web development | Build web applications by reusing components called cubes |
| Jena | Apache Jena 2.12.0 | Semantic Web development, RDF store | Framework for building Semantic Web applications |
| DL-Learner | Build 2009-05-06 | Semantic Web tool, Machine Learning tool | It Supports Supervised Machine Learning in Ontology Web Languages and also provides description logics. |
| HyperTwitter | 0.9 | Semantic MicroBlogging | Embedding triple-like statements into Twitter microblogging messages |

2. SPARQL QUERY EXECUTION USING TWINKLE AND JENA ARQ TOOL

TWINKLE and Jena ARQ are the most popular tools to execute SPARQL Query. To use both the tools it requires jdk1.5 or higher to be installed.

Twinkle Tool Twinkle2.0 can be downloaded and installed. It has a jar file in bin folder through which we can open twinkle editor to execute the SPARQL Query.

JENA with ARQ

SPARQL can be executed using sparql.bat file with the path of Query and RDF file by installing Jena framework and setting the environment variable with JENA ROOT. [13]

3. ILLUSTRATIONS OF SPARQL

In this section we represents the SPARQL queries execution in which Projection, Selection, and then one example of rewriting rule (rewrite Filter variable) is demonstrated.

Applying Projection

Query 1: The below query will find the name, course, subject and result of the faculties.

Syntax:-

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
```

```
PREFIX foaf:<http://xmlns.com/foaf/0.1/>
Select ?facultyname ?course ?subject ?result where {
?X rdf:type foaf:Person .
?x foaf:facultyname ?facultyname .
?x foaf:course ?course .
?x foaf:subject ?subject .
?x foaf:result ?result .}
order by ?facultyname
```

Output:-(On Twinkle Tool)

```
-----
| facultyname      | course | subject                                     | result |
|-----|-----|-----|-----|
| "Ashendra Saxena" | "MCA"  | "Computer Based Numerical Anaysis" | "90"   |
| "Ashish bishnoi"  | "BCA"  | "Java"                               | "88"   |
| "Deepak Sharma"  | "MCA"  | "Web Technology"                     | "95"   |
| "KANIKA"         | "BCA"  | "C Programmingk"                     | "80"   |
| "Mohan V. Gupta" | "MCA"  | "Descrete Mathematics"               | "85"   |
| "RUPAL"          | "MCA"  | "Analysis of algorithm"              | "91"   |
| "Rakesh kumar dwivedi" | "MCA" | "Digital Image Processing"           | "93"   |
|-----|-----|-----|-----|
```

Applying Selection with more Condition in Filter Clause

Query 2: The below query will find the detail of faculty with result greater than equal to 90 in "MCA" course.

Syntax:-

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
```

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
```

```
Select ?facultyname ?course ?subject ?result
```

```
Where {
?x rdf:type foaf:Person .
?x foaf:facultyname ?facultyname .
?x foaf:course ?course .
?x foaf:subject ?subject .
?x foaf:result ?result .
filter(?result >="90" && ?course="MCA")
}
order by ?facultyname
```

Output

Snapshot of query using Jena ARQ Tool

```

C:\WINDOWS\system32\cmd.exe
D:\jena-2.6.4\jena-2.6.4\bat>sparql.bat --data e:\foaf.rdf --query r3.q1

| facultyname      | course | subject                                     | result |
|-----|-----|-----|-----|
| "Deepak Sharma"  | "MCA"  | "Web Technology"                     | "95"   |
| "Rakesh kumar dwivedi" | "MCA" | "Digital Image Processing"           | "93"   |
| "RUPAL"          | "MCA"  | "Analysis of algorithm"              | "91"   |
| "Ashendra Saxena" | "MCA" | "Computer Based Numerical Anaysis"   | "90"   |
|-----|-----|-----|-----|
D:\jena-2.6.4\jena-2.6.4\bat>

```

Comparison of SPARQL Execution Tools:-

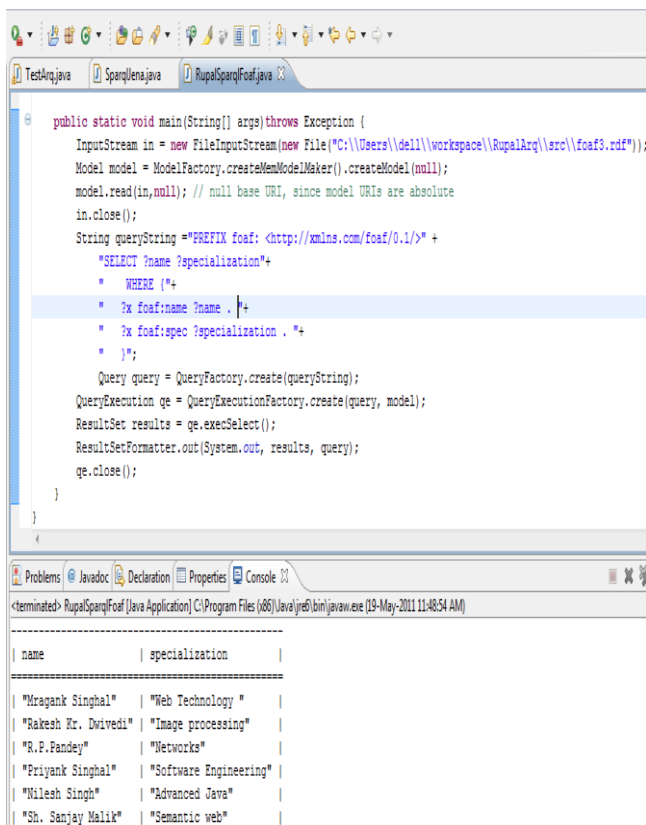
Table 4- Comparison of SPARQL execution tools

| Property/Feature | Twinkle | Jena ARQ | Java Program |
|-------------------|---------|----------|-----------------------------|
| GUI | Yes | No | Java editors are available. |
| Storage of SPARQL | Yes | No | Query |

| | | | |
|----------------------------------|------------------------|-------------------------|--|
| query through Editor | | | Code and Java Program is Written combined. |
| File format to execute the query | jar file (twinkle.jar) | batch file (sparql.bat) | Java file |
| Output storage through editor | Yes | No | Depends on Editor. |

SPARQL Queries can also be executed with the help of Java Program. Java along with JENA uses the ARQ package, which provides complete support for the SPARQL queries through various API's. To execute the program it is needed to set some environment variables and set class path for various jar files available in the ARQ package. SPARQL Query execution is shown through Java program.

SPARQL Execution using Java Program



```

public static void main(String[] args) throws Exception {
    InputStream in = new FileInputStream(new File("C:\\Users\\dell\\workspace\\RupalArq\\src\\foaf3.rdf"));
    Model model = ModelFactory.createMemModelMaker().createModel(in);
    model.read(in, null); // null base URI, since model URIs are absolute
    in.close();
    String queryString = "PREFIX foaf: <http://xmlns.com/foaf/0.1/>* +
    \"SELECT ?name ?specialization\"+
    \" WHERE {+
    \" ?x foaf:name ?name . |+
    \" ?x foaf:spec ?specialization . |+
    \" }\";
    Query query = QueryFactory.create(queryString);
    QueryExecution qe = QueryExecutionFactory.create(query, model);
    ResultSet results = qe.execSelect();
    ResultSetFormatter.out(System.out, results, query);
    qe.close();
}

```

| name | specialization |
|---------------------|------------------------|
| "Mragank Singhal" | "Web Technology " |
| "Rakesh R. Dwivedi" | "Image processing" |
| "R.P.Pandey" | "Networks" |
| "Priyank Singhal" | "Software Engineering" |
| "Nilesh Singh" | "Advanced Java" |
| "Sh. Sanjay Malik" | "Semantic web" |

4. CONCLUSION AND FUTURE WORK

In the paper we have discuss various tools used for SPARQL query in Semantic web which provides users a better understanding of SPARQL and illustrations using various tools like Twinkle , Jena ARQ and Java programs. This paper also gives a better understanding get better search results and enables people to share content on web and comprises of set of principles, various tools, techniques and collaborative work which will fulfill the need of the web. This paper will help researchers and users to summarize various tools and to choose an appropriate tool

according to their applications or requirements. This paper gives the knowledge of SPARQL query execution using Twinkle and Jena ARQ using some illustration. We can study and implement various optimization techniques/rewriting rules for SPARQL in Heterogeneous data with the consideration of distributive nature of data.

REFERENCES

- [1] Artem Chebotko, Shiyong Lu, Hasan M. Jamil, Farshad Fotouhi. "Semantics Preserving SPARQL-to-SQL Query Translation for Optional Graph Patterns", Technical Report TR-DB-052006-CLJF, 2006.
- [2] Herman, I., Semantic Web Activity, W3C, 2007 <http://www.w3.org/2001/sw/>
- [3] Groppe; Sven, Groppe; Jinghua, Kukulenz; Dirk, Linnemann; Volker, A SPARQL Engine for Streaming RDF Data, Proceedings of Third IEEE International Conference on Signal-Image Technologies and Internet-Based System, SITIS'07 Pages(s): 167-174.
- [4] Hai Dong; Hussain, F.K.; Chang, E., Application of Protégé and SPARQL in the Field of Project Knowledge Management, Proceedings of Second International Conference on Systems and Networks Communications, 2007. ICSNC 2007. Volume , Issue , 25-31 Aug. 2007 Page(s):74 – 74.
- [5] David E. Goldschmidt and Mukkai Krishnamoorthy. "Architecting a Search Engine for the Semantic Web ," Rensselaer Polytechnic Institute, Troy, New York, USA.
- [6] ARQ- A SPARQL Processor for Jena . <http://jena.sourceforge.net/ARQ/>.
- [7] Jena sparql tutorial: <http://jena.sourceforge.net/ARQ/Tutorial/query1.html>
- [8] Jerome Euzenat , "Processing Ontology Alignment with SPARQL ." 2006.
- [9] Evren Sirin, Blazej Bulka, and Michael Smith, "Terp : Syntax for OWL-friendly SPARQL queries. Washington , USA.
- [10] Hai Dong , Farookh khadeer Hussain , Elizabeth Chang, "Application of Protégé and SPARQL in the field of Project Knowledge Management", IEEE computer society, 2007.
- [11] Michael Grobe, Indiana University. "RDF, Jena, SPARQL and the Semantic Web", Indianapolis, Indiana, USA, SIGUCCS, 2009.
- [12] E. P. Steve Harris, Andy Seaborne. SPARQL 1.1 Query Language, June 2010.
- [13] R. Gupta, Sanjay Malik, "SPARQL Semantics and Execution Analysis in Semantic web using various Tools", CSNT, IEEE computer society, 2011.