

Time and Accuracy Analysis of Relational vs. RDF Databases

*Thesis submitted in partial fulfillment of the requirements for the award of
degree of*

Master of Engineering
in
Software Engineering

Submitted By
Amita Arora
(801031004)

Under the supervision of

Dr. Seema Bawa
Professor (CSED),
Dean (Student Affairs)

Dr. Shalini Batra
Assistant Professor (CSED)



COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004
June 2012


CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, "*Time and Accuracy Analysis of Relational vs. RDF Database*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Software Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Seema Bawa, Professor, CSED* and *Dr. Shalini Batra, Assistant Professor, CSED* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.



Amita Arora


This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.


Dr. Seema Bawa
Professor (CSED),
Dean (Student Affairs)


Dr. Shalini Batra
Assistant Professor
(CSED)

Countersigned by


(Dr. Maninder Singh)
Head
Computer Science and Engineering Department
Thapar University
Patiala


(Dr. S.K. Mohapatra)
Dean (Academic Affairs)
Thapar University
Patiala

ACKNOWLEDGEMENT

No volume of words is enough to express my gratitude towards my guide **Dr. Seema Bawa** and **Dr. Shalini Batra**, Department of Computer Science & Engineering, Thapar University, Patiala, who has been very concerned and has aided for all the materials essential for the preparation of this thesis report. She has helped me to explore this vast topic in an organized manner and provided me with all the ideas on how to work towards a research-oriented venture.

I would also like to thank the staff members and my colleagues who were always there at the need of the hour and provided with all the help and facilities, which I required, for the completion of my thesis work.

Most importantly, I would like to thank my parents and the almighty for showing me the right direction out of the blue, to help me stay calm in the oddest of the times and keep moving even at times when there was no hope.

Amita Arora
(801031004)

As the scale and impact of the Internet has been growing, search engines have assumed an important role in the Web's infrastructure. Huge amount of information is available on Web but accurate and relevant results cannot be fetched from the search engines because these search engines retrieve data syntactically. To fetch the data as per the user's requirements, semantic search engines are required. They are expected to be the next generation of web as Semantic Web provides a well defined meaning to the available data. In semantic web, data is represented in the languages like RDF and OWL and linked to the commonly accepted ontologies.

To provide contextual information to the user, RDF databases were introduced. Semantic processing for data retrieval is accomplished by these RDF databases as they store the meta information about the data in the form of subject, predicate and object. The annotations increase the efficiency of semantic search by associating the metadata to the resources. Although data can be stored and retrieved efficiently in these types of databases they require some methodologies to query the repository and the query languages like SPARQL, SeRQL are used.

In this thesis, two database approaches RDF database and relational database have been compared using two evaluation parameters viz. accuracy of search results and time required to access the databases. It has been experimentally evaluated that search results are faster and more accurate as compared to Relational database.

Table Of Contents

Certificate	i
Acknowledgement	ii
Abstract	iii
Table of Contents	iv
List of figures	vi
List of Tables	vii
Chapter 1: Introduction	1
1.1 Relational Database	1
1.2 Semantic Web	2
1.2.1 Structure of Semantic Web	2
1.2.2 Hypertext Mark up Language	3
1.2.3 Extensible Mark up Language	4
1.3 Resource Description Framework (RDF)	5
1.3.1 Basic RDF Model	5
1.3.2 RDF Schema	9
1.4 RDF Database	10
1.5 Relational Database vs. RDF Database	10
1.6 Structure of Thesis	11
Chapter 2: Litreature Review	12
2.1 Database Model	12
2.1.1 Relational Database	13

2.2 RDF Database	13
2.2.1 Jena	15
2.2.2 Sesame	16
2.2.3 Mulgara	17
2.2.4 Virtuoso	18
Chapter 3: Problem Stataement	19
3.1 Research Gap	19
3.2 Problem Formulation	19
3.3 Objectives	19
3.4 Methodology	20
Chapter 4: Implementation and Results	21
4.1 RDF Repository Creation	22
4.2 Installation of Sesame	27
4.3 Running query on Mysql for “cross” database	32
4.4 Comparison of query Retrieval Time	33
4.5 Query for “Montblanc” RDF Database	34
4.6 Query on Mysql for “Montblanc”	35
Chapter 5: Conclusion and Future Scope	38
5.1 Conclusion	38
5.2 Future Scope	38
References	39
List of Publications	42
Appendix-A	43

List Of Figures

Figure 1.1	Berners-Lee's Semantic Web 'Birthday Cake'	2
Figure 1.2	Simple node and arc diagram	7
Figure 2.1	Sesame Architecture	17
Figure 4.1	RDF document for "cross" repository using Notepad	22
Figure 4.2	Validation of RDF using W3C RDF Validation Service	23
Figure 4.3	RDF document for "Montblanc" repository using Notepad	23
Figure 4.4	Validation of RDF using W3C RDF Validation Service	24
Figure 4.5	RDF document for "hangover" repository using Notepad	24
Figure 4.6	Validation of RDF using W3C RDF Validation Service	25
Figure 4.7	RDF document for "Twilight" repository using Notepad	25
Figure 4.8	Validation of RDF using W3C RDF Validation Service	26
Figure 4.9	RDF document for "Eclipse" repository using Notepad	26
Figure 4.10	Validation of RDF using W3C RDF Validation Service	27
Figure 4.11	Tomcat server start up window	28
Figure 4.12	Create new repository in Open RDF workbench	29
Figure 4.13	List of created RDF Repositories	29
Figure 4.14	Querying repository using Sparql	30
Figure 4.15	Querying on repository "cross"	30
Figure 4.16	Query result for showing the all types of "cross"	31
Figure 4.17	Showing all the value of "cross"	31
Figure 4.18	Mysql Database of "cross"	32
Figure 4.19	Query Result for "cross"	32
Figure 4.20	Query retrieval time in RDF database	33
Figure 4.21	Query retrieval time in relational database	33
Figure 4.22	Running query for "Montblanc"	34
Figure 4.23	Query Result for "Montblanc"	34
Figure 4.24	Showing all the value of "Montblanc"	35

List Of Tables

Table 2.1	RDF Databases	16
Table 4.1	Detail of RDF Repositories	21
Table 4.2	Comparison Analysis of Query Results in RDF and RDBMS	36

Chapter 1

Introduction

As the scale and impact of the Internet has been growing, search engines have assumed an important role in the Web's infrastructure. Internet has emerged as a huge repository of unstructured data. This has resulted in making the data search and extraction a difficult task. Information retrieval provides user content. The retrieval of required information in minimum possible time and with maximum efficiency and accuracy is a major challenge. With the rapid development of network technique, the size and complexity of information resource is now growing day by day. So information resources require an intelligent management system which can solve the key technique about information representation, storage and semantic annotations. Many information resources are available like relational database Mysql, Oracle, *etc.* but they cannot retrieve data semantically. Relational database will not give accurate results as user wants and even take more time.

Internet has two application layer protocols, Hypertext Transfer Protocol (HTTP) and Hypertext Markup Language (HTML). Hyper Text Transfer Protocol (HTTP) and Hyper Text Markup Language (HTML) provide a reliable, communication channel. HTML is essentially a presentation layer language for formatting web pages. Since HTML does not supports semantics, i.e., it does not provide a the required description of the web page, the World Wide Web Consortium introduced standardized protocols such as Extensible Markup Language (XML), Resource Description Language (RDF) and Web Ontology Language (OWL) to address the semantics issue.

1.1 Relational databases

A relational database is a collection of data items organized as a set of formally described tables from which data can be accessed easily. A relation is defined as a set of tuples that have the same attributes and a tuple is usually represented as an object and information about that object. A relation is usually described as a table and all the data referenced by an attributes are in the same domain and conform to the same constraints.

The relational model specifies that the tuples of a relation have no specific order and that the tuples, in turn, impose no order on the attributes. Applications access data by specifying queries, which use operations such as *select* to identify tuples, *project* to identify attributes, and *join* to combine relations. Relations can be modified using the *insert*, *delete*, and *update* operators. New tuples can supply explicit values or be derived from a query. Similarly, queries identify tuples for updating or deleting [1].

1.2 Semantic Web

Semantic Web is an extension of the existing Web in which information is given a well-defined meaning. The ultimate goal of the Semantic Web is to transform the Web into a medium through which data can be shared, understood and processed by automated tools.

Semantic web allow intelligent search agents to process and integrate data from heterogeneous resources.

Semantic web is a universal medium for information exchange by giving meaning (semantics) to the content of documents on the Web. The document should contain meta information and vocabularies for the meta information, like Resource Description Framework Schema (RDFS) would be required for the realization of this idea.

1.2.1 Structure of Semantic Web

The Semantic Web is implemented in the layers of Web technologies and standards. The layers are presented in Figure 1.1 and described as follows:

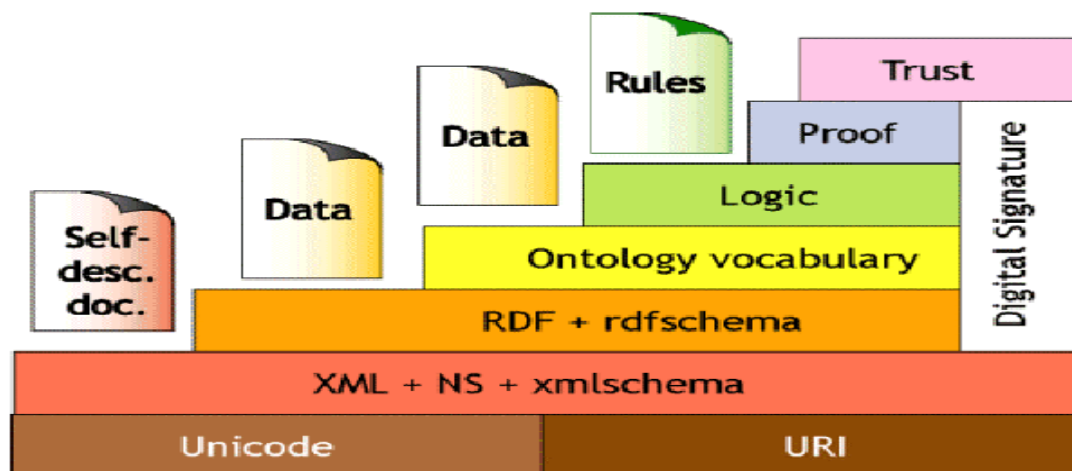


Figure 1.1: Berners-Lee's Semantic Web 'Birthday Cake' [2]

- The two technologies UNICODE and URI are taken from the World Wide Web. URI provides global identifiers and UNICODE is the universal standard encoding. URI makes sure the character sets are used and provide the means for identifying the objects in the semantic web.
- The **XML** layer with namespace and schema definitions ensures that the Semantic Web definitions are integrated with the other XML based standards. XML provides syntax for structured documents, but imposes no semantic constraints on the meaning of these documents. XML Schema is a language for restricting the structure of XML documents.
- RDF is a simple metadata representation framework, using URIs to identify Web-based resources and a graph model for describing relationships between resources. RDF Schema a simple type modelling language for describing classes of resources and properties between them. It provides a simple reasoning framework for inferring types of resources.
- The term "ontology" can be defined as an explicit specification of conceptualization. Ontology captures the structure of the domain. This includes the model of the domain with possible restrictions. The conceptualization describes knowledge about the domain i.e. the ontology layer describe the knowledge of domain supports the evolution of vocabularies as it can define relations between the different concepts.
- A digital signature is an electronic signature that can be used to authenticate the identity of the sender of a message or the signer of a document.
- The top layers Logic, Proof and Trust are currently being researched and simple application demonstrations are being constructed. The Logic layer enables the writing of rules while the Proof layer executes the rules and evaluates together with the Trust layer mechanism for applications whether to trust the given proof or not.

1.2.2 Hypertext Markup Language

HTML is used to create document on the World Wide Web. It is a collection of certain keywords called 'Tags' that are helpful in writing the document to be displayed on

Internet. It is a platform independent language and provides tags which are already defined to create documents. Example:

```
<HTML>
  <HEAD>
    <TITLE>
      My first Page
    </TITLE>
  </HEAD>
  <BODY>
    WELCOME TO MY FIRST WEB PAGE
  </BODY>
</HTML>
```

The main focus of HTML is on how to display data and tags which are used in HTML are predefined. The weakness of HTML is that it has predefined set of tags thus it has no capability to create user defined tags [3].

1.2.3 Extensible Markup Language

XML is a system independent way of representing data. In XML documents are written between the tags which are extensible according to the user requirement. XML relates what is to be displayed unlike HTML how data is to be displayed. XML does not restrict the user to a predefined set of tags instead it enables user to create arbitrary tags. XML is extensible as required tags can be created for particular type of documents and any type of data can be encoded in XML tags. The only constraint is that a well formed and valid XML document conforms to a schema i.e. it requires the XML file adhere to a predefined grammar. The XML schema provides a syntactic description but doesn't provide semantic description of entities. XML file will yield both syntactic and semantic information if web users use the same terms to describe the entities. Schemas in XML are always restricted as it should have ending tag same as starting tag. RDF overcomes the semantics limitations to a great extent and can represent semantics more efficiently than XML [4]. Example of XML document:

```
<?xml version="1.0"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note> [5]
```

1.3 Resource Description Framework (RDF)

RDF is a graphical language used for representing information about resources on the web. It is a basic ontology language. Resources are described in terms of properties and property values using RDF statements. Statements are represented as triples, consisting of a subject, predicate and object.

The World Wide Web was originally built for human consumption, everything on it was machine-readable, but was not machine-understandable. Web contains large volume of data and because of this it is very hard to manage it manually. The solution is to use metadata means "data about data" to describe the data contained on the Web. The distinction between "data" and "metadata" is not an absolute one; it is a distinction created by a particular application, and many times the same resource will be interpreted in both ways simultaneously.

RDF and its schema language Resource Description Framework Schema (RDFS) were proposed by the World Wide Web Consortium to overcome the semantic limitations of XML. Resource Description Framework (RDF) provides interoperability between applications that exchange machine-understandable information on the Web. RDF emphasizes facilities to enable automated processing of Web resources.

RDF can be used in a variety of application areas for example:

- In resource discovery to provide better search engine capabilities.
- In cataloguing for describing the content and content relationships available at a particular Website, page, or digital library.
- By intelligent software agents to facilitate knowledge sharing and exchange.
- In content rating, in describing collections of pages that represent a single Logical "document".

1.3.1 Basic RDF Model

RDF is model and XML syntax for representing information in a way that allows machines to understand the intended meaning. RDF represents information about resources (metadata) on the Web. A resource in RDF is mapped to a Uniform Resource Identifier (URI) and is described as properties. The purpose of RDF is to make statements about resources, which can be viewed as labeled edge (property) between two nodes

(object and property value). The foundation of RDF is a model for representing named properties and property values. RDF properties may be thought of as attributes of resources and in this sense it corresponds to traditional attribute-value pairs. RDF model resemble to an entity relationship diagram. In object-oriented design terminology, resources correspond to objects and properties correspond to instance variables. The RDF data model is a syntax-neutral way of representing RDF expressions. Two RDF expressions are equivalent if and only if their data model representations are the same. The basic data model consists of three object types:

- **Resources:** All things being described by RDF expressions are called resources. A resource may be an entire Web page; such as the HTML document "<http://www.w3.org/Overview.html>" for example. A resource may be a part of a Web page; e.g. a specific HTML or XML element within the document source. A resource may also be a whole collection of pages; e.g. an entire Web site. A resource may also be an object that is not directly accessible via the Web; e.g. a printed book. Resources are always named by URIs plus optional anchor ids. Anything can have a URI. The extensibility of URIs allows the introduction of identifiers for any entity imaginable.
- **Property:** A property is a specific aspect, characteristic, attribute, or relation used to describe a resource. Each property has a specific meaning, defines its permitted values, the types of resources it can describe, and its relationship with other properties. This object type does not address how the characteristics of properties are expressed.
- **Statement:** A specific resource together with a named property plus the value of that property for that resource is an RDF statement. These three individual parts of a statement are called, the subject, the predicate, and the object respectively. It's built on the concept of a statement, a triple of the form {predicate, subject, and object}. The interpretation of a triple is that <subject> has a property <predicate> whose value is <object>. In RDF a <subject> is always a resource named by a URI with an optional anchor id, <predicate> is a property of the

resource, and the <object> is the value of the property for the resource. The object of a statement (i.e., the property value) can be another resource or it can be a literal; i.e., a resource (specified by a URI) or a simple string or other primitive data type defined by XML. In RDF terms, a literal may have content that is XML mark-up but is not further evaluated by the RDF Processor.

Examples

Resources are identified by a resource identifier. A resource identifier is a URI plus an optional anchor id. Consider as a simple example the sentence: Jan Egil Refsnes is the author of resource: <http://www.w3schools.com/rdf/> Subject (Resource): <http://www.w3schools.com/rdf/>. Predicate (property): author Object (literal): "Jan Egil Refsnes" [6] the diagram of this RDF statement pictorially using directed labeled graphs (also called "nodes and arcs diagrams") is shown in Figure 1.2. In this diagram, the nodes (drawn as ovals) represent resources and arcs represent named properties.

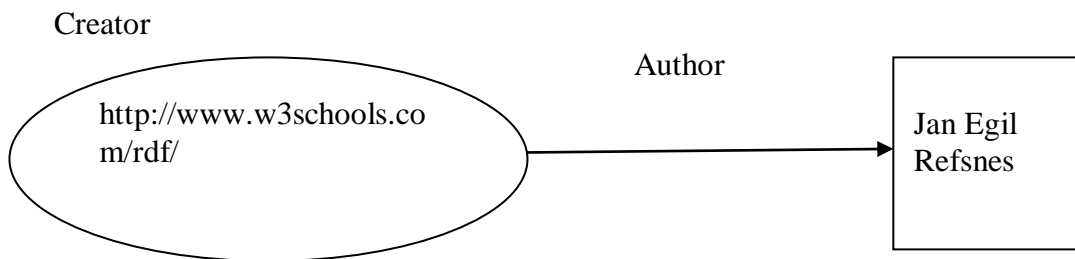


Figure 1.2: Simple node and arc diagram

Nodes that represent string literals will be drawn as rectangles. In prose, such a Sentence would be:

The individual whose name is Jan Egil Refsnes is the author of <http://www.w3schools.com/rdf/>

RDF General Format

```
<? xml version="1.0"?>
<class rdf:ID="Resource"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns="uri">
<property>value</property>
<property>value</property>
...
</Class>
```

Element description

- **rdf:ID:** RDF provides an ID attribute for identifying the resources being described. The value of rdf:ID is a "relative URI". The "complete URI" is obtained by concatenating the URL of the XML document with "#" and then the value of rdf:ID.
- **Resource:** Identifies the resource being described.
- **xmlns:rdf:** The ID attribute is in the RDF namespace.
- **Class:** Identifies the type (class) of the resource being described.
- **Property:** These are properties, or attributes, of the type (class).
- **Value:** Values of the properties.
- **rdf:about :** Instead of identifying a resource with a relative URI (which then requires a base URI to be attached before relative URI), we can give the complete identity of a resource. However, we use rdf:about, rather than rdf:ID.

Rdf:ID vs. Rdf:about

rdf: about: Instead of identifying a resource with a relative URI (which then require a base URI to be attached before relative URI), we can give the complete identity of a resource. However, we use rdf: about, rather than rdf:ID. rdf:ID versus rdf: about :

- To introduce a resource, and provide an initial set of information about resource use rdf:ID.

- To extend the information about a resource use rdf: about

Data Model-XML Syntax

- Well-formed XML
- Validation through RDF schemas

Here is an example of an XML document that specifies data about Jan Egil Refsnes:

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:si="http://www.w3schools.com/rdf">
<rdf:Description rdf:about="http://www.w3schools.com">
  <si:title>W3Schools.com</si:title>
  <si:author>Jan Egil Refsnes</si:author>
</rdf:Description>
</rdf:RDF> [6].
```

The three different types of container objects defined by the RDF specification are Bag, Sequence and Alternative.

- **Bag** It is an unordered list of resources or literals. *Bags* are used to declare that a property has multiple values and when there is no significance to the order in which the values are given.
- **Sequence** It is an ordered list of resources or literals. Sequence is used to declare that a property has multiple values and when the order of the values is significant. Sequence might be used, for example, to preserve an alphabetical ordering of values. Duplicate values are permitted.
- **Alternative** It is a list of resources or literals that represent alternatives for the (single) value of a property. Alternative might be used to provide alternative language translations for the title of a work, or to provide a list of Internet mirror sites at which a resource might be found. An application using a property whose value is an alternative collection is aware that it can choose any one of the items in the list as appropriate.

1.3.2 RDF Schema

RDF asserts statements about resources in terms of properties. The description of these properties is provided by the RDF schema specification. RDF schemas add semantics to RDF. They provide a type of system for resources. They also define relationships between resources and can be encoded using RDF resources, such as Class, SubclassOf, Property, etc. which are defined in the namespace of RDFS. RDF document provide a mechanism to describe metadata about resources. Every entity in RDF is a resource and a class represents a group of resources. Every resource in RDF is an instance of at least one class. The core set of classes and properties are used to describe metadata about web resources by RDF schemas. There exists a subset of RDF resources using that all other RDF schema constructs can be defined. The core classes in RDF schema are rdfs:class, rdfs:resource, rdfs:property and rdfs:constraint resource.

- i. rdfs:class it represents the set of entities used to describe concepts in a schema.
- ii. rdfs:resource represents the set of all resources.
- iii. rdfs:property represents the set of all RDF properties.

- iv. `rdfs:constraint`
- v. `resource` represents the set of all constraints applicable to RDF resources.

Core properties include properties of the class `rdfs:property` namely, `rdf:type`, `rdfs:subclassof`, `rdfs:range` and `rdfs:domain`. The RDF model assigns a dual role to these core classes and properties, which is used to define other schema constructs and can be used in the specification of an RDF schema itself that causes self-referentiality in the RDF model.

Some important feature of RDFS:

- RDF describes resources with classes, properties, and values. In addition, RDF also needs a way to define application-specific classes and properties.
- Application-specific classes and properties must be defined using extensions to RDF. One such extension is RDF Schema RDFS.
- RDF Schema does not provide actual application-specific classes and properties. Instead RDF Schema provides the framework to describe application specific classes and properties.
- Classes in RDF Schema are much like classes in object oriented programming languages. This allows resources to be defined as instances of classes, and subclasses of classes

1.4 RDF Database

RDF Database is a collection of data in the form of triples (subject, object, predicate) and this data can be written in any RDF language like RDF/XML in notepad or RDF editor. RDF Database actually stores metadata (information about the data) in the form of triples and the purpose of triple store which stores the RDF data (for .e.g. Jena, Sesame, Virtuoso, *etc.*) are databases for the storage and retrieval of RDF metadata.

1.5 Relational Database vs. RDF Database

In general, each RDBS is designed to store a particular data set and is described by the terms commonly used and understood by its local users. Since the individual

RDBSs are designed and built independently, many challenges arise when two or more RDBSs must interoperate with each other. One of the main challenges is semantic conflicts, which include:

- (i) Naming conflicts, such as homonyms and synonym
- (ii) Generalization and specialization conflicts
- (iii) Atomic conflicts, composite conflicts, *etc.*

1.6 Structure of the Thesis

The rest of the thesis is organized in the following order:

Chapter 2: “Literature Review” - Provides a review of all the work done in the area of Relational database and RDF Data.

Chapter 3: “Problem Statement” - Gives the problem statement and methodology used to solve it.

Chapter 4: “Implementation and Results”- Explains the experiments performed and evaluates the results achieved.

Chapter 5: “Conclusion and Future Scope” - Presents the conclusion of this thesis and suggestions for future work.

Chapter2

Literature Review

In 1990's, the World Wide Web (WWW) has rapidly evolved into a huge mine of global information and it is growing in size every day. Huge amount of data is available on web, and to retrieve the data from web, search engines are used by web users. These search engines store data in information resources. Web user send query to these search engines and retrieve require data but this data is not as accurate as user wants. There are some data resources available to store data.

2.1 Database Model

A database is an integrated collection of data records, files, and other objects. The data is stored in the form of table in which every row is called Tuple or Record that can be retrieved through the query languages. DBMSs may use a variety of database models, such as the relational model or object model, to conveniently describe and support applications. It typically supports query languages to retrieve the data from data base.

E.F Codd in early 1970's offered well motivated and appropriately restricted computational paradigm for Database Management. Most of the system's effort in 1970's was devoted to efficient implementation of this approach which is a largely successful endeavor which has made a Relational Database Management System [7]. Codd highlighted the concept of level of abstraction by introducing a clean separation between physical and logical levels. Due to its simplicity of modeling it gained wide popularity among developers and business applications. It is based on the simple mathematical notion of relation, which together with its associated algebra and logic, made the relational model a primary model for database research. In particular, its standard query and transformation language, SQL, became a paradigmatic language for querying [8]. IBM started working on a prototype system loosely based on Codd's concepts in the early 1970s.

2.1.2 Relational Database

A relational database is a collection of data items organized as a set of formally-described tables from which data can be accessed or reassembled in many different ways without having to reorganize the database tables. Relational database allows organizations to conveniently develop databases for various applications. The relational model is directed to simple record-type data with a structure known in advance (airline reservations, accounting, *etc.*). Extensibility is a difficult task because the schema is fixed and integration of different schemas is not easy. The query language does not support paths, neighborhoods and queries that address connectivity (an exception is transitivity) [8].

Relational model is not enough for data intensive domains such as CAD/CAM software, computer graphics, and information retrieval through this model is not accurate and efficient, therefore semantic model is introduced.

According to Navathe the origin of semantic model is necessary for more expressiveness and incorporation of a richer set of semantics into the database from the user point of view. They allow database designers to represent objects and their relations by using high-level abstraction concepts such as aggregation, classification and instantiation, sub- and super-classing, attribute inheritance and hierarchies [9]. A well-known example is the entity-relationship model [10]. It has become a basis for the early stages of database design, but due to lack of preciseness cannot replace models like relational or object oriented.

So there was a need of the kind of databases which stores the data in the form of meta data and where both data manipulation and representation are graph-based. One of the proposed solutions by World Wide Web Consortium (W3C) was RDF databases.

2.2 RDF Databases

The *Hypertext Transport Protocol* (HTTP) is used to request documents which are received by programs and a Web server uses a URL provided in the request to determine which file to deliver. HTML suffered from two problems. Firstly, HTML is insufficient simply add additional tags to documents, resulting in a number of non-standard variants. Secondly, because HTML was mostly designed for presentation to humans, it is difficult for machines to extract content and perform automated processing on the documents.

To solve these problems, the World Wide Web Consortium (W3C) developed the *Extensible Markup Language* (XML) [11]. XML was designed to transport and store data, with focus on what data is whereas, HTML was designed to display data, with focus on how data looks. It can be said that HTML is about displaying information XML is about carrying information. But, XML could not provide machine understandable construct as the XML schema provides a syntactic description of data and it does not provide semantics to the data. *Resource Description Framework* (RDF) has been developed by the W3C to encode metadata [12]. RDF can be used to describe any kind of resource. These resources can be accessed and identified through a URI. This allows for the distribution of a resource's description over different nodes. RDF can be represented in many ways, e.g., with a graph, with triples, or in an XML mark-up called RDF/XML. The *RDF Vocabulary Description Language, RDF Schema* (RDFS), adds semantics to an RDF data model by defining its vocabulary. To enhance the semantic interpretability of data, semantic *annotation* can be added [13]. The semantic information can be annotated to the documents using the RDF graph data model, RDF query languages are used to retrieve the data.

RDF structure of any expression is a collection of triples, each consisting of a subject, a predicate and an object. A set of these triples is called an RDF graph.

According to Berners-Lee, "The Semantic Web extends the World Wide Web through the use of standards, mark up languages and related processing tools".

The Semantic Web builds a common framework that allows data to be shared and reused among applications, enterprises and community boundaries.

According to Chen *et al.* data in big organizations is normally stored in relational databases or other appropriately formatted documents [14]. Relational database cannot handle unstructured data so, organizations emphasizing on those techniques and applications which handles unstructured document. One of the methods to handle unstructured data is usage of semantic web. The vision of Semantic Web is:

- (i) To interconnect distributed located legacy databases using richer semantics.
- (ii) To provide ontology-based query, search and navigation as one huge distributed database.

- (iii) To add additional deductive capabilities to increase the usability and reusability of data [14].

Semantic Web contain documents enriched by annotations that provide metadata about the documents as well as machine interpretable statements capturing some of the meaning of the documents' content. Retrieval of information of these documents offers new challenges and new opportunities. Many frameworks are presented for integrating search some of them are like Sesame, Jena, Virtuoso, etc. Sesame which is open source creates one specialized table per class. Tightly coupling the table layout to schema structure can facilitate inference by allowing the systems to exploit the explicit schema relationships [5]. Different types of RDF data base stores are as follows:

2.2.1 Jena

Jena is a Java framework for building semantic web applications. Jena implements APIs for dealing with Semantic Web building blocks such as RDF and OWL.

Jena is a Semantic Web programmers' toolkit, which is an open-source Project, implemented in Java, and offers a simple abstraction of the RDF graph as its central internal interface. It implements the interpretation of the RDF specifications and was developed to satisfy two goals:

- To provide an API that was easier for the programmer to use than alternative implementations.
- To be conformant to the RDF specifications.

Table 2.1 RDF databases (abbreviations: o. = ongoing, disc.= discontinued, e.d.s. = early developing stage, u. = unknown)

Name	State	Programming language	Supported Query Language	Supported Storage	Part of Evaluation	Licence
Jena	o.	Java	SPARQL,RDQL	In memory, relative disk storage, relational backends	Yes	Open Source
Sesame	o.	Java	SPARQL,SeRQL	In memory, relative disk storage, relational backends	Yes	BSD style Licence
Virtuoso	o.	Java	SPARQL	Relational database	No	BSD style Licence
Mulgara	o.	Java	SPARQL,TQL and Jena Binding	Integrated Database	No	Open Software Licence v 3.0

2.2.2 Sesame

Sesame is an open-source framework for querying and analyzing RDF data. It was created by the Dutch software company Aduna. It contains a triple store. Sesame supports two query languages SeRQL and SPARQL. Sesame is written in Java and has BSD style license. Sesame is a framework for storage of data and inference of RDF and RDF schema. Sesame is a database server for Remote access to the repositories of RDF data.

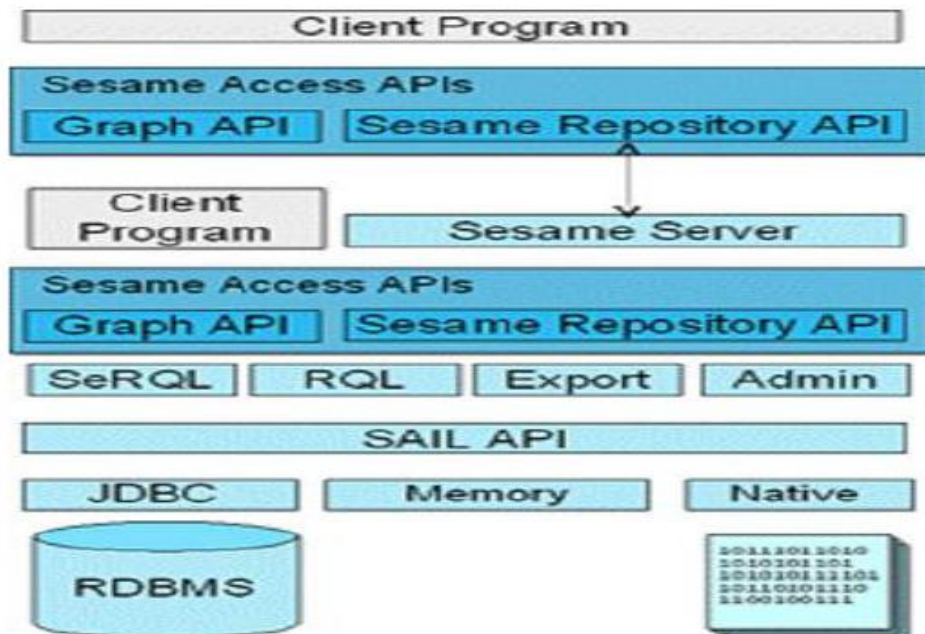


Figure 2.1: Sesame Architecture [15]

In Sesame architecture (Figure 2.1) the Storage and Inference Layer, or SAILAPI, is an internal Sesame API that abstracts from the storage format to provide functionality such as caching or concurrent access handling. Each Sesame repository has its own SAIL object to represent it and on top of the SAIL, there are Sesame's functional modules, such as the SeRQL, RQL and RDQL query engines, the admin module, and RDF export. Access to these functional modules is available through Sesame's Access APIs, consisting of two separate parts: the Repository API and the Graph API. The Repository API provides high-level access to Sesame repositories, such as querying, storing of RDF files, extracting RDF, etc. The Graph API provides more fine-grained support for RDF manipulation, such as adding and removing individual statements, and creation of small RDF models directly from code. The Access APIs provide direct access to Sesame's functional modules, either to a client program or to the next component of Sesame's architecture, the Sesame server.

2.2.3 Mulgara

Mulgara is a native RDF triple store written in Java. It provides a Connection API that can be used to connect to the Mulgara store. Being a native triple store it has a 'load'

script which can be used to load RDF data to the triple store. In addition to supporting SPARQL queries through the connection API, these can be performed through the TQL shell. The TQL shell is a command line interface that allows queries on models present in the store [16].

2.2.4 Virtuoso

Virtuoso is a native triple store available in both open source and commercial licenses. It provides command line loaders, a connection API, and support for SPARQL and web server to perform SPARQL queries and uploading of data over HTTP. Virtuoso provides bridges for it to be used with Jena and Sesame [16].

Al-Ghamdi *et al.* Information sharing, exchanging and retrieving from heterogeneous data sources not only needs complete data accessibility, but also needs solving data heterogeneity between these data sources. To solve this rising problems of heterogeneity, a lot of recent work has been done towards solving this issue. To develop distributed heterogeneous data sources some software system has been developed which are based on ontology to semantically integrate heterogeneous data such as XML and RDF to solve some conflicts that occur in these sources [17].

After going through various proposals presented for RDF database, it has been realized that accuracy of the retrieved results can be enhanced by adding annotation to the words available on WWW. RDF, which serves as a basic language for Semantic Web can be used as a means to describe all the resources based on some predefined rules. This makes the searching process more accurate because search engine directly approaches the RDF documents for getting the reliable information related to search and returns more relevant results.

Chapter 3

Problem Statement

While doing literature survey of RDF database, following research gaps are observed.

Research Gaps

- i. Relational Database cannot retrieve data semantically.
- ii. Relational database takes more time while run the query.
- iii. Key word search technique which is applied on relational database gives the data which has same key word, but no accurate result.
- iv. Relational Database is not machine understandable language.
- v. Relational Data Base does not support unstructured data.
- vi. In relational database, the data is in tables as it is but cannot give annotations (tell about the data).
- vii. Relational database does not store the data in the form of graphs because of this it does not give accurate results.

Problem Formulation

During the literature review, it was analyzed that Semantic Search aims to make the process of data retrieval through WWW more efficient than Syntactic Search and it focuses on satisfying the needs of user by making the search operations based on annotations rather than by using the traditional Relational Database techniques only. This thesis observes some of the points during research gap analysis which includes study of relational database and RDF database and their comparison regarding accuracy and time.

Objectives

1. To retrieve the data semantically so that it can understand user context and gives query for that data.
2. To measure accuracy of data retrieval between Relational and RDF database.
3. To measure the time of retrieval between Relational database and RDF database.

Methodology

- i) a) Study of the semantic Web technologies like RDF, OWL etc.
- ii) a) Installation of Sesame tool and querying the RDF repository using the SPARQL and SeRQL.
b) Create database on relational database and querying database using Mysql.
c) Compare relational database and RDF database query retrieval time.
- iii) a) study RDF databases and stores data in RDF database because RDF database stores data in the form of graph which is made up of subject, object and predicate

Chapter4

Implementation And Results

In order to measure accuracy and time of data retrieval from RDF and relational databases, RDF Repository and Relational database is created. Notepad is used to make RDF document and Sesame triple store is used to store and query RDF metadata. RDF database are queried using SPARQL query language while Mysql query language is used to retrieve the data from the relational database and then comparison is performed between them.

RDF repositories are created in Sesame RDF framework. Five repositories are created the description is as follows:

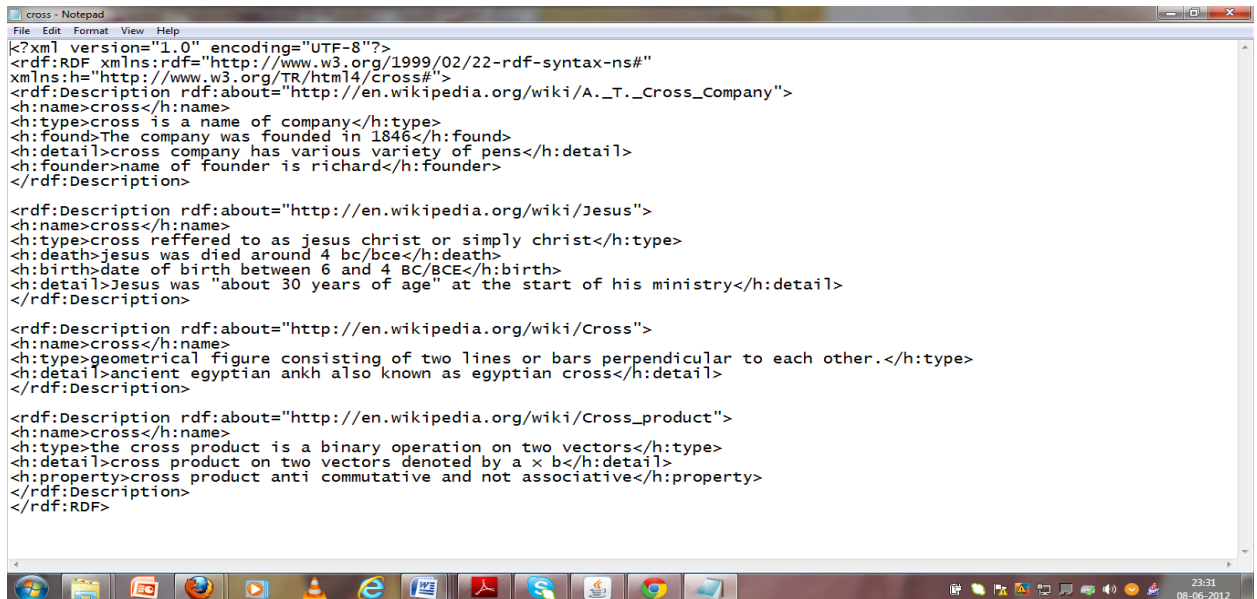
Table 4.1: Details of RDF Repositories

S.No	Name of Repository	ID	Title	Location	Sesame Server
1	Hangover	hh	hang	http://localhost:8080/openrdf-sesame/repositories/hh	http://localhost:8080/openrdf-sesame
2	Montblanc	22	montblanc	http://localhost:8080/openrdf-sesame/repositories/22	http://localhost:8080/openrdf-sesame
3	Twilight	tt	twi	http://localhost:8080/openrdf-sesame/repositories/tt	http://localhost:8080/openrdf-sesame
4	cross	cc	cross	http://localhost:8080/openrdf-sesame/repositories/cc	http://localhost:8080/openrdf-sesame
5	Eclipse	ce	ece	http://localhost:8080/openrdf-sesame/repositories/ce	http://localhost:8080/openrdf-sesame

In RDBMS Eclipse database is created and then further “Hangover”, “Montblanc”, “Twilight”, “cross”, “Eclipse1” tables are created in Eclipse database. Mysql query language is used to retrieve the data.

4.1 RDF Repository Creation

RDF editor or notepad has been used to make the RDF document for “Cross”, “Twilight”, “Montblanc”, “Hangover” and “Eclipse” repository. Self defined tags are used to annotate the words and RDF files are saved with the extension (.rdf). Figure 4.1, 4.3, 4.5, 4.7, 4.9 shows RDF document created using Notepad. The first line of RDF document is XML declaration. The XML declaration followed by the root elements of RDF documents: rdf:RDF. The xmlns: rdf namespace, specifies that elements with the RDF prefix are from the namespace "http://www.w3.org/1999/02/22-rdf-syntax-ns#". The xmlns:h namespace, specifies that elements with the h prefix are from the namespace "http://www.w3.org/TR/html4/cross#". The <rdf: Description> element contains the description of resource identified by the rdf: about attribute. The elements h: type, h: found, h: name *etc* are the properties of the resource.

The image shows a Notepad window titled 'cross - Notepad' containing an RDF document. The document is an XML file with the following content:

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:h="http://www.w3.org/TR/html4/cross#">
<rdf:Description rdf:about="http://en.wikipedia.org/wiki/A._T._Cross_Company">
<h:name>cross</h:name>
<h:type>cross is a name of company</h:type>
<h:found>The company was founded in 1846</h:found>
<h:detail>cross company has various variety of pens</h:detail>
<h:founder>name of founder is richard</h:founder>
</rdf:Description>

<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Jesus">
<h:name>cross</h:name>
<h:type>cross reffered to as jesus christ or simply christ</h:type>
<h:death>jesus was died around 4 bc/bce</h:death>
<h:birth>date of birth between 6 and 4 BC/BCE</h:birth>
<h:detail>Jesus was "about 30 years of age" at the start of his ministry</h:detail>
</rdf:Description>

<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Cross">
<h:name>cross</h:name>
<h:type>geometrical figure consisting of two lines or bars perpendicular to each other.</h:type>
<h:detail>ancient egyptian ankh also known as egyptian cross</h:detail>
</rdf:Description>

<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Cross_product">
<h:name>cross</h:name>
<h:type>the cross product is a binary operation on two vectors</h:type>
<h:detail>cross product on two vectors denoted by a x b</h:detail>
<h:property>cross product anti commutative and not associative</h:property>
</rdf:Description>
</rdf:RDF>
```

Figure 4.1: RDF document for “cross” repository using Notepad

The RDF document has been validated through online W3C RDF validation service. This validation process generates the triples (subject, predicate, and object) for the RDF document (Data Model). The validation process is shown in Figure 4.2 ,4.4,4.6,4.8.

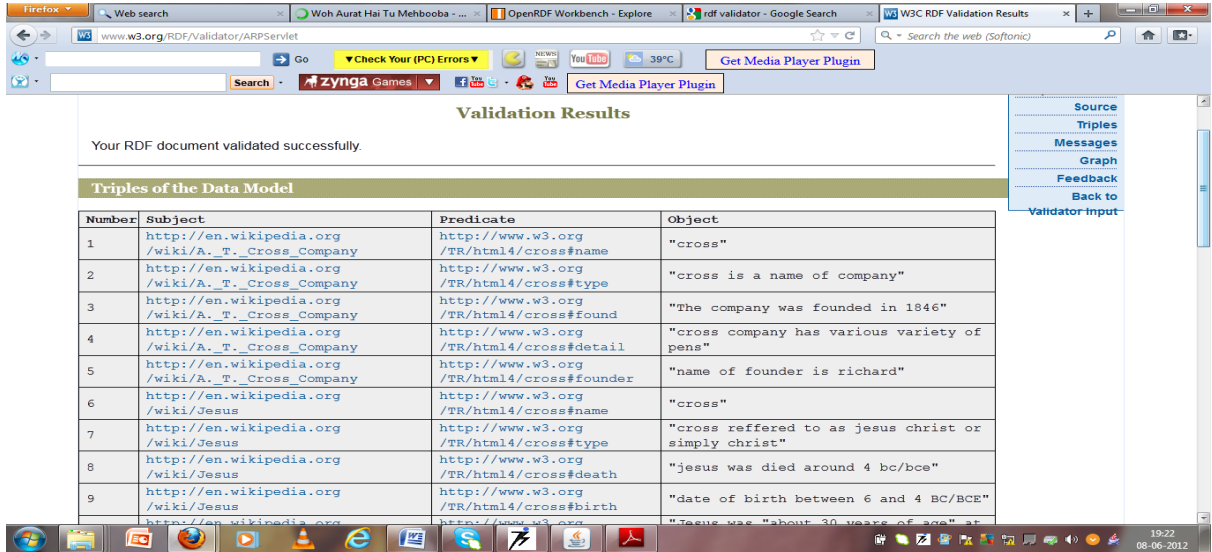


Figure 4.2: Validation of RDF using W3C RDF Validation Service

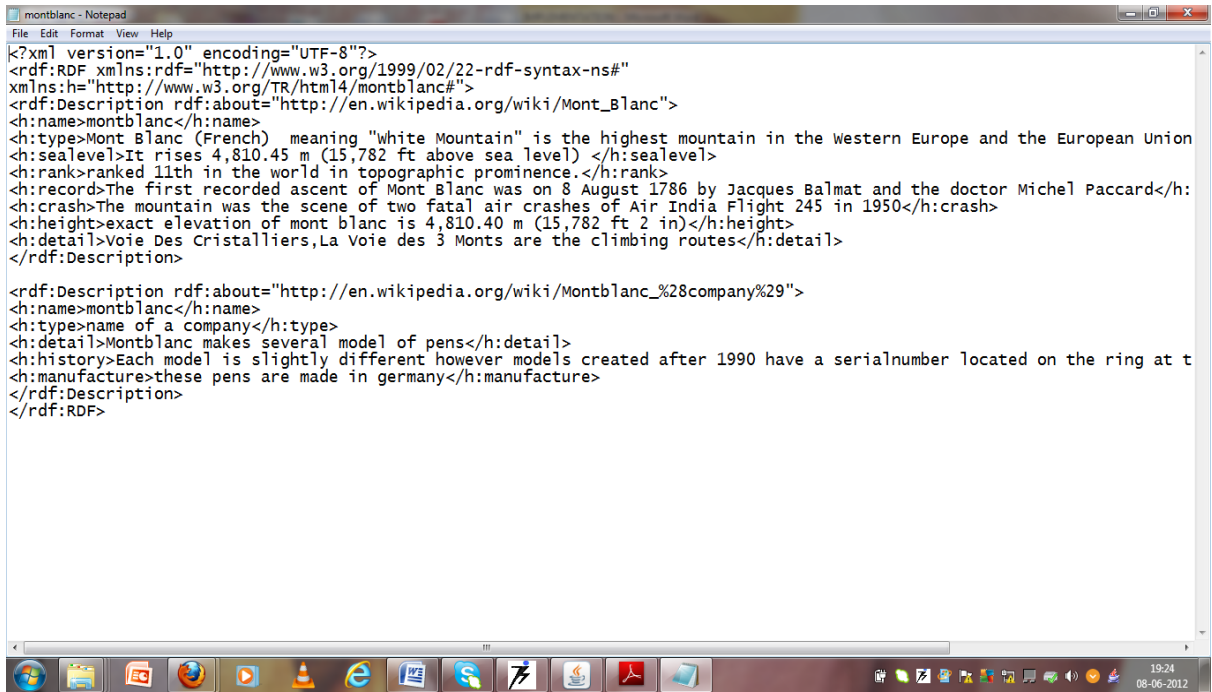


Figure 4.3: RDF document for “Montblanc” repository using Notepad



Figure 4.4: Validation of RDF using W3C RDF Validation Service

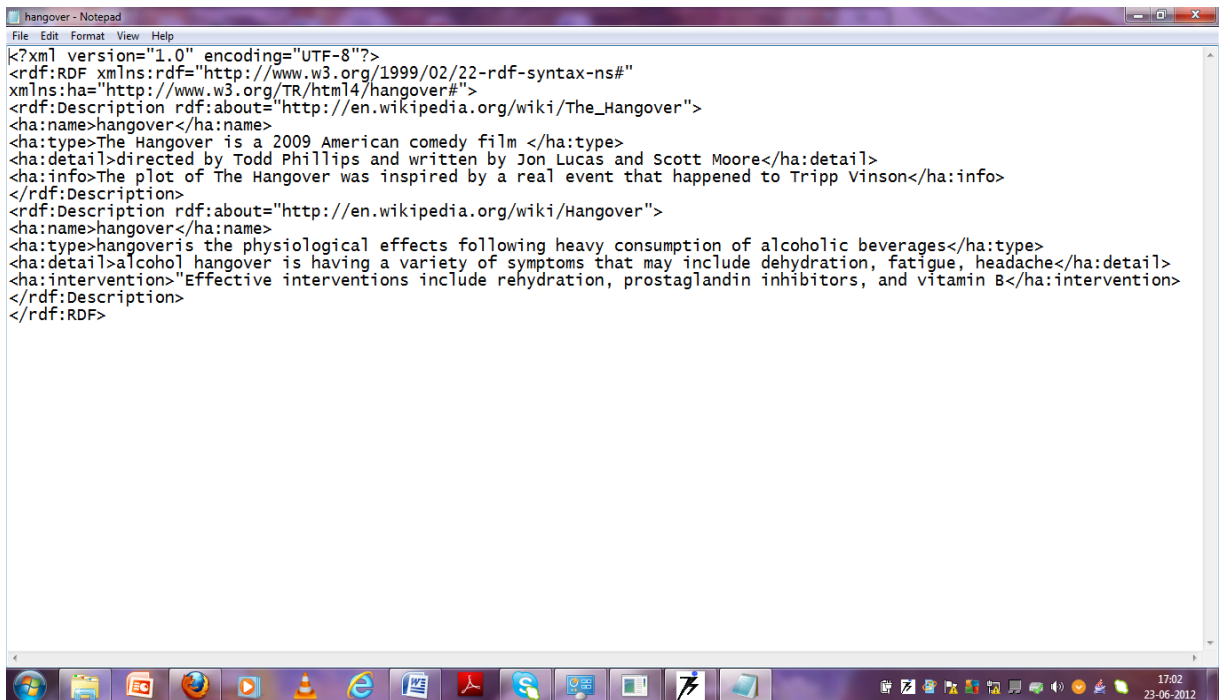


Figure 4.5: RDF document for "hangover" repository using Notepad

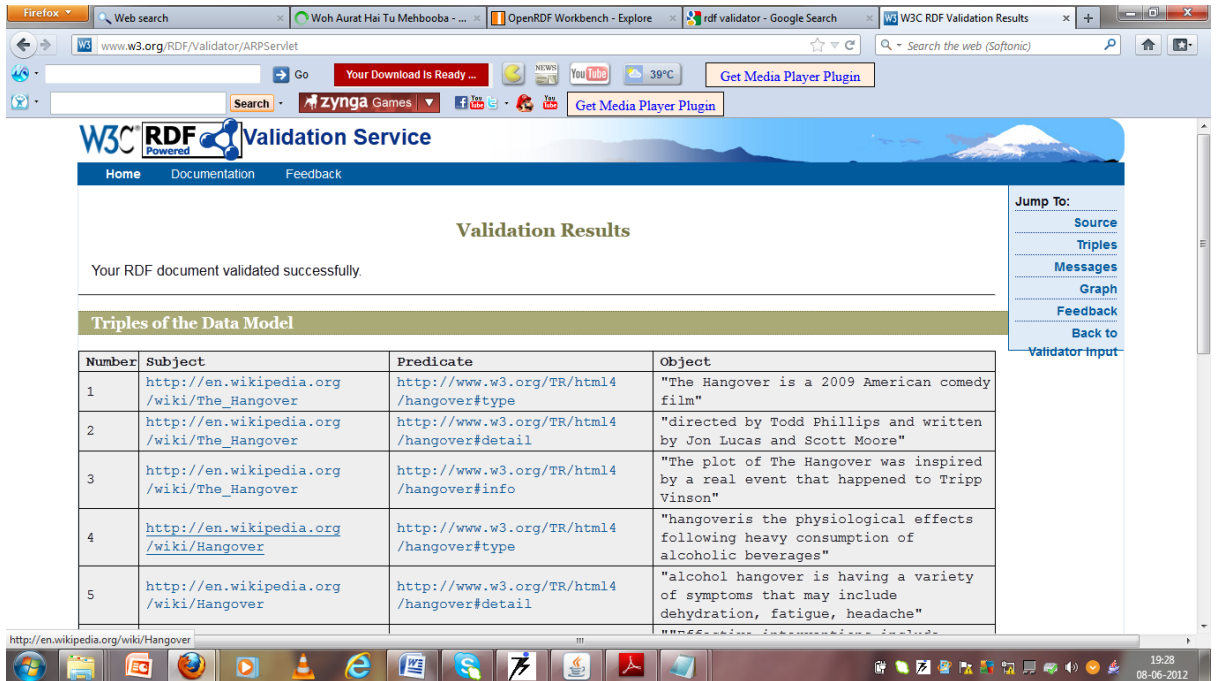


Figure 4.6: Validation of RDF using W3C RDF Validation Service

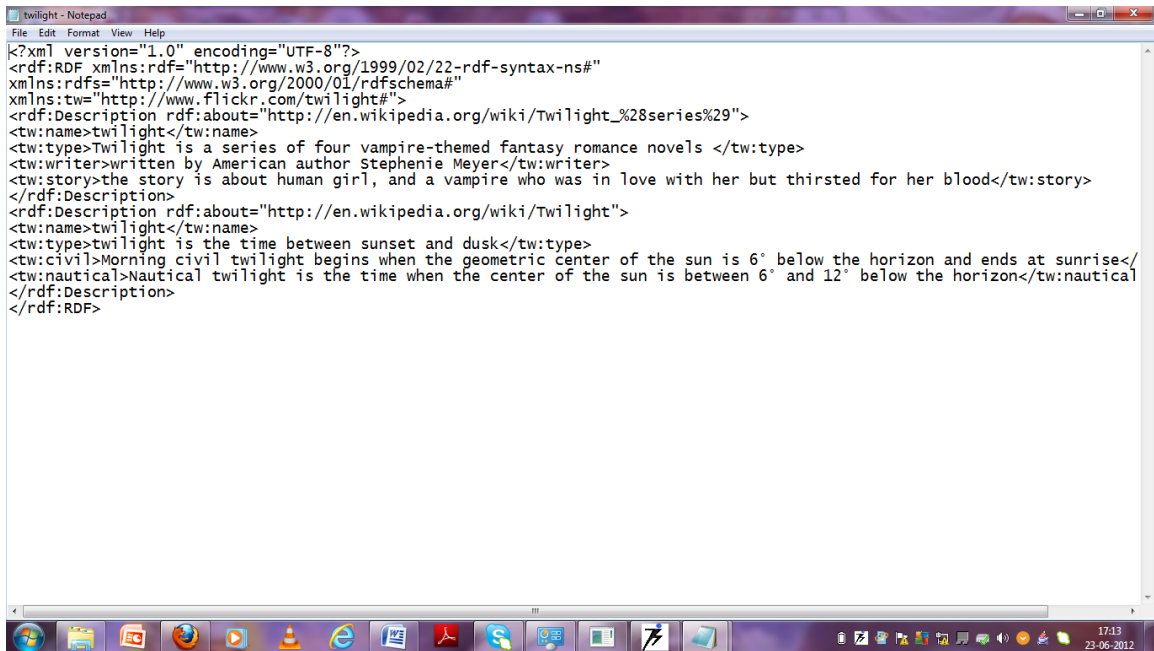


Figure 4.7: RDF document for “Twilight” repository using Notepad

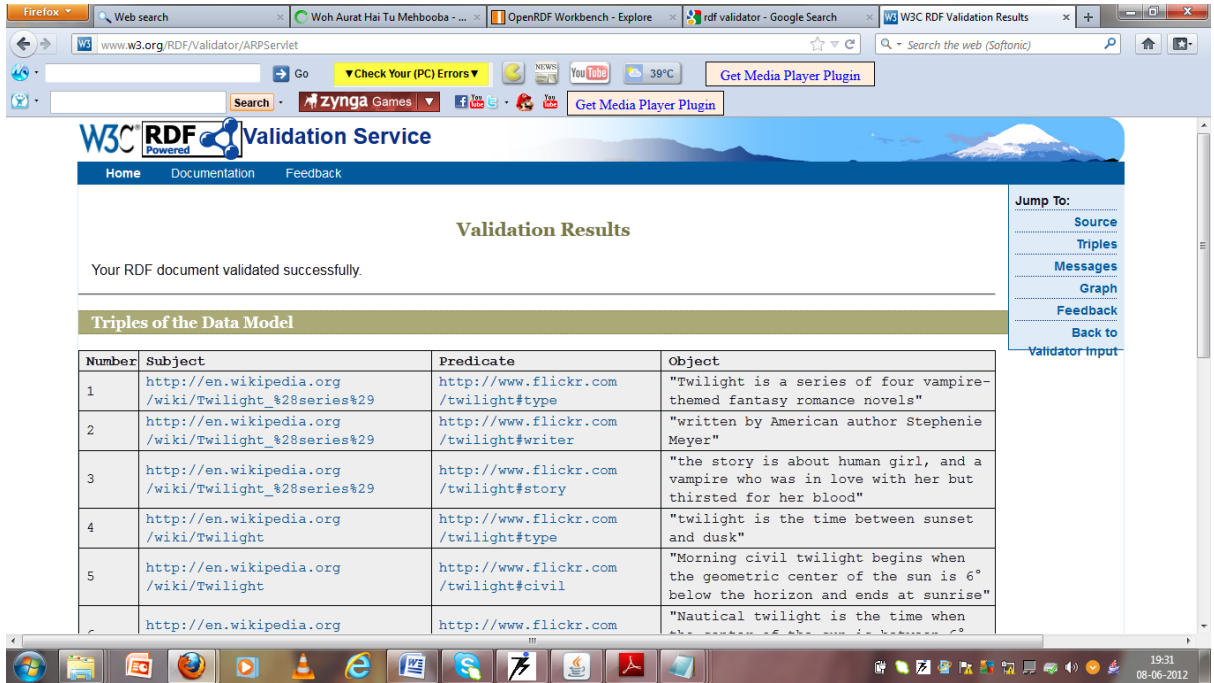


Figure 4.8: Validation of RDF using W3C RDF Validation Service

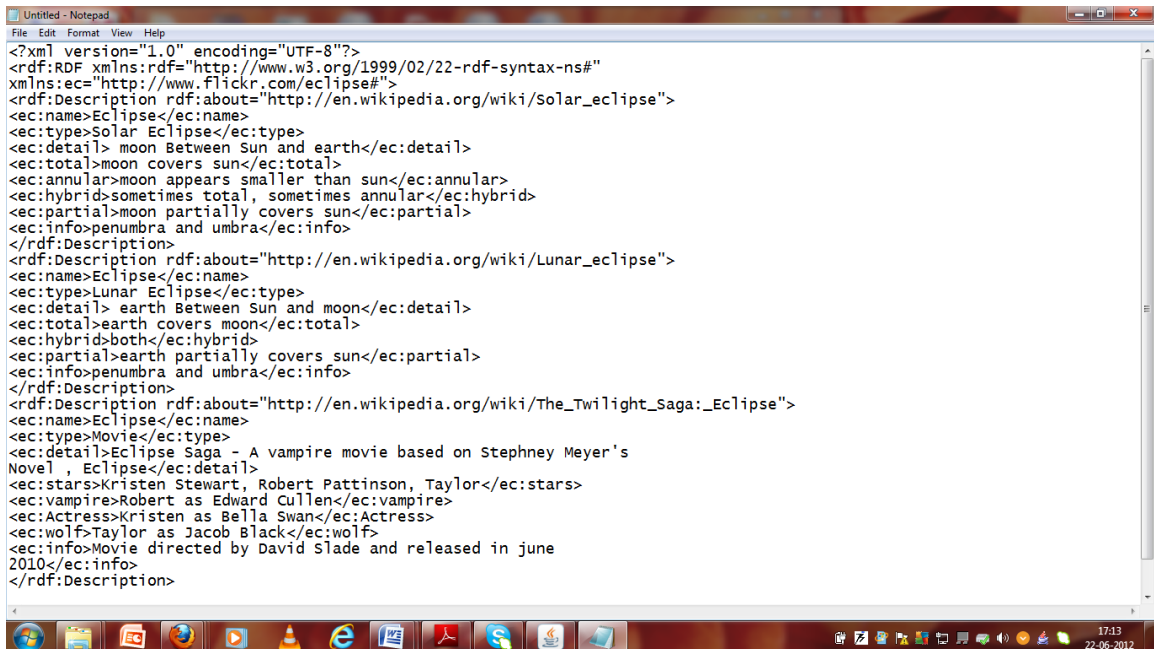


Figure 4.9: RDF document for “Eclipse” repository using Notepad



Figure 4.10: Validation of RDF using W3C RDF Validation Service

4.2 Installation of Sesame

The Sesame 2.0 server software comes in the form of two Java Web Applications:

- Sesame (HTTP) server
- Open RDF Workbench.

Sesame Server provides HTTP access to Sesame repositories. Apart from some functionality to view the server's log messages, it doesn't provide any user oriented functionality which is provided by the Open RDF workbench. Open RDF Workbench provides a web interface for querying, updating and exploring the repositories of a Sesame Server. Firstly, Sesame 2.0 SDK is downloaded. The Sesame Server and Open RDF Workbench are found in the war directory of the SDK. The war-files in this directory are deployed in a Java Servlet Container i.e. apache tomcat and this deployment process is container specific. Installation process starts with installing JAVA 5 or newer with its proper path then the environment variable has to be set to JAVA_HOME and its path must be the location of the java where it is installed.

Next step is to start Apache Tomcat server. This can be done by the following steps:

- Start button
- Run

- cmd
- Go to the location directory where tomcat is installed
- Run Start up

After following these steps Tomcat Server will start and new command prompt window will open with name Tomcat in the title bar and it will as show the message “INFO: Server start up in 5029 ms” the message show that sever has started successfully and the number of milliseconds shown in message may not be same at all the time.

Once Java is installed Apache Tomcat server is installed.

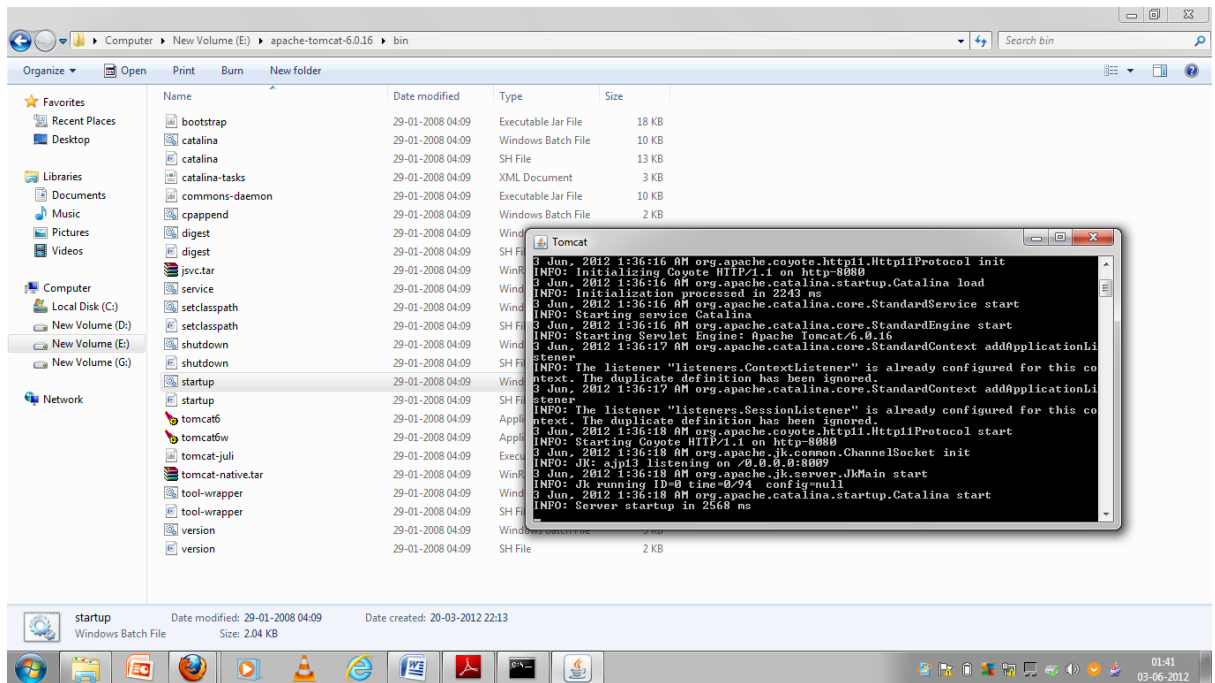


Figure 4.11: Tomcat server start up window

The Sesame Server and OpenRDF Workbench war-files have to be deployed here. After deploying the Sesame Server webapp it can be accessed at path /Open RDF sesame. Similarly, after deployment, the OpenRDF Workbench should be available at path /OpenRDF-workbench. After deploying Open RDF Workbench, create new repository. New Repository is created by select “New Repository” in navigation bar on the left. Enter id and title details in the form that appears and then click next button.

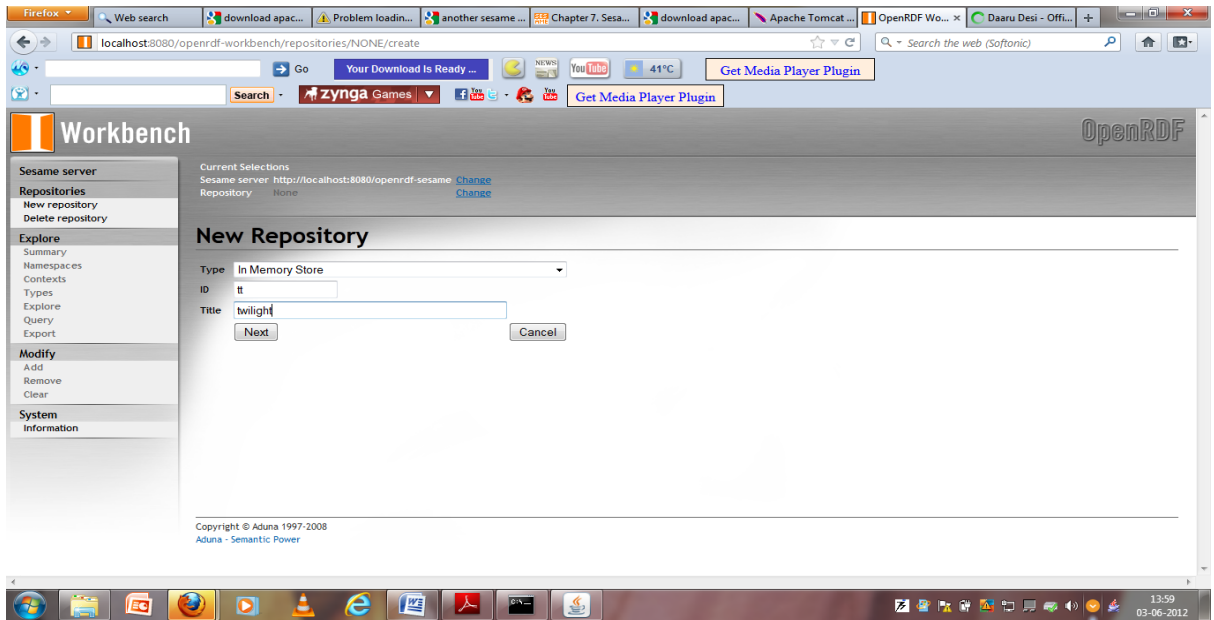


Figure 4.12: Create new repository in Open RDF workbench

The list of repositories along with the default System repository are shown in Figure 4.13

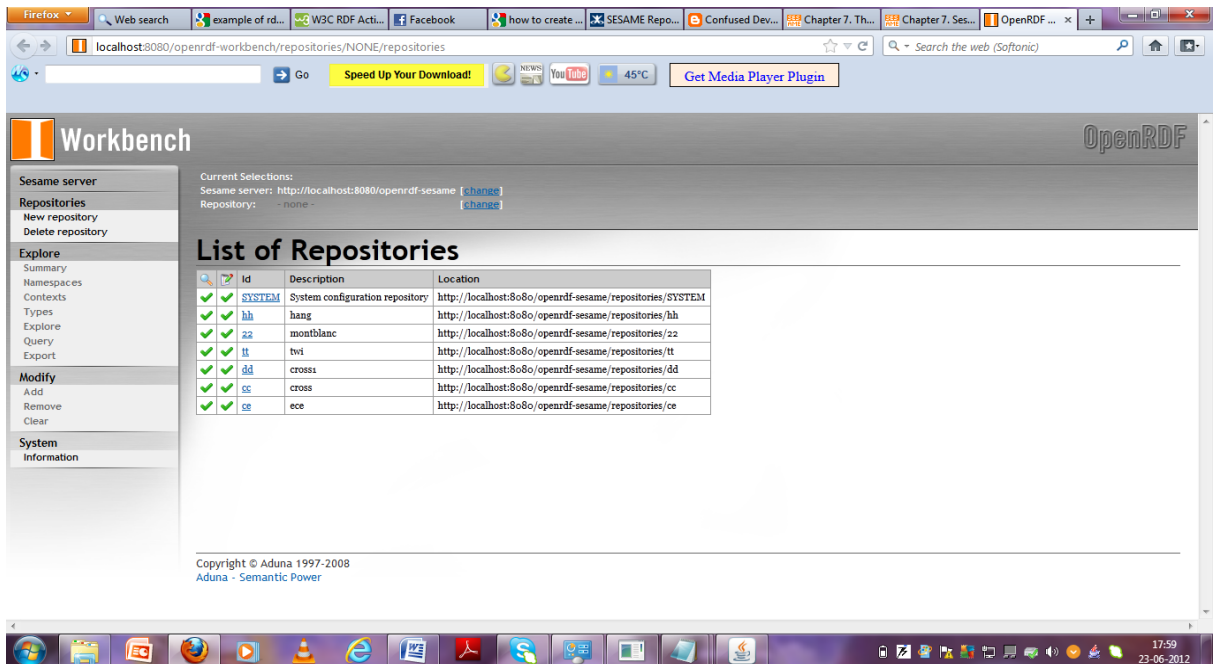


Figure 4.13: List of created RDF Repositories

Next step is to query the repositories using SPARQL query language (Figure 4.14)

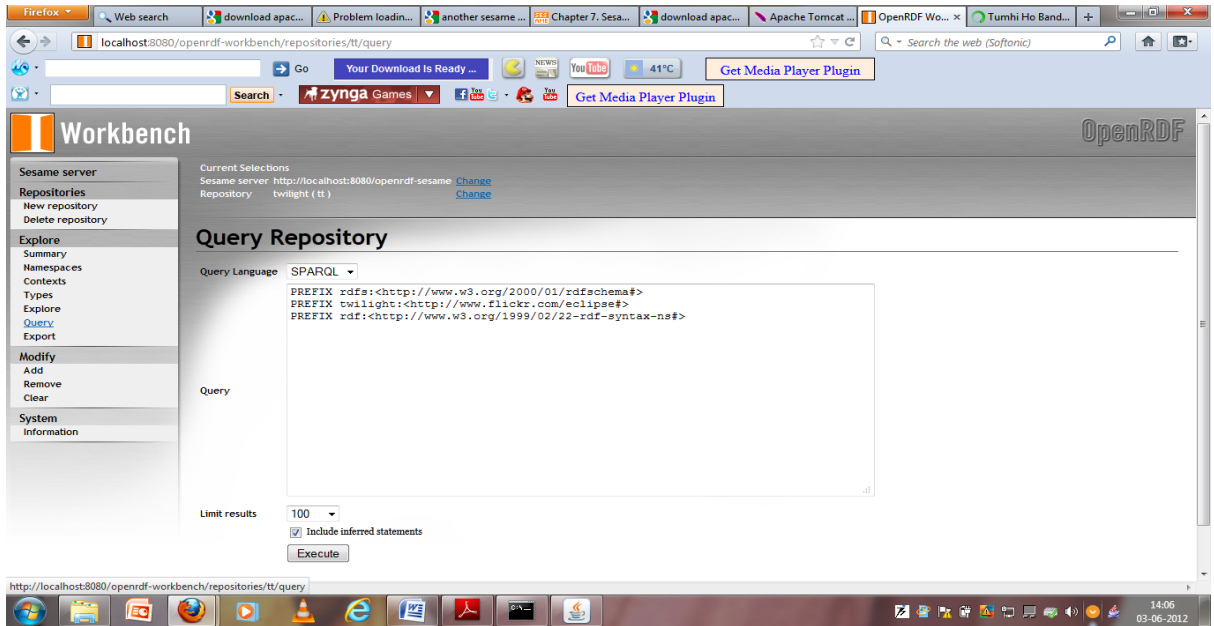


Figure 4.14: Querying repository using Sparql

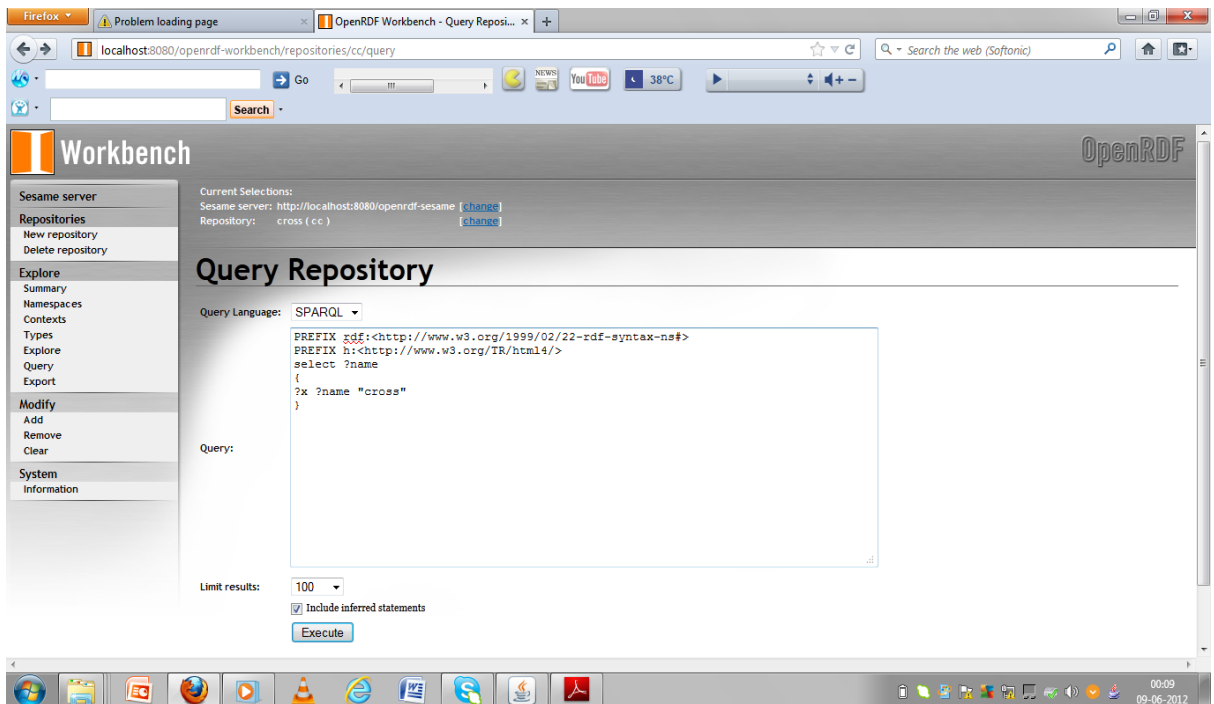


Figure 4.15: Querying on repository “cross”

Querying SPARQL (in Figure 4.15) on RDF repository “cross” by selecting the “name” tag of repository is shown in Figure 4.16

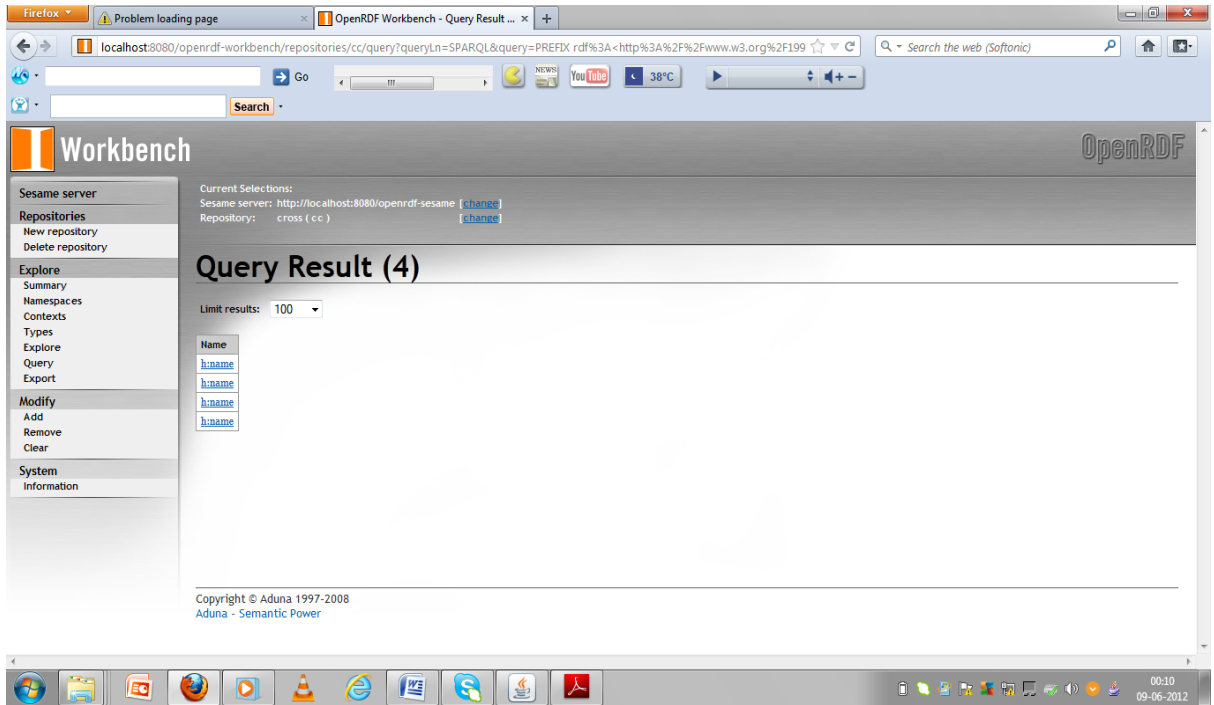


Figure 4.16: Query result for showing the all types of “cross”

By clicking on any link shown in Figure 4.16, all the resources of “cross” are displayed.

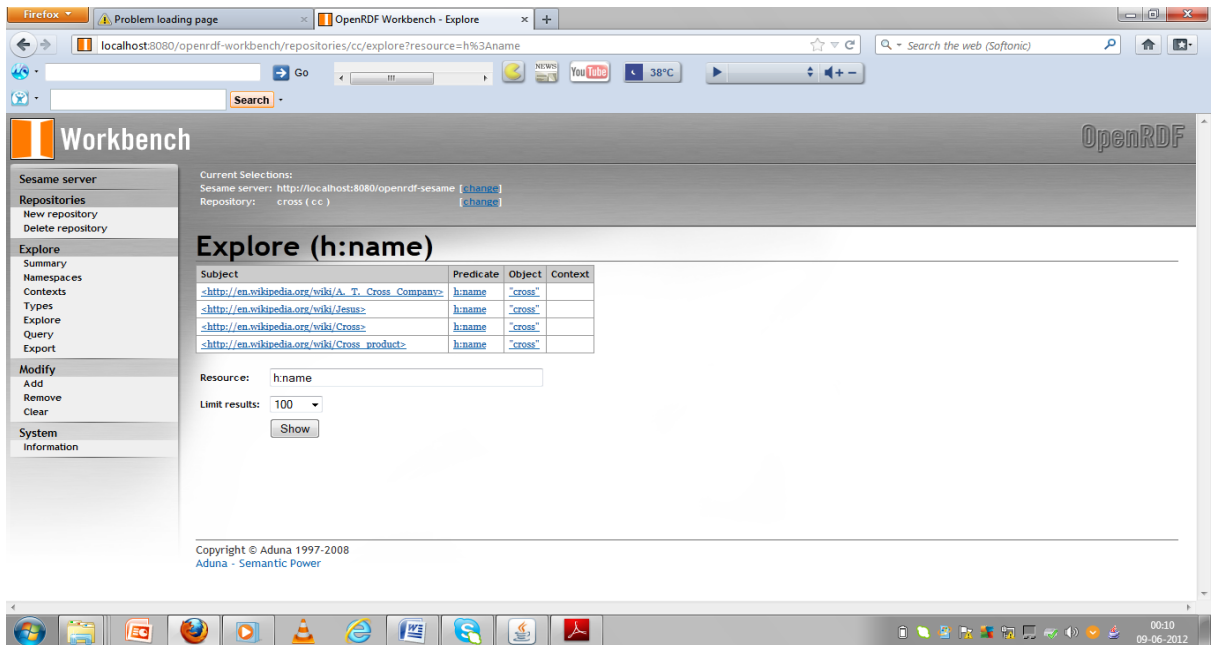


Figure 4.17: Showing all the value of “cross”

Figure 4.17 shows all the resources related to “cross” where user can select the URL as per his requirement.

RDF database stores the metadata (data about data) in the form of triple store, so it gives accurate result for term “cross”.

4.3 Running query on Mysql for “cross” database

A new table is created in Mysql (Figure 4.18)

```

C:\Program Files\MySQL\MySQL Server 5.1\bin>mysql.exe
mysql> select * from cri;
+-----+-----+-----+-----+
| name | type | death | detail | fo |
+-----+-----+-----+-----+
| under | property | birth | found | fo |
+-----+-----+-----+-----+
| cross | cross is the name of company | NULL | cross has various va | |
| ruety of pens | NULL | NULL | founded in 1846 | ri |
| chard | NULL | NULL | NULL | NULL |
| cross | jesus christ cross | died around 4 bc | jesus was about 30 y |
| ears of age when he started ministry | between 6 and 4 bc | NULL | NULL | NU |
| cross | geometrical figure | NULL | egyptian ancient ank |
| h | NULL | null | NULL | NU |
| cross | cross product | NULL | denoted by a*b |
| LL | anti commutative and not associative | null | NULL | NU |
+-----+-----+-----+-----+
4 rows in set (0.05 sec)

mysql> _

```

Figure 4.18: Mysql Database of “cross”

```

C:\Program Files\MySQL\MySQL Server 5.1\bin>mysql.exe
mysql> select * from cri;
+-----+-----+-----+-----+
| name | type | death | detail | fo |
+-----+-----+-----+-----+
| under | property | birth | found | fo |
+-----+-----+-----+-----+
| cross | cross is the name of company | NULL | cross has various va | |
| ruety of pens | NULL | NULL | founded in 1846 | ri |
| chard | NULL | NULL | NULL | NULL |
| cross | jesus christ cross | died around 4 bc | jesus was about 30 y |
| ears of age when he started ministry | between 6 and 4 bc | NULL | NULL | NU |
| cross | geometrical figure | NULL | egyptian ancient ank |
| h | NULL | null | NULL | NU |
| cross | cross product | NULL | denoted by a*b |
| LL | anti commutative and not associative | null | NULL | NU |
+-----+-----+-----+-----+
4 rows in set (0.05 sec)

mysql> select name from cri;
+-----+
| name |
+-----+
| cross |
| cross |
| cross |
| cross |
+-----+
4 rows in set (0.00 sec)

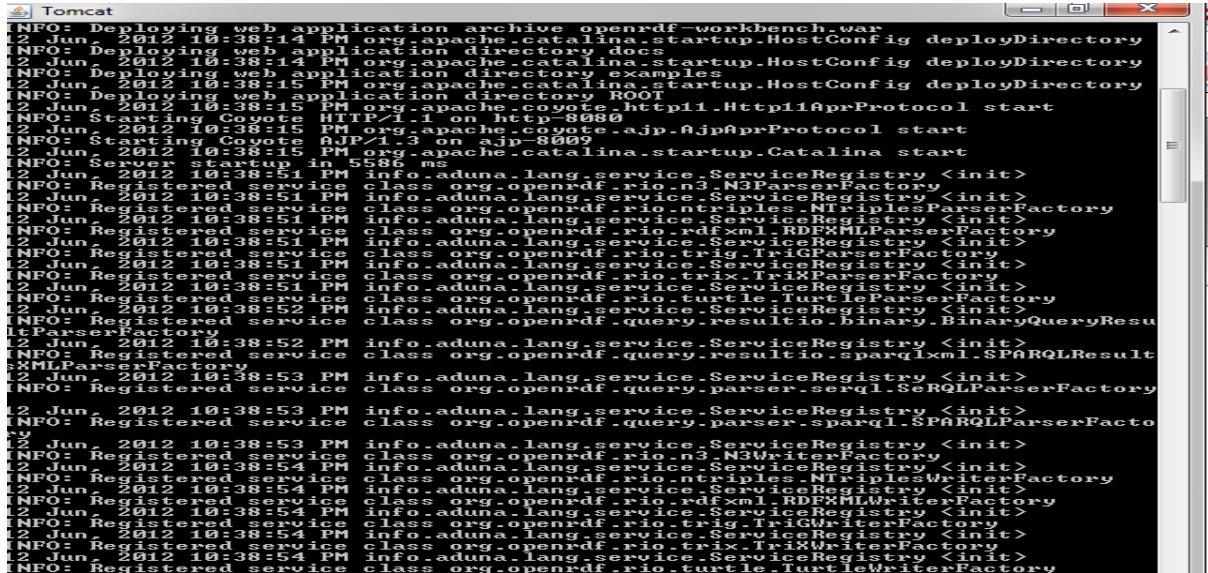
```

Figure 4.19: Query Result for “cross”

While user selects the “name” column from table by giving query, user obtains the result “cross” (only a word), but no details are provided.

4.4 Comparison of query retrieval time

Figure 4.20 depicts Sesame server's log showing the time taken in executing SPARQL query.



```
Tomcat
INFO: Deploying web application archive openrdf-workbench.war
12 Jun 2012 10:38:14 PM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory doc
12 Jun 2012 10:38:14 PM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory examples
INFO: Deploying web application directory ROOT
12 Jun 2012 10:38:15 PM org.apache.coyote.http11.Http11AprProtocol start
INFO: Starting Coyote HTTP/1.1 on http-8080
12 Jun 2012 10:38:15 PM org.apache.coyote.ajp.Ajp13AprProtocol start
INFO: Starting Coyote AJP/1.3 on ajp-8009
12 Jun 2012 10:38:15 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 536 ms
12 Jun 2012 10:38:51 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.n3.N3ParserFactory
12 Jun 2012 10:38:51 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.ntriples.NTriplesParserFactory
12 Jun 2012 10:38:51 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.rdfxml.RDFXMLParserFactory
12 Jun 2012 10:38:51 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.trig.TrigParserFactory
12 Jun 2012 10:38:51 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.trix.TriXParserFactory
12 Jun 2012 10:38:51 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.turtle.TurtleParserFactory
12 Jun 2012 10:38:52 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.query.resultio.Binary.BinaryQueryResultParserFactory
12 Jun 2012 10:38:52 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.query.parser.sparqlxml.SPARQLResultXMLParserFactory
12 Jun 2012 10:38:53 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.query.parser.sparql.SPARQLParserFactory
12 Jun 2012 10:38:53 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.query.parser.sparql.SPARQLParserFactory
12 Jun 2012 10:38:53 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.n3.N3WriterFactory
12 Jun 2012 10:38:54 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.ntriples.NTriplesWriterFactory
12 Jun 2012 10:38:54 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.rdfxml.RDFXMLWriterFactory
12 Jun 2012 10:38:54 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.trig.TrigWriterFactory
12 Jun 2012 10:38:54 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.trix.TriXWriterFactory
12 Jun 2012 10:38:54 PM info.aduna.lang.service.ServiceRegistry <init>
INFO: Registered service class org.openrdf.rio.turtle.TurtleWriterFactory
```

Figure 4.20: Query retrieval time in RDF database

RDF database takes 3 seconds of retrieving the query.



```
File Edit View History Bookmarks Tools Help
http://127.0.0.1/cr.php
http://127.0.0.1/cr.php
BackTrack Linux Offensive Security Exploit-DB Aircrack-ng SESEORG.org Music
cross
3.10
```

Figure 4.21: Query retrieval time in relational database

Figure 4.21 is showing time taken by Relational Database. Relational Database takes 3.10 ms for retrieving same query. Although the difference in time is only .1 ms as the size of the databases will go on increasing this time difference will go on increasing.

4.5 Query for “Montblanc” RDF database

A query running on “Montblanc” shown in figure 4.22 is:

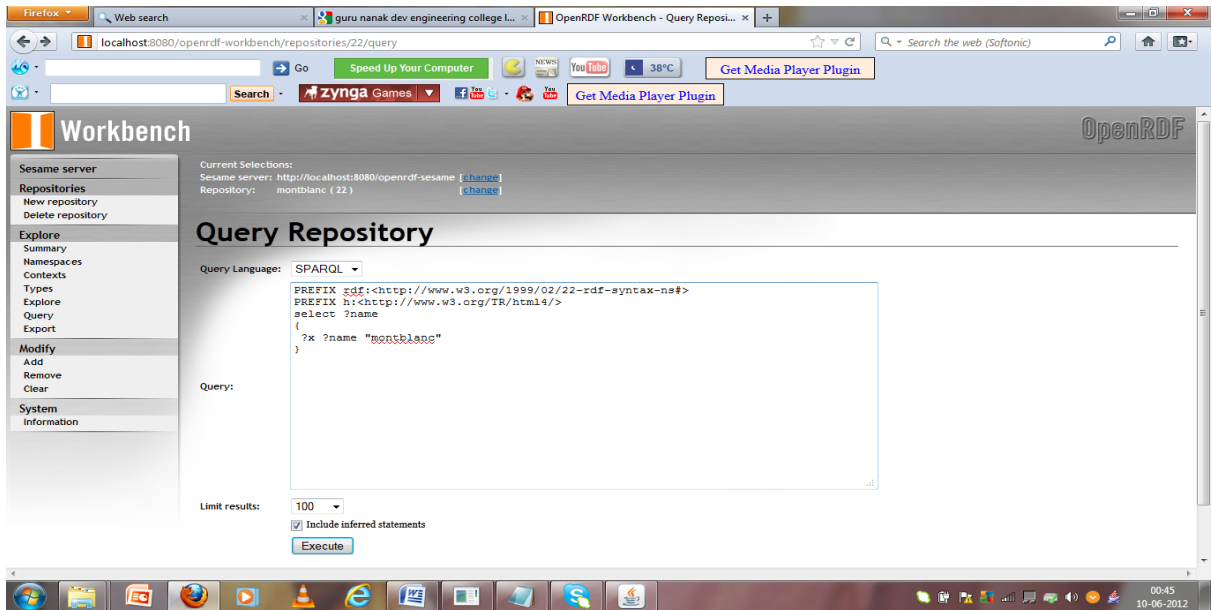


Figure 4.22: Running query for “Montblanc”

While write the query in SPARQL(in Figure 4.22) on RDF repository “Montblanc” by selecting the “name” tag of repository it gives result as shown in Figure 4.23

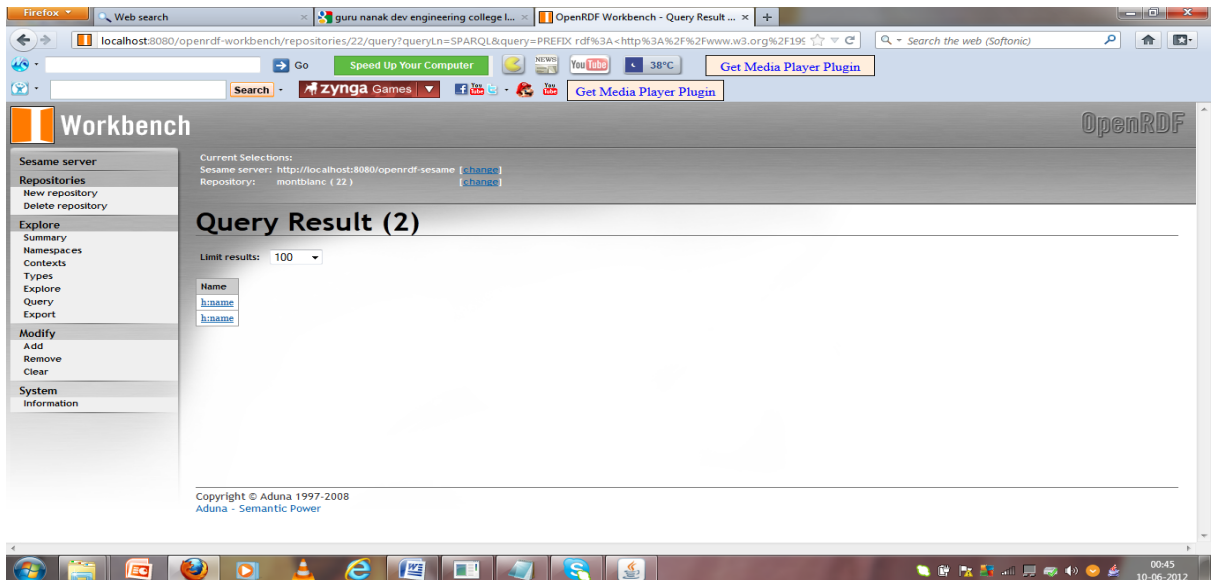


Figure 4.23: Query Result for “Montblanc”

By clicking on any link which are shown in Figure 4.23 it gives all the resources of cross.

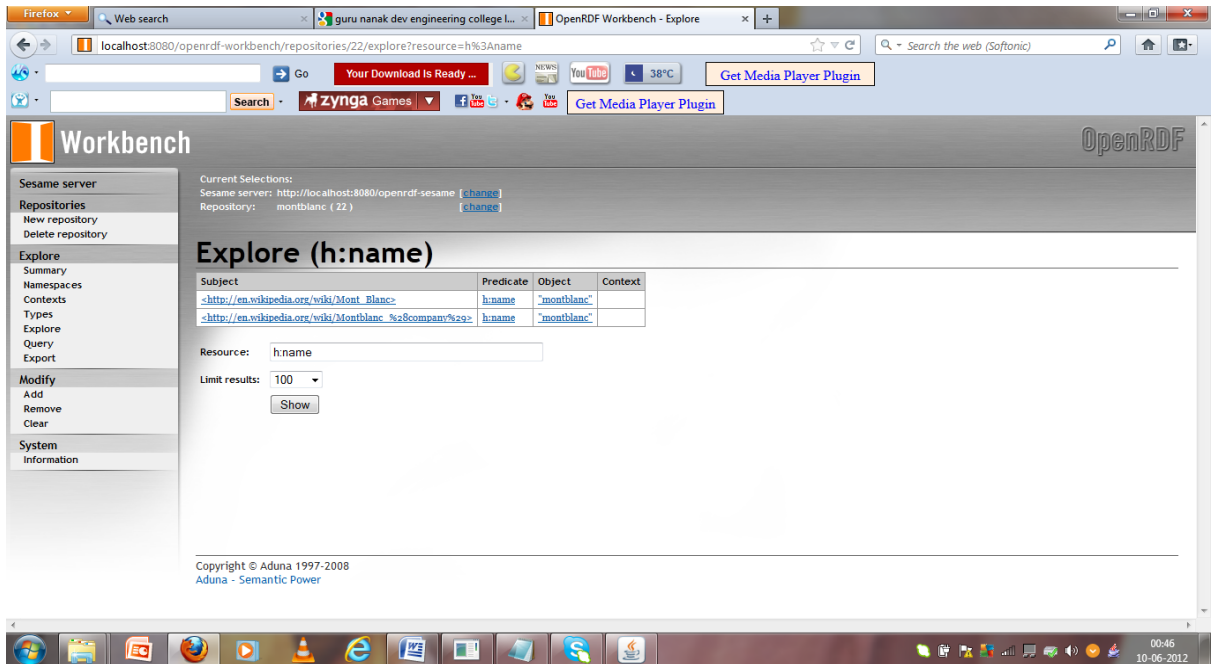


Figure 4.24: Showing all the value of “Montblanc”

Figure 4.24 shows all the resources of “Montblanc” user can selects any one which user wants.

4.6 Query on MySql for “Montblanc”

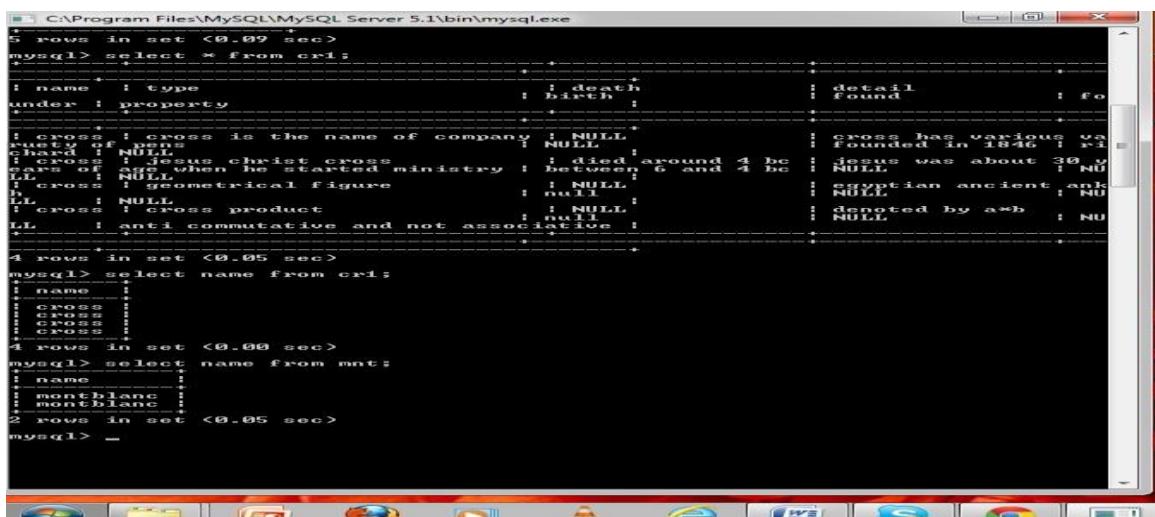


Figure 4.25: Query Result for “Montblanc” in Relational database

While user selects the “name” column from table by giving query, user obtains the result “Montblanc” (only a word), but no details are provided.

Table 4.2 Comparative Analysis of Query Result in RDF and RDBMS

S. No.	Input Query	Time retrieval to run query		Result Retrieve	
		RDF	RDBMS	RDF	RDBMS
1.	“cross”	.045	.05	Cross: http://en.wikipedia.org/wiki/A._T._Cross_Company Cross: http://en.wikipedia.org/wiki/Jesus Cross: http://en.wikipedia.org/wiki/Cross Cross: http://en.wikipedia.org/wiki/Cross_product	Cross Cross Cross Cross
2	“Montblanc”	.035	.07	Montblanc: http://en.wikipedia.org/wiki/Mont_Blanc Montblanc: < http://en.wikipedia.org/wiki/Montblanc_%28company%29 >	Montblac Montblac
3	“Twilight”	.030	.04	Twilight: http://en.wikipedia.org/wiki/Twilight_%28series%29 Twilight: < http://en.wikipedia.org/wiki/Twilight >	Twilight Twilight
4	“Hangover”	.050	.06	Hangover: http://en.wikipedia.org/wiki/The_Hangover Hangover: < http://en.wikipedia.org/wiki/Hangover >	Hangover Hangover
5	“Eclipse”	.080	.085	Eclipse: http://en.wikipedia.org/wiki/Solar_eclipse Eclipse: http://en.wikipedia.org/wiki/Lunar_eclipse Eclipse: http://en.wikipedia.org/wiki/The_Twilight_Saga:_Eclipse	Eclipse Eclipse Eclipse

Table 4.2 provides a comparative analysis of query retrieval time for a input term in RDF database and RDBMS along with the output achieved on querying a specific term using both databases. Results achieved clearly indicate that retrieval time for RDF database is less compared to the time required to retrieve the data in RDBMS. Similary the output result corresponding to the input query in RDBMS is the simple word while RDF database query provide links to all the possible options for the term queried.

Graphical Representation of Query retrieval time in RDF and RDBMS

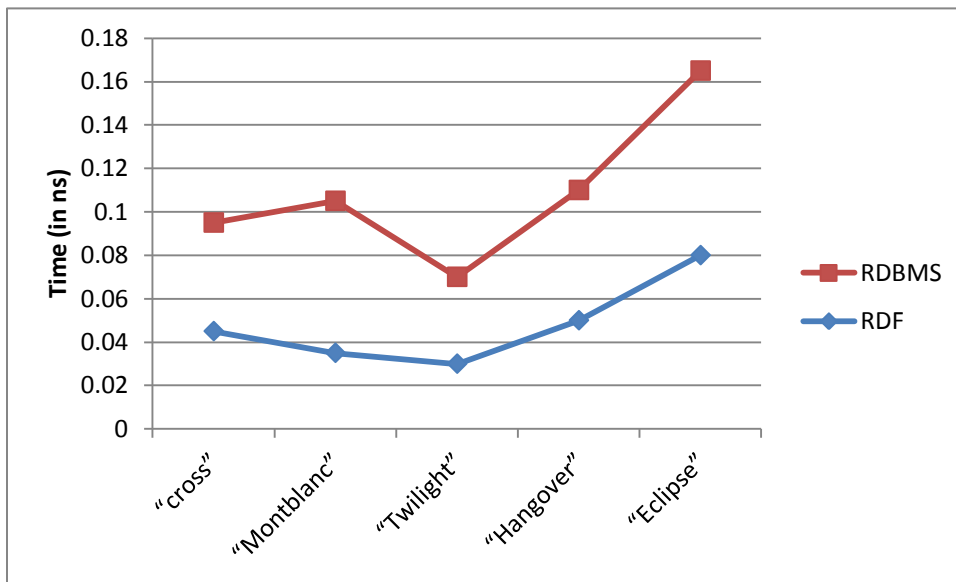


Figure 4.26: Query Retrieval time for terms in RDF and RDBMS

Figure 4.26 gives the graphical representation of the query retrieval time in RDF databases and RDBMS indicating that although the query retrieval time is marginally less in RDF database compared to RDBMS in some cases but overall retrieval time in RDF databases is always less than the time required to query the data in RDBMS.

5.1 Conclusion

In this thesis focus has been on comparative analysis of contextual information retrieved using RDF and relational databases. RDF repositories and RDBMS tables have been created and output retrieved by querying both the data bases with same query terms has been examined. The RDF database is very powerful tool for semantic search as RDF databases stores the meta data in terms of triples i.e. subject, object and predicate. Since machines can understand the data stored in the RDF format, search results are faster and accurate.

It has been experimentally evaluated that RDF data bases are faster and accurate in retrieving contextual information compared to relational databases. Further, less time is required to retrieve the query results in RDF databases compared to relational databases.

5.2 Future Scope

The data considered for storage and retrieval can be further extended to multiple relational tables and multiple RDF repositories.

Major problem with RDF is that query languages used in RDF database are difficult and complicated requiring a good technical understanding. In future, work can be extended to make these RDF query languages easily understandable to all.

At present semantic Web is in its nascent stages and languages like RDF and OWL are being accepted as building blocks for context sensitive search but a lot of standardization needs to be done in this direction.

References

- [1] http://en.wikipedia.org/wiki/Relational_database.
- [2] From Data Federation Pyramid to the Semantic Web Birthday Cake
<http://www.mkbergman.com>.
- [3] “Introduction to HTML”, [Online Document] Available: “<http://www.nios.ac.in>”
- [4] Decker S., Melnik S., Van Harmelen F., Fensel D., Klein M., Broekstra J., Erdmann M., and Horrocks I., “The Semantic Web: The Roles of XML and RDF”, IEEE internet Computing, Vol. 4(5), pp. 63-74, Sept/Oct , 2000.
- [5] “XML Tutorial”, [Online Document] Available: “<http://www.w3schools.com/xml/>”
- [6] “RDF Tutorial” [Online Document] Available: <http://www.w3schools.com/rdf/default.asp>
- [7] Kanellakis P., “Elements of Relational Database Theory”. Handbook of Theoretical Computer Science, North-Holland, to appear.
- [8] Angles R. Gutierrez C., “Querying rdf data from a graph database perspective”. In Proceeding of 2nd European Semantic Web conference, ESWC, pages 346–360, Crete, 2005.
- [9] Navathe S B., “Evolution of data modeling for databases”, Communications of the ACM, 35(9):112–123, 1992.
- [10] Chen P., “The entity-relationship model-toward a unified view of data”, ACM Transactions on Database Systems, 1(1):9–36, 1976.
- [11] Decker S., Melnik S., Van Harmelen F., Fensel D., Klein M., Broekstra., Erdmann M. and Horrocks I., “The Semantic Web: The Roles of XML and RDF”, IEEE Internet Computing, Vol. 4(5), pp. 63-74, Sept/Oct, 2000.
- [12] Resource Description Framework (RDF) [Online document] Available: “<http://www.w3.org/RDF/>”
- [13] Brickley D., Guha R., “Resource Description Framework (RDF) Schema Specification”, W3C Candidate Recommendation, March, 2000, [Online document] Available: <http://www.w3.org/TR/2000/CR-rdf-schema-20000327>.

- [14] Chen H., Wang Y., Wang H., Mao Y., Tang J., “Towards a semantic web of relational databases: A practical semantic toolkit and an in-use case from traditional chinese medicine”.The Semantic Web 2006 Springer
- [15] “Sesame Architecture for Storing and Querying RDF Data”, [Online Document] Available:<http://www.openrdf.org/doc/sesame/users/ch01.html>:
- [16] Ilyas I., Aref W., Elmagarmid A., Elmongui H., Shah R., and Vitter J. “Adaptive rank-aware query optimization in relational databases”. ACM Transaction on Database Systems, 31(4):1257–1304, 2006.
- [17] Al-Ghamdi N., Saleh M., Eassa F., “Ontology-Based Query in Heterogeneous & Distributed Data Sources”, International Journal of Electrical & Computer Sciences IJECS-IJENS Vol: 10 No: 06, 2010
- [18] Shanmugavadivu S.L., “Semantic Based Multiple Web Search Engine”. International Journal on Computer Science and Engineering Vol.02, no. 05, pp 1722-1728, 2010.
- [19] Mayfield J., Finin T., “Information retrieval on the Semantic Web: Integrating inference and retrieval”. SIGIR 2003 Semantic Web Workshop, August 2003
- [20] Wilkinson K., Sayers C, Kuno H, Reynolds D.,“Efficient RDF Storage and Retrieval in Jena2”.
- [21] Stegmaier F., Grobner U., D oller M., Kosch H ., Gero B., “Evaluation of Current RDF Database Solutions”. In Proceedings of the 10th International Workshop on Semantic Multimedia Database Technologies (SeMuDaTe), 4th International Conference on Semantics And Digital Media Technologies (SAMT), pp 39-55,2009.
- [22]“Semantic Web Architecture”, [Online Document] Available: <http://www.obitko.com/tutorials/ontologies-semantic-web-architecture.html>
- [23] Trinh Q., Barker, K., and Alhajj R., “Semantic Interoperability Between Relational Database Systems”, In proceedings of the 11th International Database Engineering and Applications Symposium, Banff, Canada, pp.208- 215,6-8 September 2007.
- [24] Ma L., Mi J., Pan Y., Kulkarni K., Fokoue A., Ranganathan A., “Semantic Web technologies and Data Management”. In Proceeding of W3C Workshop on RDF Access to Relational Databases, 2007
- [25] Sellis T. ,“Multiple query optimization”, ACM Transactions on Database Systems, 10(3), 1986.

- [26] Markowetz A., Yang Y., Papadias D., “Keyword search on relational data streams”.
In SIGMOD, 2007.
- [27] Jason C. Hung, “The Semantic Annotated Documents - From HTML to the
Semantic Web”, Proceedings of the 200 WSEAS International Conference on
Computer Engineering and Applications, Gold Coast, Australia, January 2007,
[Online document] Available: <http://member.mine.tku.edu.tw/www/jhung>
- [28] <http://www.openrdf.org/doc/sesame/users/ch01.html>
- [29] <http://tomcat.apache.org/>
- [30] “Comparison of triple stores” [Online Document]
Available: “<http://www.bioComparison.com> of triple stores”.
- .

List of Publication

1. Amita Arora, Seema Bawa, Shalini Batra, “*Comparative analysis of Relational and RDF Databases*” at International Journal of Soft Computing and Engineering.
(Communicated)

Appendix-A

The original RDF/XML document developed for Cross, Twilight, Montblanc, Hangover

Cross

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:h="http://www.w3.org/TR/html4/cross#">
<rdf:Description rdf:about="http://en.wikipedia.org/wiki/A._T._Cross_Company">
<h:name>cross</h:name>
<h:type>cross is a name of company</h:type>
<h:found>The company was founded in 1846</h:found>
<h:detail>cross company has various variety of pens</h:detail>
<h:founder>name of founder is richard</h:founder>
</rdf:Description>

<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Jesus">
<h:name>cross</h:name>
<h:type>cross reffered to as jesus christ or simply christ</h:type>
<h:death>jesus was died around 4 bc/bce</h:death>
<h:birth>date of birth between 6 and 4 BC/BCE</h:birth>
<h:detail>Jesus was "about 30 years of age" at the start of his ministry</h:detail>
</rdf:Description>

<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Cross">
<h:name>cross</h:name>
```

<h:type>geometrical figure consisting of two lines or bars perpendicular to each other.</h:type>

<h:detail>ancient egyptian ankh also known as egyptian cross</h:detail>

</rdf:Description>

<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Cross_product">

<h:name>cross</h:name>

<h:type>the cross product is a binary operation on two vectors</h:type>

<h:detail>cross product on two vectors denoted by $a \times b$ </h:detail>

<h:property>cross product anti commutative and not associative</h:property>

</rdf:Description>

</rdf:RDF>

Twilight

<?xml version="1.0" encoding="UTF-8"?>

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">

xmlns:rdfs="http://www.w3.org/2000/01/rdfschema#">

xmlns:twilight="http://www.flickr.com/twilight#">

<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Twilight_%28series%29">

<twilight:type>Twilight is a series of four vampire-themed fantasy romance novels

</twilight:type>

<twilight:writer>written by American author Stephenie Meyer</twilight:writer>

<twilight:story>the story is about human girl, and a vampire who was in love with her but thirsted for her blood</twilight:story>

</rdf:Description>

<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Twilight">

<twilight:type>twilight is the time between sunset and dusk</twilight:type>

<twilight:civil>Morning civil twilight begins when the geometric center of the sun is 6° below the horizon and ends at sunrise</twilight:civil>

<twilight:nautical>Nautical twilight is the time when the center of the sun is between 6° and 12° below the horizon</twilight:nautical>

</rdf:Description>

</rdf:RDF>

Hangover

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:hangover="http://www.w3.org/TR/html4/hangover#">
<rdf:Description rdf:about="http://en.wikipedia.org/wiki/The_Hangover">
<hangover:type>The Hangover is a 2009 American comedy film </hangover:type>
<hangover:detail>directed by Todd Phillips and written by Jon Lucas and Scott
Moore</hangover:detail>
<hangover:info>The plot of The Hangover was inspired by a real event that happened to
Tripp Vinson</hangover:info>
</rdf:Description>
<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Hangover">
<hangover:type>hangoveris the physiological effects following heavy consumption of
alcoholic beverages</hangover:type>
<hangover:detail>alcohol hangover is having a variety of symptoms that may include
dehydration, fatigue, headache</hangover:detail>
<hangover:intervention>"Effective interventions include rehydration, prostaglandin
inhibitors, and vitamin B</hangover:intervention>
</rdf:Description>
</rdf:RDF>
```

Montblanc

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:h="http://www.w3.org/TR/html4/montblanc#">
<rdf:Description rdf:about="http://en.wikipedia.org/wiki/Mont_Blanc">
<h:name>montblanc</h:name>
<h:type>Mont Blanc (French) meaning "White Mountain" is the highest mountain in the
Western Europe and the European Union</h:type>
<h:sealevel>It rises 4,810.45 m (15,782 ft above sea level) </h:sealevel>
<h:rank>ranked 11th in the world in topographic prominence.</h:rank>
```

<h:record>The first recorded ascent of Mont Blanc was on 8 August 1786 by Jacques Balmat and the doctor Michel Paccard</h:record>

<h:crash>The mountain was the scene of two fatal air crashes of Air India Flight 245 in 1950</h:crash>

<h:height>exact elevation of mont blanc is 4,810.40 m (15,782 ft 2 in)</h:height>

<h:detail>Voie Des Cristalliers,La Voie des 3 Monts are the climbing routes</h:detail>

</rdf:Description>

<rdf:Description

rdf:about="http://en.wikipedia.org/wiki/Montblanc_%28company%29">

<h:name>montblanc</h:name>

<h:type>name of a company</h:type>

<h:detail>Montblanc makes several model of pens</h:detail>

<h:history>Each model is slightly different however models created after 1990 have a serialnumber located on the ring at the top of the clip</h:history>

<h:manufacture>these pens are made in germany</h:manufacture>

</rdf:Description>

</rdf:RDF>