Journal of Information & Communication Technology
Vol. 3, No. 2, (Fall 2009) 87-98

Semantic Web & its Content Creation Process

Zia Ahmed Shaikh*
Institute of Business &Technology, Biztek, Pakistan
Noor Zaman*
College of Computer Science & IT, King Faisal University Saudi Arabia
Zahid Hussain Khuhro*
Institute of Business &Technology, Biztek, Pakistan

ABSTRACT

In this paper we are discussing the needs of Semantic Web and the process to create content for Semantic Web. We will also discuss the main pillars of this technology and the being used in this process. We will cover in-depth study of RDF - Resource Description Framework, its vocabularies, metadata concepts, data representation model and its syntax. and a short introduction to SPARQL - SPARQL Protocol And RDF Query Language Which is used for querying RDF data.

INSPEC Classification : C6130, C7210, D2080, E0430

Keywords : Semantic Web, Programming Languages, RDF, Query Language

1. INTRODUCTION

Semantics is the study of meaning. (The word "semantic" comes from the Greek semantikos, or "significant meaning," derived from sema, or "sign.") Semantic Web techniques helps in separating meanings of content from data, documents, or from application's code, using technologies based on open standards. The term Semantic Web is used to represent a mesh of information connected up together in such a way as to be easily process able by machines, on a global scale. We can think of it as a globally linked Database, which makes efficient representation of data on World Wide Web.

The Semantic Web was thought up by Tim Berners-Lee, inventor of the WWW, URIs, HTTP, and HTML. And now it is under research phase all over the world (Clarke, 2002).

2. WHO IS WORKING ON IT

As Semantic Web is in its research phase, therefore a dedicated team of people at the World Wide Web consortium (W3C) is working to improve, extend and standardize the system. So Many languages, publications, tools have already been developed. However, Semantic Web technologies are still very much in their infancies, Though the future of

* The material presented by the authors does not necessarily portray the viewpoint of the editors and the management of the Institute of Business and Technology (Biztek) and King Faisal University Saudi Arabia.

*Noor Zaman : nzaman@kfu.edu.sa
*Zahid Hussain Khuhro : zahid_khuhro@yahoo.com
*Zia Ahmed Shaikh : ziaahmedshaikh@gmail.com

the project in common appears to be clear, there seems to be little harmonies about the likely direction and characteristics of the early Semantic Web.

3. NEED OF SEMANTIC WEB

Data is present everywhere on web. That is generally hidden in HTML files and is often useful in some situations, but not in others. The problem with the majority of data on the Web that is in this form at the moment is that it is difficult to use on a large scale, because there is no global system for publishing data in such a way as it can be easily processed by anyone.

For example, the information about local events, sports, weather information, plane departure times, and TV Channel guides... all of this information is presented by numerous sites, but all in HTML. The problem with that is that, in some contexts, it is difficult to use this data in the ways that one might want to have it in a form. So the Semantic Web can be seen as a huge engineering solution... but it is more than that. We will find that as it becomes easier to publish data in a re-purposable form, so more people will want to publish data, and there will be a incidental and effect. We may find that a large number of Semantic Web applications can be used for a variety of different tasks, increasing the modularity of applications on the Web.

4. WHAT MAKES SEMANTIC WEB?

The Semantic Web is generally built on syntaxes which use URIs to represent data, usually in triples based structures: i.e. many triples of URI data that can be held in databases, or interchanged on the world Wide Web using a set of particular syntaxes developed especially for the task. These syntaxes are called "Resource Description Framework" syntaxes.

Based on the metadata, intelligent applications such as semantic portals can be created. Metadata creation includes two major parts. First, the ontologies and vocabularies used as the basis in metadata descriptions are defined. Second, the web resources are annotated with metadata conforming to the definitions.

A crucial question for the breakthrough of the Semantic Web approach is how easily the needed metadata can be created. Annotating data by hand is laborious and resource-consuming and usually economically infeasible with larger datasets. Automation of the annotation process is therefore needed.

5. BASIC BUILDING BLOCKS

To represent the Semantic Web, we'll use the following technologies:

- URIs: A global naming scheme
- RDF: A standard syntax for describing data
- RDF Schema: A standard means of describing the properties of that data.
- OWL: A standard means of describing relationships between data items (ontologies defined with the OWL Web Ontology Language)

6. URI - UNIFORM RESOURCE IDENTIFIER

A URI is simply a Web identifier, like the strings starting with http or ftp that you often see on the World Wide Web. Anyone can create a URI, and the ownership of URIs is clearly delegated, so they form an ideal base technology on top of which to build a global Web. In fact, the World Wide Web is such a thing: anything that has a URI is considered to be "on the Web." Every data object and every data schema/model in the Semantic Web must have a unique URI.

A Uniform Resource Locator (URL) is a URI that, in addition to identifying a resource, provides a means of acting upon or obtaining a representation of that resource by describing its primary access mechanism or network location. For example, the URL http://www.mydomain.com is a URI that identifies a resource (mydomain' home page) and implies that a representation of that resource (such as the home page's current HTML code, as encoded characters) is obtainable through HTTP from a network host named www. mydomain.com.

A Uniform Resource Name (URN) is a URI that identifies a resource by name in a particular namespace. You can use a URN to talk about a resource without implying its location or how to dereference it. For example, the URN urn:biztek:b1-0-7666-98-0 is a URI that, like a Biztek book number, allows one to talk about a book, but doesn't suggest where and how to obtain an actual copy of it.

7. RDF - RESOURCE DESCRIPTION FRAMEWORK

The Resource Description Framework (RDF) is an infrastructure that enables the encoding, exchange and reuse of structured metadata. RDF is an application of XML that imposes needed structural constraints to provide unambiguous methods of expressing semantics. RDF additionally provides a means for publishing both human-readable and machine-processable vocabularies designed to encourage the reuse and extension of metadata semantics among disparate information communities. The structural constraints RDF imposes to support the consistent encoding and exchange of standardized metadata provides for the interchangeability of separate packages of metadata defined by different resource description communities.

7.1 Introduction

The World Wide Web affords unprecedented access to globally distributed information. Metadata, or structured data about data, improves discovery of and access to such information. The effective use of metadata among applications, however, requires common conventions about semantics, syntax, and structure. Individual resource description communities define the semantics, or meaning, of metadata that address their particular needs. Syntax, the systematic arrangement of data elements for machine-processing, facilitates the exchange and use of metadata among multiple applications. Structure can be thought of as a formal constraint on the syntax for the consistent representation of semantics.

The Resource Description Framework (RDF), developed under the sponsorship of the World Wide Web Consortium (W3C), is an infrastructure that enables the encoding, exchange, and reuse of structured metadata. This infrastructure enables metadata interoperability through the design of mechanisms that support common conventions of semantics, syntax, and structure. RDF does not specify semantics for each resource description community, but rather provides the ability for these communities to define metadata elements as needed. RDF uses XML (eXtensible Markup Language) as a common syntax for the exchange and processing of metadata. The XML syntax is a subset of the international text processing standard SGML (Standard Generalized Markup Language) specifically intended for use on the Web. The XML syntax provides vendor independence, user extensibility, validation, human readability, and the ability to represent complex structures. By exploiting the features of XML, RDF imposes structure that provides for the unambiguous expression of semantics and, as such, enables consistent encoding, exchange, and machine-processing of standardized metadata.

RDF supports the use of conventions that will facilitate modular interoperability among separate metadata element sets. These conventions include standard mechanisms for representing semantics that are grounded in a simple, yet powerful, data model discussed below. RDF additionally provides a means for publishing both human-readable and machine-processable vocabularies.

7.2 RDF Vocabularies

Vocabularies are the set of properties, or metadata elements, defined by resource description communities. The ability to standardize the declaration of vocabularies is anticipated to encourage the reuse and extension of semantics among disparate information communities. For example, the Dublin Core Initiative, an international resource description community focusing on simple resource description for discovery, has adopted RDF. Educom's IMS Instructional Metadata System, designed to provide access to educational materials, has adopted the Dublin Core and corresponding architecture and extended it with domain-specific semantics. RDF is designed to support this type of semantic modularity by creating an infrastructure that supports the combination of distributed attribute registries. Thus, a central registry is not required. This permits communities to declare vocabularies which may be reused, extended and/or refined to address application or domain specific descriptive requirements.

The goals of RDF are broad, and the potential opportunities are enormous. This introduction to RDF begins by discussing the background context of the RDF initiative and relates it to other metadata activities. A discussion of the functionality of RDF and an overview of the model, schema and syntactic considerations of this framework follow.

7.3 Metadata

The history of metadata at the W3C began in 1995 with PICS, the Platform for Internet Content Selection. PICS is a mechanism for communicating ratings of web pages from a server to clients. These ratings, or rating labels, contain information about the content of web pages: for example, whether a particular page contains a peer-reviewed research article, or was authored by an accredited researcher, or contains violence, foul language, etc. Instead of being a fixed set of criteria, PICS introduced a general mechanism for creating rating systems. Different organizations could rate content based on their own objectives and values, and users -- for example, parents worried about their children's web usage -- could set their browsers to filter out any web pages not matching their own criteria. Development of PICS was motivated by the anticipation of restrictions on Internet content in the USA and elsewhere.

Through a series of meetings with the digital library community, limitations in the PICS specifications were identified, and functional requirements were outlined to address the more general problem of associating descriptive information with Internet resources based on the PICS architecture. As a result of these discussions, the W3C formed a new working group, PICS-NG Next Generation to address the more general issues of resource description.

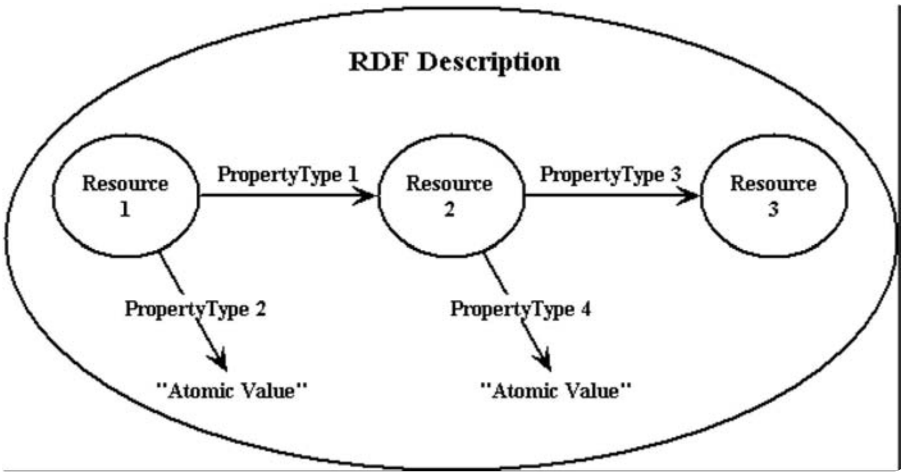
Shortly after the PICS-NG working group was chartered, it became clear that the infrastructure designed in the early document specifications was applicable in several additional applications. As a result, the W3C consolidated these applications as the W3C Resource Description Framework working group.

RDF is the result of a number of metadata communities bringing together their needs to provide a robust and flexible architecture for supporting metadata on the web. While the development of RDF as a general metadata framework, and as such, a simple knowledge representation mechanism for the web, was heavily inspired by the PICS specification, no one individual or organization invented RDF. RDF is a collaborative design effort. Several W3C Member companies are contributing intellectual resources. It is drawing upon the XML design as well as proposals submitted by Microsoft's and Netscape. Other metadata efforts, such as the Dublin Core and the Warwick Framework have also influenced the design of the RDF.

7.4 The RDF Data Model

RDF provides a model for describing resources. Resources have properties (attributes or characteristics). RDF defines a resource as any object that is uniquely identifiable by an Uniform Resource Identifier (URI). The properties associated with resources are identified by property-types, and property-types have corresponding values. Property-types express the relationships of values associated with resources. In RDF, values may be atomic in nature (text strings, numbers, etc.) or other resources, which in turn may have their own properties. A collection of these properties that refers to the same resource is called a description. At the core of RDF is a syntax-independent model for representing resources and their corresponding descriptions. The following figure illustrates a generic RDF description.

Figure: 7.4 a



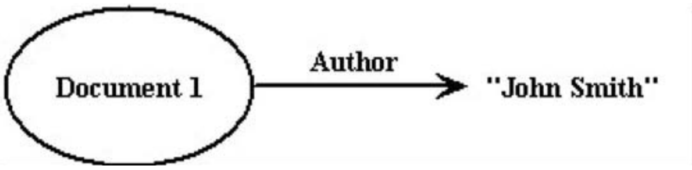
The application and use of the RDF data model can be illustrated by concrete examples. Consider the following statements

1. "The author of Document 1 is John Smith"
2. "John Smith is the author of Document 1"

To humans, these statements convey the same meaning (that is, John Smith is the author of a particular document). To a machine, however, these are completely different strings. Whereas humans are extremely adept at extracting meaning from differing syntactic constructs, machines remain grossly inept. Using a triadic model of resources, property-types and corresponding values, RDF attempts to provide an unambiguous method of expressing semantics in a machine-readable encoding.

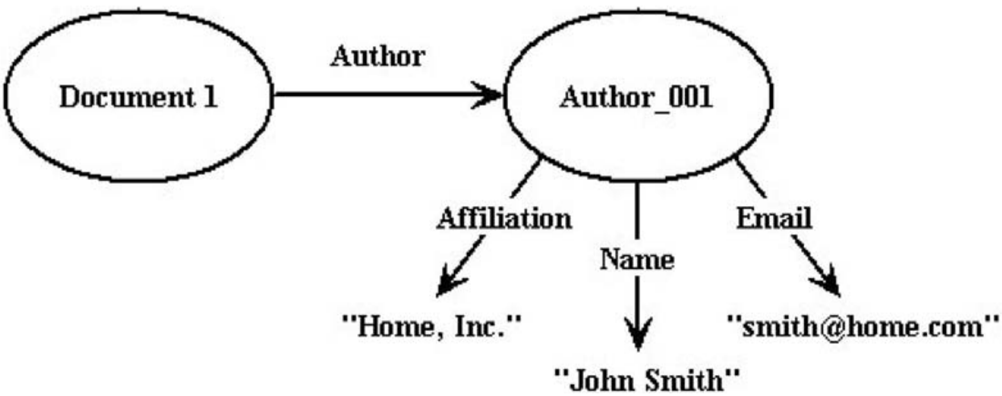
RDF provides a mechanism for associating properties with resources. So, before anything about Document 1 can be said, the data model requires the declaration of a resource representing Document 1. Thus, the data model corresponding to the statement "the author of Document 1 is John Smith" has a single resource Document 1, a property-type of author and a corresponding value of John Smith. To distinguish characteristics of the data model, the RDF Model and Syntax specification represents the relationships among resources, property-types, and values in a directed labeled graph. In this case, resources are identified as nodes, property-types are defined as directed label arcs, and string values are quoted. Given this representation, the data model corresponding to the statement is graphically expressed as.

Figure: 7.4 b



If additional descriptive information regarding the author were desired, e.g., the author's email address and affiliation, an elaboration on the previous example would be required. In this case, descriptive information about John Smith is desired. As was discussed in the first example, before descriptive properties can be expressed about the person John Smith, there needs to be a unique identifiable resource representing him. Given the directed label graph notation in the previous example, the data model corresponding to this description is graphically represented as

Figure: 7.4 c



In this case, "John Smith" the string is replaced by a uniquely identified resource denoted by Author_001 with the associated property-types of name, email and affiliation. The use of unique identifiers for resources allows for the unambiguous association of properties. This is an important point, as the person John Smith may be the value of several different property-types. John Smith may be the author of Document 1, but also may be the value of a particular company describing the set of current employees. The unambiguous identification of resources provides for the reuse of explicit, descriptive information.

In the previous example the unique identifiable resource for the author was created, but not for the author's name, email or affiliation. The RDF model allows for the creation of resources at multiple levels. Concerning the representation of personal names, for example, the creation of a resource representing the author's name could have additionally been described using "firstname", "middlename" and "surname" property-types. Clearly, this iterative descriptive process could continue down many levels. What, however, are the practical and logical limits of these iterations?

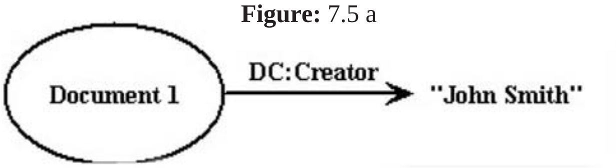
There is no one right answer to this question. The answer is dependent on the domain requirements. These issues must be addressed and decided upon in the standard practice of individual resource description communities. In short, experience and knowledge of the domain dictate which distinctions should be captured and reflected in the data model.

The RDF data model additionally provides for the description of other descriptions. For instance, often it is important to assess the credibility of a particular description (e.g., "The Library of Congress told us that John Smith is the author of Document 1"). In this case the description tells us something about the statement "John Smith is the author of Document 1", specifically, that the Library of Congress asserts this to be true. Similar constructs are additionally useful for the description of collections of resources. For instance, "John Smith is the author of Documents 1, 2, and 3". While these statements are significantly more complex, the same data model is applicable. A more detailed discussion of these issues is outside the scope of this overview, but more information is available in the RDF Model and Syntax Specification.

7.5 The RDF Syntax

RDF defines a simple, yet powerful model for describing resources. A syntax representing this model is required to store instances of this model into machine-readable files and to communicate these instances among applications. XML is this syntax. RDF imposes formal structure on XML to support the consistent representation of semantics. RDF provides the ability for resource description communities to define semantics. It is important, however, to disambiguate these semantics among communities. The property-type "author", for example, may have broader or narrower meaning depending on different community needs. As such, it is problematic if multiple communities use the same property-type to mean very different things. To prevent this, RDF uniquely identifies property-types by using the XML namespace mechanism. XML namespaces provide a method for unambiguously identifying the semantics and conventions governing the particular use of property-types by uniquely identifying the governing authority of the vocabulary. For example, the property-type "author" defined by the Dublin Core Initiative as the "person or organization responsible for the creation of the intellectual content of the resource" and is specified by the Dublin Core CREATOR element. An XML namespace is used to unambiguously identify the Schema for the Dublin Core vocabulary by pointing to the definitive Dublin Core resource that defines the corresponding semantics. If the Dublin Core RDF Schema, however, is abbreviated as "DC", the data model representation for this example would be

Figure: 7.5 a



This more explicit declaration identifies a resource Document 1 with the semantics of property-type Creator unambiguously defined in the context of DC (the Dublin Core vocabulary). The value of this property-type is John Smith.

The corresponding syntactic way of expressing this statement using XML namespaces to identify the use of the Dublin Core Schema is

Script: 7.5 a

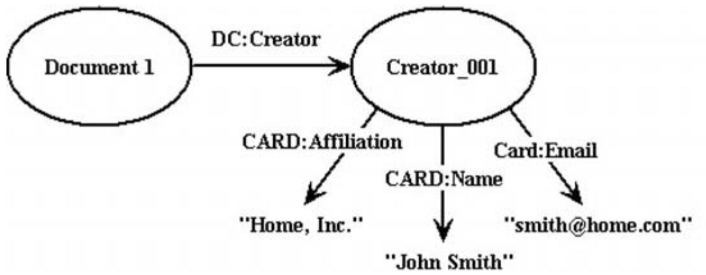
```
<?xml:namespace ns = "http://www.w3.org/RDF/RDF/" prefix ="RDF" ?>
<?xml:namespace ns = "http://purl.oclc.org/DC/" prefix = "DC" ?>

<RDF:RDF>
  <RDF:Description RDF:HREF = "http://uri-of-Document-1">
    <DC:Creator>John Smith</DC:Creator>
  </RDF:Description>
</RDF:RDF>
```

In this case, both the RDF and Dublin Core schemas are declared and abbreviated as "RDF" and "DC" respectively. The RDF Schema is declared as a boot-strapping mechanism for the declaration of the necessary vocabulary needed for expressing the data model. The Dublin Core Schema is declared in order to utilize the vocabulary defined by this community. The URI associated with the namespace declaration references the corresponding schemas. The element <RDF:RDF> (which can be interpreted as the element RDF in the context of the RDF namespace) is a simple wrapper that marks the boundaries in an XML document where the content is explicitly intended to be mappable into an RDF data model instance. The element <RDF:Description> (the element Description in the context of the RDF namespace) is correspondingly used to denote or instantiate a resource with the corresponding URI http://uri-of-Document-1. And the element <DC:Creator> in the context of the <RDF:Description> represents a property-type DC:Creator and a value of "John Smith". The syntactic representation is designed to reflect the corresponding data model.

In the more advanced example, where additional descriptive information regarding the author is required, similar syntactic constructs are used. In this case, while it may still be desirable to use the Dublin Core CREATOR property-type to represent the person responsible for the creation of the intellectual content, additional property-types "name", "email" and "affiliation" are required. For this case, since the semantics for these elements are not defined in Dublin Core, an additional resource description standard may be utilized. It is feasible to assume the creation of an RDF schema with the semantics similar to the vCard specification designed to automate the exchange of personal information typically found on a traditional business card, could be introduced to describe the author of the document. The data model representation for this example with the corresponding business card schema defined as CARD would be (Figure 5).

Figure: 7.5 b



This, in turn, could be syntactically represented as

Script 7.5

```
<?xml:namespace ns = "http://www.w3.org/RDF/RDF/" prefix = "RDF" ?>
<?xml:namespace ns = "http://purl.oclc.org/DC/" prefix = "DC" ?>
<?xml:namespace ns = "http://person.org/BusinessCard/" prefix = "CARD" ?>

<RDF:RDF>
  <RDF:Description RDF:HREF = "http://uri-of-Document-1">
    <DC:Creator RDF:HREF = "#Creator_001"/>
  </RDF:Description>

  <RDF:Description ID="Creator_001">
    <CARD:Name>John Smith</CARD:Name>
    <CARD:Email>smith@home.net</CARD:Email>
    <CARD:Affiliation>Home, Inc.</CARD:Affiliation>
  </RDF:Description>
</RDF:RDF>
```

in which the RDF, Dublin Core, and the "Business Card" schemas are declared and abbreviated as "RDF", "DC" and "CARD" respectively. In this case, the value associated with the property-type DC:Creator is now a resource. While the reference to the resource is an internal identifier, an external URI, for example, to a controlled authority of names, could have been used as well. Additionally, in this example, the semantics of the Dublin Core CREATOR element have been refined by the semantics defined by the schema referenced by CARD. This construct is similar to the Warwick Framework, a recognition of separate maintainable and interchangeable packages of descriptive information used in the description of resources. The structural constraints RDF imposes to support the consistent encoding and exchange of standardized metadata provides for the interchangeability of separate packages of metadata defined by different resource description communities.

Ref: (Clarke, 2002; W3C, 2003; Fensel; http://www.w3.org, Aug, 15th 2008).

8. THE RDF SCHEMA

RDF Schemas are used to declare vocabularies, the sets of semantics property-types defined by a particular community. RDF schemas define the valid properties in a given RDF description, as well as any characteristics or restrictions of the property-type values themselves. The XML namespace mechanism serves to identify RDF Schemas.

A human and machine-process able description of an RDF schema may be accessed by de-referencing the schema URI. If the schema is machine-processable, it may be possible for an application to learn some of the semantics of the property-types named in the schema. To understand a particular RDF schema is to understand the semantics of each of the properties in that description. RDF schemas are structured based on the RDF data model. Therefore, an application that has no understanding of a particular schema will still be able to parse the description into the property-type and corresponding values and will be able to transport the description intact (e.g., to a cache or to another application).

The exact details of RDF schemas are currently being discussed in the W3C RDF Schema working group. It is anticipated, however, that the ability to formalize human-readable and machine-processable vocabularies will encourage the exchange, use, and extension of metadata vocabularies among disparate information communities. RDF schemas are being designed to provide this type of formalization.

RDF properties may be thought of as attributes of resources and in this sense correspond to traditional attribute-value pairs. RDF properties also represent relationships between resources.

RDF however, provides no mechanisms for describing these properties, nor does it provide any mechanisms for describing the relationships between these properties and other resources. That is the role of the RDF vocabulary description language, RDF Schema. RDF Schema defines classes and properties that may be used to describe classes, properties and other resources.

This document does not specify a vocabulary of descriptive properties such as "author". Instead it specifies mechanisms that may be used to name and describe properties and the classes of resource they describe.

RDF's vocabulary description language, RDF Schema, is a semantic extension of RDF. It provides mechanisms for describing groups of related resources and the relationships between these resources. RDF Schema vocabulary descriptions are written in RDF using the terms described in this document. These resources are used to determine characteristics of other resources, such as the domains and ranges of properties.

The RDF vocabulary description language class and property system is similar to the type systems of object-oriented programming languages such as Java. RDF differs from many such systems in that instead of defining a class in terms of the properties its instances may have, the RDF vocabulary description language describes properties in terms of the classes of resource to which they apply. This is the role of the domain and range mechanisms described in this specification. For example, we could define the eg:author property to have a domain of eg:Document and a range of eg:Person, whereas a classical object oriented system might typically define a class eg:Book with an attribute called eg:author of type eg:Person. Using the RDF approach, it is easy for others to subsequently define additional properties with a domain of eg:Document or a range of eg:Person. This can be done without the need to re-define the original description of these classes. One benefit of the RDF property-centric approach is that it allows anyone to extend the description of existing resources, one of the architectural principles of the Web BERNERS-LEE98.

This specification does not attempt to enumerate all the possible forms of vocabulary description that are useful for representing the meaning of RDF classes and properties. Instead, the RDF vocabulary description strategy is to acknowledge that there are many techniques through which the meaning of classes and properties can be described. Richer vocabulary or 'ontology' languages such as DAML+OIL, W3C's language, inference rule languages and other formalisms (for example temporal logics) will each contribute to our ability to capture meaningful generalizations about data in the Web. RDF vocabulary designers can create and deploy Semantic Web applications using the RDF vocabulary description language 1.0 facilities, while exploring richer vocabulary description languages that share this general approach.

The language defined in this specification consists of a collection of RDF resources that can be used to describe properties of other RDF resources (including properties) in application-specific RDF vocabularies. The core vocabulary is defined in a namespace informally called 'rdfs' here. That namespace is identified by the URI-Reference http://www.w3.org/2000/01/rdf-schema# and is associated with the prefix 'rdfs'. This specification also uses the prefix 'rdf' to refer to the RDF namespace

http://www.w3.org/1999/02/22-rdf-syntax-ns#.

For convenience and readability, this specification uses an abbreviated form to represent URI-References. A name of the form prefix:suffix should be interpreted as a URI-Reference consisting of the URI-Reference associated with the prefix concatenated with the suffix. (Mcilraith, 2002; W3C, 2003; www.xml.com, Aug, 15th 2008).

9. QUERYING RDF DATA

The W3C Data Access Working Group has developed the SPARQL Query Language. SPARQL defines queries in terms of graph patterns that are matched against the directed graph representing the RDF data. SPARQL contains capabilities for querying required and optional graph patterns along with their conjunctions and disjunctions. The result of the match can also be used to construct new RDF graphs using separate graph patterns. SPARQL can be used as part of a general programming environment, like Jena, but queries can also be sent as messages to remote SPARQL endpoints using the companion technologies SPARQL Protocol and SPARQL Query Result in XML. Using such SPARQL endpoints, applications can query remote RDF data and even construct new RDF graphs, without any local processing or programming burden. For more questions on SPARQL, see also the separate FAQ on SPARQL.

9.1 SPARQL

SPARQL is a recursive acronym standing for SPARQL Protocol And RDF Query Language. As the name implies, SPARQL is a general term for both a protocol and a query language.

Most uses of the SPARQL acronym refer to the RDF query language. In this usage, SPARQL is a syntactically-SQL-like language for querying RDF graphs via pattern matching. The language's features include basic conjunctive patterns, value filters, optional patterns, and pattern disjunction.

The SPARQL protocol is a method for remote invocation of SPARQL queries. It specifies a simple interface that can be supported via HTTP or SOAP that a client can use to issue SPARQL queries against some endpoint.

Both the SPARQL query language and the SPARQL protocol are products of the W3C's RDF Data Access Working Group.

9.2 Using SPARQL to insert, edit and delete RDF data

The current, standardized version of SPARQL deals only with retrieving selected data from RDF graphs. There is no equivalent of the SQL INSERT, UPDATE, or DELETE statements. Most RDF-based applications handle new, changing, and stale data directly via the APIs provided by specific RDF storage systems. Alternatively, RDF data can exist virtually (i.e. created on-demand in response to a SPARQL query). Also, there are systems which create RDF data from other forms of markup, such as Wiki markup or the Atom Syndication Format.

However, there is significant active work going on to extend SPARQL to support update operations. This feature is certainly one of the facilities frequently asked for in relation to a possible SPARQL "Next" version.

9.3 SPARQL Query Example

```
PREFIX table:  <http://www.test.org /TestTable#>
SELECT ?name ?number
FROM <http:// www.test.org /TestTable.owl>
WHERE
{
  ?element table:name ?name;
          table:phoneNumber ?number;
          table:group table:group_male.
}
ORDER BY ?number
```

Result
This example selects the name and phoneNumbers of all records having gender "male" from given table, the noble gases. The ORDER BY clause indicates that the records should be ordered by their phoneNumbers, in ascending order.

(Clarke, 2000; www.xml.com, Aug, 15th 2009; www.w3.org, Aug, 15th 2008).

10. FUTURE OF SEMANTIC WEB

Once the Semantic Web comes in its mature shape, it can provide the accessibility all content on the Web, describe what each piece of information is about and give semantic meaning to the content item. Therefore making search engines more effective than they are now, and users can find the precise information they are searching for. Organizations that provide various services can tag those services with meaning; using Web-based software agents, you can dynamically find these services on the fly and use them to your benefit or in collaboration with other services.

11. CONCLUSION

Though a lot of progress has been made under the umbrella of Semantic Web, and every research regarding this is focusing on providing an efficient mechanism to share information among different distributed and stand alone applications. But still Semantic Web is in its infancies, and a lot more has to be done down the road. The major research is on the representation of Data in such a way that it can be used throughout the desired scope in a more secure way.

Also the research work is in progress on SPARQL to fetch more meaningful records from semantic data. And another big work has yet to be done for inserting, editing and deleting the existing data in Semantic Web Portals.

12. REFERENCES

IAN Clarke (2000). "Freenet: A Distributed Anonymous Information Storage and retrieval System." Lecture Notes in Computer Science.

Mcilraith, S.A.S.t, (2002). "Adspting Golog for Composition of Semantic Web Services." In Proceedings of the Eighth International Conference on Knowledge Representation and reasoning (KR2002), Toulouse, France, April, (2002).

W3C (2003B). "OWL-S Coalition: OWL-s 1.0 Release."

Semantic Web: Bringing the World Wide Web to Its Full Potential, by Tim Berners-Lee, Dieter Fensel, James A. Hendler

[Abiteboul 1999] Data on the Web: From Relations to Semistructured Data and XML. Abiteboul, Serge / Buneman, Peter / Suciu, Dan Morgan Kaufmann Publishers, 1999, ISBN: 155860622X.

[Hjelm 2001] Creating the Semantic Web with RDF by Johan Hjelm, 2001, Wiley
http://www.ibm.com/developerworks/web/library/wa-semweb/
http://en.wikipedia.org/wiki/Semantic_Web (Aug, 15th 2008)

http://www.w3.org/2001/sw/ (Aug, 15th 2008)

www.xml.com/pub/a/2005/11/16/introducing-sparql-querying-semantic-web-tutorial.html (Aug, 15th 2008)

www.w3.org/TR/rdf-sparql-query Aug, 15th 2008)