Towards a Semantic Web Stack Applicable for Both RDF and Topic Maps: A Survey

F. Alam[1], S. U. Rahman[2], S. Khusro[3], S. Ali[4]

[1,2]*Department of Computer Science & IT, University of Malakand, Pakistan*
[3,4]*Department of Computer Science, University of Peshawar, Pakistan*
fakhrealam@uom.edu.pk

Abstract-Semantic Web extends the World Wide Web by transforming the Web into more machines processable, and intelligent. Semantic Web enables users to share contents beyond the limits of applications and websites. To successfully implement Semantic Web, technologies are developed to effectively represent, navigate, and create metadata relationship among the information. As synthetic sugar, both RDF and Topic Maps have come up as the leading technologies for successfully for realizing the vision of Semantic Web into reality. Semantic Web Stack is a layered model representing architecture of the Semantic Web. The layered model integrates and defines relationships among the technologies and languages essential for the Semantic Web. Semantic Web Stack is developed originally for the RDF exclusively and has no direct support for the Topic Maps. However, each layer of the model is equally comparable and applicable to the Topic Maps paradigm. This paper investigates and analyzes the Semantic Web Stack for determining its applicability for the Topic Maps. We have come up with the conclusion that the stack has potential for accommodating Topic Maps equally but subjected to little more efforts from the research communities.

Keywords-Semantic Web, Resource Description Framework (RDF), Topic Maps, Semantic Web Stack

I. Introduction

Resource Description Framework (RDF) and Topic Maps are the two standards developed by the different standard making organizations for fulfilling the vision of Semantic Web. Semantic Web extends the current web in such a way that the problem of finding precise information at the right time and place will be possible due to its advanced techniques of inferencing, intelligence and machine based searching.

RDF is a W3C standard for representing metadata relationship between web resources. RDF paves the way for software applications to interchange web resources semantically, enhances interoperability between software applications and enables machines for automatic processing of web resources. A RDF model expresses meanings by consisting of statements (triples) where each statement relates web resources by using the analogy of a subject, a predicate, and an object, corresponding to subject, verb, and object of an elementary sentence [i]. Topic Maps is a ISO standard for discovering, linking, filtering and retrieving relevant information on the Web. Topic Maps models web resources in the form of topics, associations between the topics, and occurrences of the topics and associations.

The advent of standards encouraged researchers for contributing extensive research efforts utilizing both of them in parallel for fostering the growth of Semantic Web, resulting into the division of Semantic Web into two separate islands. A plethora of supporting technologies including ontology modelling languages, query languages, interchange formats, and storages are devised for RDF and Topic Maps technologies. The technologies developed for RDF are divided into multiple layers where layer consumes services and offers services to one another. The cumulative efforts resulted into a layered model called Semantic Web Stack, which was first proposed by Tim Berners-Lee (i.e. inventor of the World Wide Web) [ii]. Semantic Web Stack is the general architecture of the Semantic Web. The layered approach provides a number of advantages including dividing the overall process into smaller and simpler components that are easier to develop, facilitate standardization of technologies belonging to individual layers, preventing changes in one layer affecting the other layers, allowing different technologies to work with each other, providing simplicity to enhance understandability and debugging, and accommodating addition of further layers.

Topic Maps, on the other hand, was originally developed for the representation of back of the book index construction [iii]. The original idea of Topic Maps was further extended by the researchers and used it for wider applications like to represent exchange and convey knowledge on the Semantic Web [i]. A Topic Map represents networks of nodes instead of tree hierarchy consisting of topics associations, occurrences, and scope [iv]. To compete with RDF, a comparable technological layered stack is proposed for Topic Maps consisting ontology modelling languages, and query languages providing almost the same functionalities provided by technologies in the Semantic Web Stack. However, RDF related technologies comparatively provides a rich set of features and succeeded in gaining attention of the wider Semantic Web research community as compared to Topic Maps by reaching acceptable levels of maturity. To help Topic Maps to preventing from getting wiped and save the invested efforts, the Semantic Web Stack should extend support for Topic Maps.

This research paper is aimed to provide a comprehensive study by analysing the Semantic Web Stack, covering all of its possible aspects, and presenting its pros and cons. It also attempts to find out how this model can be used for Topic Maps. Main contributions of this paper include:

- The key contribution is the detailed analysis of the Semantic Web Stack and its strength of supporting Topic Maps paradigm.
- The topic is almost unique in its integrity and opens new area of research. No prior work exists in the literature addressing the same problem in a comprehensive manner.
- To organize and classify the available literature about the topic in an attractive manner to catch and boost interest of the new researchers in the area and take them into new avenues of research.
- The paper is expected to provide a compact platform for researchers for finding new research dimensions and discovering solutions for the existing ones.

II. SEMANTIC WEB

The World Wide Web (WWW) has new ways of accessing electronically available information. The WWW, at present, contains billions static web pages, accessed by millions of users around the globe. However, this tremendous quantity of information has given birth to the increasingly difficult problems of finding, accessing, presenting and maintaining the information needed by different users. Furthermore, today's web suffers with information overload problem which can significantly affect its very usefulness [v]. A reason is synthetic nature of the today's web, where information is presented primarily in natural language and computer presents the information only while the interpretation and identification of pertinent information is delegated to users. Thus, a considerable gap has came out between the information available for automated tools aimed at solving the problems stated above and the information maintained in human interpretable form.

According to Oxford dictionary [vi] the word "semantic" is concerned with the meaning of words, phrases, and sentences. Semantic Web distinguished as the next generation of the Web advocates that information will not only aim for human readers but for machine processing as well, which will enable intelligent information services (i.e. intelligent search agents, semantic search-engines, and intelligent information filtering etc.) to provide greater functionality and interoperability as compared to the current isolated services [vii]. Tim Berners-Lee, inventor of the WWW, URIs, HTTP, and HTML toasted the idea of Semantic Web in his historical article "The Semantic Web" and characterized Semantic Web as an extension of the World Wide Web which will enable giving well-defined meanings to information and make the exchange of machine-readable information easy and efficient [viii]. The Semantic Web is a network of information which enables people and computers to work in cooperation through giving well defined meanings to information. Semantic Web provides a platform where data sources using ontologies, semantic rule, web services, and web processes can be integrated. Information are linked up making it easy for machine processing for different purposes such as effective searching, integration, automation, and reuse across various applications [ix]. Thus, Semantic Web enhances machine abilities to solve a well defined problem by performing well-defined operations on existing well defined data.

Semantic Web turns the Web of information (current web) into web of knowledge (future web, knowledge-based web) as shown in Fig. 1 to provide qualitatively new levels of services through unambiguous representation of the underlying data, programs, pages, and any other web resources using semantics. Enabling automated services to understand content on the Web will improve their human assistance capabilities in providing more accurate filtering, categorization, and search of information sources [x-xi]. Therefore, Semantic Web can be thought as an infrastructure for which applications can be developed not an application by itself [ix].

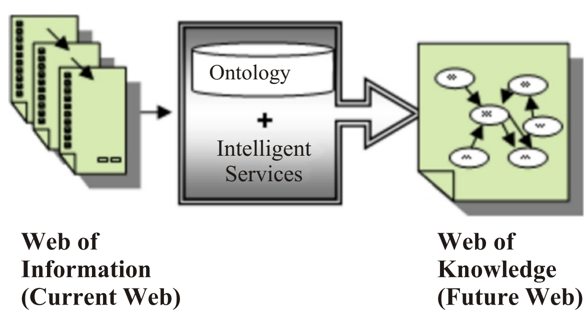

Fig. 1. From the current web to the web of future [viii, xii]

W3C (World Wide Web Consortium) [xiii] assumes to achieve full potential of the Web; it should be turned into a place where information can be shared and processed by automated tools as well as by people. A typical example of Semantic Web application will be automated travel agent which will come up to the user with suitable travel or vacation suggestions under certain restrictions and preferences. To derive suggestions, software agent (automated travel agent) will not only use the already determined sources of information but will search the Web in a similar way as a human user might do when planning a vacation [xiv]. But the problem is that web pages are mainly concerned with presentation to and utilization by human users [xv]. Annotations are used to identify contents of a web page. Annotations are typically in the form of natural language strings or tags which are interpretable and understandable to human beings but not to automated tools (e.g. software agents). To solve this problem and give machine accessible semantics to annotations, Semantic Web uses a number of technologies including ontologies [xiv]. Ontologies are metadata and getting the status of backbone of Semantic Web [xii]. Ontologies provide a generic presentation of domain knowledge and a commonly agreed understanding of a domain to help people and machine communicate efficiently. Therefore, success and proliferation of Semantic Web mainly depends on the cheap and fast construction of web ontologies [xvi].

III. Semantic Web Stack

The basic idea behind the development of Semantic Web Stack by Tim Berners-Lee was to integrate all these technologies and languages into a model through which the realization of the Semantic Web becomes possible. Semantic Web development continues in step wise fashion and each step constructs a layer on top of another. Building a layer on top of another, generally, follows two principles. First is downward compatibility, where an agent fully aware of one layer would be able to interpret and use information written at lower levels. Second is upward partial understandability, where an agent fully aware of one layer should be able to take partial advantage of information at higher levels [xvii].

Generally, Semantic Web Stack model can be divided into three layers as shown in Fig. 2. The bottom layer consists of URIs, Unicode character scheme, XML and XML schema, providing base for the Semantic Web and is already implemented for writing structured web document with user-defined vocabularies. The middle layer is especially for the implementation of Semantic Web core techniques and technologies for developing Semantic Web applications and consists of RDF, RDF Schema, ontology languages, and query languages. The top layer consists of the technologies including logic frame work, trust and proofs etc., which are not standardized but providing enhancement to the lower layers by allowing writing of application specific declarative knowledge, representation of proofs in web language from lower layers and proof validation, and gaining users' trust for its operations and information provided. Users interact with the Semantic Web through applications user interfaces built on top of the top layer. Since the model is developed and presented by the W3C working group, therefore, no information about Topic Maps standard is included in the model. However, the Topic Maps standard and its relevant technologies are comparable to the technologies belonging to each layer of the RDF standard depicted in the Semantic Web Stack [xviii].

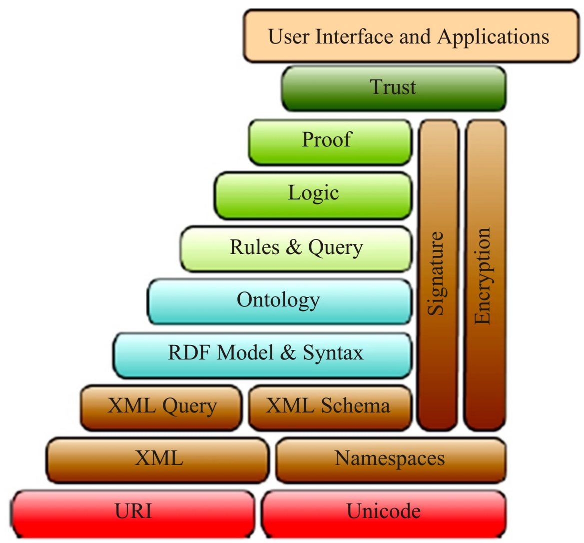

Fig. 2. The Semantic Web Stack

A. URI/IRI and Unicode

Uniform Resource Identifier (URI) or International Resource Identifier (IRI) and Unicode character scheme are the technologies belonging to the bottom layer of the Semantic Web Stack inherited from the Web. A URI is the general form of URL used for accessing a web page in today's web. However, instead of restricting URIs for representing web pages addresses, they have broad spectrum of applicability for identifying any abstract or physical resource on the Web. A URI has the potential of identifying diverse and small object such as email address, mobile number, locations, ISBN, and author of a book. Since RDF establishes a metadata relationship between web resources and resources they can be identified on the Web using the notion of subject, predicate, and object (SPO), therefore, it becomes potentially impossible for URL technology to indentify small and diverse objects and their interrelationships. On the other hand, URIs provides effective support for addressing small and complicated resources and their metadata relationships precisely which goes beyond the scope of URLs.

Along with URI, an encoding scheme called Unicode is used at the same layer. Unicode provides support for representing any type of text in any language in the world uniquely in the computer.

Unicode has the capability to record more than one millions characters and supports approximately one hundred scripts. Unicode system is standardized and used in connection with several new and emerging technologies such as XML, Java, and .NET platform etc. Statement <?xml version="1.0" encoding="UTF-8"?> is included at the top of each RDF document signifying the encoding scheme used. UTF-8 is the mostly used and implemented character encoding scheme, which makes the Semantic Web a universal platform. Unicode in combination with URI extends support for identifying any type of resource in the Semantic Web regardless of its text and scripting language.

B. XML, Namespaces and XML Schema

The layer at the top of URI and Unicode layer accommodate XML, XML Schema, and Namespace technologies for representing low level semantics. In Semantic Web, resources will be divided into pieces and structured in such a way that there will be metadata relationship between resources. HTML provides constructs for formatting web pages and hypertext documents but cannot encode information about resources divided into pieces and structurally related with each other.

1) Extensible Markup Language (XML)

XML is a general purpose markup language for creating special-purpose markup languages, which is simpler to parse and process than SGML. In Semantic Web paradigm, XML will structure the data in such a way that the contents of web resources will be easily accessible to the machine. XML includes features for describing each piece of information, and implementing nesting structures and properties of objects in an easily understandable pattern. The pieces of information can be stored, structured and associated with other web resources using XML [xix]. XML enables users to create XML document with the liberty of defining their own tags according to their choices and needs, storing in plain text file, running on any platform. For example, if someone wants to create a XML document for representing information about name and email address of an object would have contents as follows:

```
<?xml version="1.0"encoding=
"UTF8"?>
<email>
<to>Chief Editor
</to>
<em-address>kareem1@yahoo.com</em-
address>
<from>Alam</from>
<Em-address>fakhrealam@uom.edu.pk
</em-address>
<subject>about my paper</subject>
<body> Reviewers had suggested some
changes which are now accommodated in
the paper</body>
</Email>
```

When a document is created using XML editor, it would contain the definitions of data along with their structural relationships [xix]. The self-made tags created by the users are checked by the XML parser for integrity and validity. An XML document can be easily stored in the form of a text file which can be transferred to any other platform, system, and program. XML documents can be used with and retrieved from different types of databases such as relational, object oriented and XML's own storage servers etc. Representing XML documents in plain text format improve XML's interoperability by running on multiple platforms and programs extensibility by easily integrating new programs on top of the older programs. Other features of XML include language independence, reuse with HTML, and exchangeable structure using DTDs and XML Schema.

2) Namespace

Programmers can define elements in an XML document according to their choices and needs which could arise naming conflict in situations when different XML documents are merged from various applications of the same type [xx]. XML namespaces are devised to resolve such problems, because namespaces uniquely and universally identifies each element and attribute. Using the namespaces feature, a XML documents can be easily and quickly created by merging and reusing code from different XML documents, which can further transferred and reused with other XML documents. A namespace prefix representing the namespace has to be embedded at the start of each element and attribute definition to distinguish it from other similar elements and attributes.

The general syntax of namespace declaration is: *xmlns:prefix="URI"*, where xmlns is the reserved word, prefix is any valid namespace identifier for representing namespace resource designated by the unique URI.. To define a default namespace for a document, the prefix part can be omitted and declaration can be restricted to *xmlns="URI"*. In declaration of namespace statement, *xmlns:xhtml= http://www.w3.org/1999/xhtml* the prefix xhtml means that this document is defined inside the XHTML namespaces and can be mapped into it. The default namespace can also be described for the above statement as *xmlns=http://www.w3.org/1999/xhtml*. The type of statement with no explicit prefix definition will be considered to be in the XHTML namespace. When an attribute is declared with no explicit namespace prefix would mean that the attribute does not belong to any namespace. Therefore it can be concluded that the attributes do not necessarily depend on the default namespace.

XML data model has been proven much better and sophisticated than other data models such as relational and object oriented. Therefore, several types of query languages such as Xquery and XQL have been developed to query its resources [xxi]. The XML query languages provide simple and stylish interface to users due to which they can choose query style of their own choice. XQuery is the powerful XML based query language developed for Semantic Web Stack, which can easily relate web resources, documents, and databases of the Semantic Web. XQuery provides the potential for reading XML files, selecting a particular value, arranging data in an order, and returning final results in another XML document format.

3) XML Schema

XML schema is the extension of XML DTD with tremendous advancements in features and functionalities including support for data types and namespaces, and extensible to future additions [xxii]. XML Schema describes information about the basic structure and internal format of a XML document. XML Schema language, also called XML Schema Definition (XSD), provides potentials for defining and expressing elements and attributes, childs of elements and attributes along with number and order, data types of elements and attributes as well as their fixed values etc. in a XML document. The validity of a XML document can be verified by computer programs using XML Schema documents. The programmer can take help from the schema document to create accurate and valid XML documents which can be further used by computer programs.

C. RDF and RDF Schema

XML has the expressive power to express and encode any type of resource if proper grammar is defined for the resource [xiv]. Similarly, XML has strong parsing capability to parse any type of data using its parser libraries to determine its usability and validity for other applications due to its support for syntactic interoperability. However, XML fails in mapping an unknown data with a known data and establishing a semantic relationship between them. To overcome the limitations of XML and increase supporting and sharing semantic/metadata between heterogeneous web resources in Semantic Web, RDF technology is developed for creating data models on top of the XML data models.. RDF technology produces data models, implementing and establishing semantically enhanced metadata relationships between different types of web resources. RDF model increases applications' interoperability by exchanging previously machine dependent web resources between applications and automatic processing of web resources.

1) RDF

RDF is a W3C recommendation providing a standard for metadata for describing data about web resources. Although RDF provides much better and rich syntax and semantics as compared to the XML, but still consumes the services from XML by working in collaboration with XML. The RDF framework has sophisticated features of expressive power and syntactic interoperability over XML, but the feature distinguishing RDF from XML is its semantic interoperability [vii]. The main reason of this property of RDF model is due to its independent object-attribute structure, no need of objects translation, defining relationships between similar domains, mapping two different RDF descriptions, and the availability of knowledge representation techniques. These features empower RDF to interchange data at much higher level as compared to XML parser. A major reason of RDF propagation is its inherent flexibility to represent the full spectrum from highly structured data similar to a relational database to unstructured data as it may be found in social networks [xxiii]. However, to fulfill Semantic Web needs requires availability of universally shared knowledge representation language. Although the need is not practically satisfied as yet with a single technology but using technologies such as RDF, its schema and ontology languages in combination can provide considerable solution.

Recent years have witnessed a tremendous increase in the publically available RDF datasets semantically related and interoperable with each other, whose example is LOD (Linked Open Data) cloud a remarkable collection of interlinked RDF datasets [xxiv]. To achieve semantically interoperability, RDF model is serialized into XML syntax called RDF/XML. RDF/XML syntax can merge and integrate distributed and heterogeneous resources on the Semantic Web. In addition to RDF/XML, RDF model can be represented in other interchange formats as well such as Notation-3 (N3) which is more simple and easy to read and write, and provides logic and inference mechanism as compared to RDF/XML.

2) RDF Schema

RDF describes relationship between web resources using subject, predicate and object notion, however, web resources may also contain vocabularies which could belong to other resources [xi]. RDF framework provides constructs from its own vocabulary for mapping relationship between web resources using, and fails to map relationships if web resources are semantically related by using constructs from other resources. To describe such type of semantic relationships between vocabularies, a schema language for RDF called RDF Schema is used which can itself be expressed as a RDF model. The RDF Schema language works above RDF in the Semantic Web Stack and extends the original RDF model with some special semantic mechanisms to add numerous constructs for defining classes of resources and the properties specific to those resources. In other words, RDF Schema language extends the expressiveness of

RDF frameworks by providing a set of novel constructs [xxiv].

In database paradigm, schema language only works to impose restrictions on DBMS and define the internal storage structure for databases such as tables, fields and relationship structure and size. RDF schema, on the other hand, provides constructs for defining concepts, concepts' properties, values of the properties, and the relationship between all these concepts, properties and values. With the help of RDF schema, users can not only determine what the data is about but also all others related information to this data. While describing all vocabularies, RDF schema divides vocabularies into the form of classes, the property belongs to particular classes, sub classes and into the instances of classes. RDF schema also imposes constraints on RDF and its properties that are expressed in a particular domain.

D. Ontology

The widespread increases in the size of web contents have created serious limitations in the hypertext system. The finding, sharing, interpreting and integrating of the required information from multiple sources are difficult with using the technologies provided by the current web such as keyword based search etc. However, some techniques have been developed in the current web, also called Web 2.0, for solving these problems, but they are able to solve specific problems such as information integration and sharing according to specific situations and users. Furthermore the general problem of finding and integrating information by the machines automatically according to users' demands and preferences cannot be solved by these techniques. Semantic Web aim to solve the problems by sharing information among wider communities and processing the information automatically using the Semantic Web tools [xxv].

The expressive power of RDF and RDF schema also fails while dealing with complex type of resources and their semantic relationships [xii]. Ontologies are deemed as supporting technology for Semantic Web to help in solving the problem of semantic annotations between different types of web contents. Gruber formerly defined the notion of ontology in 1993 as "explicit specification of conceptualization" [xxvi]. Typical web ontology consists of taxonomy and a set of inference rules. The taxonomy defines hierarchy of classes of objects and relationships among them. A large number of relationships among objects of the classes can be described by assigning properties to classes which can be inherited by the subclasses. Using ontologies, effective reasoning can be possible, better syntax can be written, precise meaning will be assigned to the knowledge, and the appropriate expressions can be possible quickly.

The origin of ontology goes back to the philosophy and can be defined in a different ways because of their potential applications in many fields. In the Semantic Web, ontology gives formal meanings to the web contents which are further interpreted and transferred into semantic annotation. The success of Semantic Web depends on the existence of multiple distributed ontologies enabling users to annotate their data for improving shared machine readable content [xi]. Knowledge which consists of multiple concepts in a specific domain can formally be represented and related with the help of ontology. With the use of ontology, different types of shared vocabularies and taxonomies, from which a specific domain is created and which consist of multiple concepts/objects along with their properties and relations with other concepts and objects, can be easily retrieved [xxvii]. Ontology can be used in almost each field of computer science for the proper organization of information. For the proper organization of information in the Semantic Web, an ontology language called Web Ontology Language (OWL) is developed.

1) Web Ontology Language (OWL)

RDF and RDF Schema describes vocabularies by using constructs for supertype and subtype relationships, classes and their instances, and superimposing restrictions for domains and ranges of properties [xvii, xxviii]. However, the power of RDF Schema becomes unsuccessful in implementing special types of features and restrictions such as range restrictions for classes, disjointness of classes, defining compound class properties using boolean and set theory (union, intersection, complement), and describing cardinality restrictions. To the problems unbearable for RDF and RDF Schema, an advance ontological language is developed by the Semantic Web community called OWL, which works one level above the RDF/RDFs in the Semantic Web Stack.

The Web Ontology Language (OWL) is a W3C standard used to create ontologies for the World Wide Web. OWL (i.e. successor of the web ontology language DAML + OIL) is a description logic (i.e. SHOIN(D)) based ontology language with an RDF/XML syntax [xxviii]. OWL includes constructs for describing much richer semantics, integrations, and interoperability between web resources as compared to RDF Schema. OWL can be effectively used in computer programs because of its machine oriented methods for checking the validity and consistency of knowledge and making implicit knowledge explicit. The documents created using OWL are called ontologies, which can be published and shared on the World Wide Web. The OWL overcomes the weakness of RDF Schema by adding more vocabularies for describing classes and properties including disjointness, equality, symmetric, and transitive etc. as well as restriction on classes and their properties. The syntax of OWL is based on XML and has three sub-languages: OWL Lite, OWL DL, and OWL Full. Every Lite ontology is also a DL ontology and every DL ontology is also a Full ontology [xxix].

OWL Lite is easier to learn and implement as compared to Full and DL, and is especially designed for situations where users' requirements are limited to implementing simple constraints features and a classification hierarchy [xxx]. OWL Lite is an extension of RDF and includes features for more specific or intersection based class definition, creating class individuals, describing equality or difference between class, characterizing inter-class relationships, and characterizing properties as inverse and transitive etc . OWL Lite is effective if used for simple domains such as if users' want to implement cardinality constraints of only 1 and 0 values. It is very simple to provide tool support for OWL Lite for enabling quick migration of thesauri and other taxonomies [xxvi]. However, the available tools for supporting OWL Lite are complex and can be used with OWL DL as well.

OWL DL provides all of the OWL language constructs for helping users to gain maximum expressiveness without losing computability. DL stands for description logic, which represents a decidable fragment of first order logic and included in OWL DL for obtaining required computational properties and reasoning capabilities. OWL DL is mature enough for providing foundation to the OWL for creating knowledge based ontologies [xxvi, xxviii]. OWL DL implements several types of restrictions on OWL and RDFs syntax to obtain processing efficiency for reasoning. However, due to the rapid change and restriction imposed on RDF document before converting it into OWL DL documents, full compatibility has been lost between OWL DL and RDF.

OWL Full is fully compatible with RDF Schema and can represent resources on the Web both syntactically and semantically as RDF and RDF schema [xxvi, xv]. OWL Full helps users who want maximum expressiveness and syntactic freedom of RDF with no computational guarantee. In OWL full, constraints regarding classes, subclasses, properties, individuals can be represented in ontology more efficiently and expressively such as a class can depict simultaneously a collection of individuals and a single individual in its own right. OWL Full borrows some of the semantics from the other species of OWL (i.e. OWL Lite and OWL DL). With the help of OWL full, one can build an ontology which can extend the meaning of already built RDF and OWL vocabulary.

E. *Rules and Query*

Rules and query languages are defined in layer at top of the RDF and Ontology layer in the Semantic Web Stack. Rules languages add more inference mechanism web ontology languages and query languages are used to retrieve data from the RDF model. Rule languages address the logic problems related to description and inference mechanisms which cannot be satisfied by the existing RDF and OWL technologies [xxxi]. The rule language used in the Semantic Web Stack is called Semantic Web Rules Language (SWRL) and the query language used is called SPARQL.

Semantic Web Rule Language (SWRL), standardized by W3C in 2004, is result of the combination of OWL DL and OWL Lite sublanguages of the OWL with the Unary/Binary Datalog RuleML sublanguage of the Rule Markup Language [xxxi]. Rule langue is need for several reasons including reusing of the existing rule sets, improving OWL expressivity, and providing ease in reading and writing rules. SWRL has high level abstract syntax for Horn-like rules and expresses all of the rules in terms of OWL concepts, properties, and individuals. The main features of SWRL include its strong support for several popular tools such as SWRL Tab, KAON2 and Pellet.

Query languages help Semantic Web Stack for enabling users to retrieve data from RDF model. SPARQL is the Semantic Web query language, which can be used for retrieving data from RDF graphs stored in triple format. SPAQRL was first standardized in 2008 by W3C and its extension SPARQL 1.1 is also standardized recently [xxxii]. Data in a RDF model is stored in triple format which reflects a certain graph pattern. , data is stored in the form of triple format which is based on certain graph patterns. Graph patterns in RDF models can be queried using SPARQL queries for retrieving results to the users if triple patterns in the models matches with the queries. Information retrieved by a SPARQL query could be in the form of URIs, blank nodes, and plain and typed literals. A reason of SPARQL inclusion in Semantic Web Stack could be the ability to transfer SPARQL query to another query format such as RDBMS query language (i.e. SQL) or XML query language (i.e. XQuery) etc.

F. *Logic, Proof and Trust*

In Semantic Web Stack, there are also some unrealized technologies such as Logic, Proof and Trust, working above on the standardized technologies such as XML, RDF and OWL. The basic vision of the Semantic Web highly requires logic and knowledge to be essential parts of the Semantic Web Stack. Logic provides detailed explanation for query answering, shows precise knowledge, and delivers easy to understand formal semantics. The OWL sublanguages OWL Lite and OWL DL come with first order logic and descriptive logic respectively, having inference capability to deduce complex knowledge from the ontologies. However, a powerful logical language is necessary for the Semantic Web to enhance reasoning and inferencing capabilities with the aid of logic either from a single ontology or ontologies in combination.

Proof layer in the Semantic Web Stack is included by Tim Berners-Lee to access cognitive and meta information [xxxiii]. The basic reason would be when a client submits a request to the server for a particular resource along with proofs, the server will reply to the client based on proof. Generally, meta information about a resource are considered as a proof of its contents and the proof of one resource can also become the proof of another resource when the information from one resource is incorporated with another one. Proof can also be used for the presentation of graphical data and natural languages, due to which humans can easily deduce answers and solve problems by using proof as a template. Due to these reasons, the importance of proof becomes a practical reality in the Semantic Web.

In Semantic Web Stack, proof is enhanced with Trust because only proofs cannot provide much confidence for people to publish their data on the Semantic Web [xxxiv]. The proof depends on statements due to which a true assumption is not possible for the user and in some situations, at a particular time; it becomes impossible for the people to understand it. With the help of Trust, RDF data in the Semantic Web will be secured more by digital signature techniques from their respective web authors. This type of trust will work globally and the users and Semantic Web agents will reach to more precise RDF statements with full confidence.

G. User Interface and Applications

The top layer of Semantic Web Stack is the user interface and applications through which people will interact with RDF model though applications [xxxv]. To embed the semantic structure in the current web, there is need for applications which integrate data and improve the search mechanism to a more specialized and intelligent level. Applications generally show the characteristics of RDF data in the Semantic Web. With the help of these applications, distributed and heterogeneous information resources can be accessed visually. Applications based on RDF can also store and organize knowledge in a better way and manage large repositories in an efficient way.

The applications can provide adaptive and customized views by analyzing users' current task/activity and can generate response to the users according to the context identified. Specialized browsers based on these applications are used for visualization and navigation within specific domains due to which users can easily and quickly explore inside the domain and build-up their own thinking of conceptual associations and problem solving paths. In short, due to these applications, the RDF and Topic Maps based information can be efficiently manipulated, the structure of knowledge/resources can be maintained precisely, and both knowledge and information layer can be easily searched, navigated and visualized.

IV. Semantic Web Stack for Topic Maps

Semantic Web Stack is developed by the W3C RDF data models exclusively and containing no information about handling Topic Maps. However, Topic Maps is a parallel technology to RDF, performing almost the same functions and in the similar way as RDF do. Like RDF, several techniques and technologies are developed for Topic Maps to help in realizing the vision of Semantic Web. Each of the technology developed for Topic Maps is equivalent to corresponding technologies in the Semantic Web Stack. Fig. 3 shows layer wise comparison of the technologies for both of the standards. Table I presents layer wise comparison of the technologies capabilities for both of the standards.

A. Serialization/Interchange Formats

Serialization is the process in which data in one format is semantically converted into another format for the storage and transmission purpose. It is due to the serialization that different types of data formats can be stored and run on different hardware platform regardless of their underlying architecture, thus, promoting interoperability. XML provides persistent method for the interchange of data on the web to be stored or communicated regardless of the programming languages in a human readable format [i]. A number of serialization formats are defined for mapping RDF models including RDF/XML and N3 etc., whereas, the serialization formats developed for Topic Maps to function at lower layer of the Semantic Web Stack are XTM and LTM. The Topic Maps serialization formats can work on the same layer of the Semantic Web Stack where RDF serialization formats functions. The serialization formats provided for both of technologies can be either XML based or non-XML based [i].

XTM is a XML based serialization format for Topic Maps, which utilizes XML for storage and URL for relating and referencing. XTM is more simple, flexible, and easy to interpret as compared to other format and improves knowledge accessibility across applications. XTM is inherently implemented in web browsers, making knowledge representation and navigation on the Web considerably easy LTM is non-XML based serialization format for Topic Maps which can work on Semantic Web Stack similarly XTM. LTM enables creating Topic Maps documents in simple textual format in any text editor, which could be either processed by Topic Maps software or converted into XML format for further processings. However, LTM representation of Topic Maps information is simple, efficient, and takes less space as compared to XTM. LTM representations of a Topic Map data model can be easily converted into XTM format but have lower expressive power than XTM. Furthermore, LTM interchange format is suitable for representing small Topic Maps such as email, presentations, discussions and personal use.

B. Data Modeling and Ontology

Topic Maps data model can work on the same layer on which RDF model works in the Semantic Web Stack. A Topic Maps data model describes semantic relationship between web resources in the similar way RDF technologies perform. Like its counterpart, the Topic Maps data modeling and ontology technologies will enable machine dependent web resources to be interchanged and processed automatically by applications to work in interoperable way.

In Semantic Web, RDF Schema and OWL technologies provide much richer semantics and constraints for RDF data models. Topic Maps Constraint Language (TMCL) standardized by the ISO no.19756 defines schema for Topic Maps [xxxvi, xxxvii]. For Topic Maps, TMCL technology provides increased semantic interoperability and constraints for Semantic Web. In relational database, the schema is used to show the relation between tables, field size, arrange data in a graphical structure, and to impose some types of constraints on data. In Topic Maps paradigm, schema language is also used to implement some types of constraints on information. These constraints include simple and best user interface to properly document the structure of Topic Maps and to verify the information in a consistent and meaningful way. The mechanism used to implement constraints in TMCL is so simple, precise and clear that the job can be automatically validated in machine readable form. These validations can be performed by a mechanism called semantic validate, which verify every document and report errors when the condition is not according to the context and will remain silent when the constraints are fulfilled. Several types of constraint languages are available to implement constraints on Topic Maps. Among them XTche language, AsTMa! Language, and OSL language are more popular.

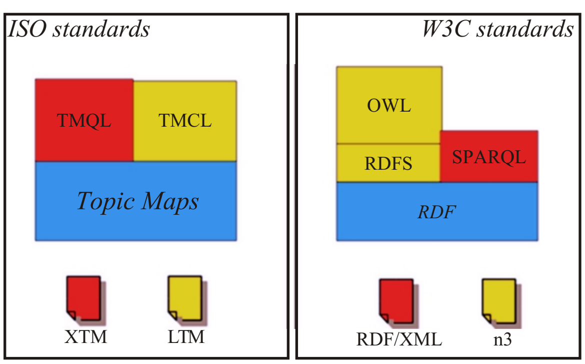

Fig. 3. Layer wise technologies comparison of RDF and Topic Maps

TABLE I
LAYER WISE TECHNOLOGIES CAPABILITIES FOR RDF AND TOPIC MAPS

Layers	Characteristics		RDF	TMs
Serialization/ Interchange Layer	XML Based		✓	✓
	Non-XML Based		✓	✓
Data Modelling and Ontology Layer	Data Modelling Language	Statements Declaration	✓	✓
	Schema Language	Schema Definition	✓	✓
	Ontology Development Language	Restrictions	✓	✓
		Concepts	✓	✓
		Properties	✓	✓
		Occurrences	✓	✓
Query Language Layer	Implementation		✓	✓
	Accuracy		✓	✓
	Scalability		✓	✓
	SQL Resemblance		✓	✓

C. Query Language

Topic Maps Query Language (TMQL) can be implemented on the Semantic Web Stack to retrieve data from Topic Maps data model (TMDL), in the similar way as SPARQL retrieves from a RDF data model and OWL ontology. This query language was standardized by the ISO and is easy to learn and implement on the Semantic Web paradigm due to its similarity with SQL and XML. Semantic Web information, regardless of its huge size and rapidly changing nature, can be accessed with the help of TMQL.

D. Concluding Remarks

The Semantic Web Stack is a model developed by W3C for accommodating technologies developed exclusively for semantically annotating RDF data models only and there is no such model developed for Topic Maps until now. However, the technologies developed for Topic Maps data modeling can be exactly layered up as technologies prescribed for RDF data models. Like RDF technology, the Topic Maps data modeling and ontology development technologies builds a layer over the serialization/interchange formats, and query languages builds a layer over the data modeling and ontology development technologies layer. The performance, expressibility, accuracy, reliability, and usability of the technologies developed for both of the paradigm are almost the same. Therefore, instead of creating a separate Semantic Web Stack for Topic Maps, the existing W3C Stack should be extended with introduction of new set of protocols and logic capabilities applicable equally to RDF and Topic Maps paradigms.

V. CONCLUSION

Semantic Web vision is a response to the limitation of the current web which aims to make web content machine processable by adding metadata annotations using the technologies of RDF and Topic Maps. The basic idea behind the development of RDF and Topic Maps was to extend and present the current web as a useful and flexible medium where users can easily dig out information as per their requirements. RDF and Topic Maps make information machine processable by enriching them with semantics and reasoning capabilities. The Semantic Web Stack developed by Tim Berners-Lee is a model which shows the basic architecture of Semantic Web. The model consists of multiple layers and the basic purpose is to integrate all the technologies and languages into a single model for the realization of the Semantic Web. Semantic Web Stack was developed for RDF only and has no direct support for Topic Maps. However, each of its layers can be comparable and applicable to the Topic Maps paradigm. Semantic Web Stack is a model developed by W3C keeping in view RDF and its succeeding OWL, whereas, Topic Maps are least considered for complying with the standard requirements defined in the model.

In this paper, we presented a comprehensive study of RDF Semantic Web Stack covering each layer of this model and compared it with Topic Maps tools and techniques. The on-hand knowledge about the available techniques in the Semantic Web Stack for RDF and their equivalent in Topic Maps is presented wisely to elaborate their capabilities and are compared and analyzed in a format to give insight knowledge to the people to help them in selecting one suitable for their needs. During investigation, it has been found that Semantic Web Stack is rich enough and fruitful to fulfill the varied needs of users in Topic Maps as well, but requiring certain more efforts. Therefore, instead of developing a separate model for Topic Maps, the existing Semantic Web Stack model has the potential to accommodate Topic Maps technology.

REFERENCES

[i] F. Alam, M. A. Khan, S. Ali, and S. Khusro. "The Jigsaw of Resource Description Framework (RDF) and Topic Maps Serialization Formats: A Survey," *Pakistan Academy of Sciences,* vol. 51, p. 101-114, 2014.

[ii] A. Gerber, A. Merwe, and A. Barnard. "A functional semantic web architecture," in *Proceedings of the 5th European semantic web conference on The semantic web: research and applications,* Tenerife, Canary Islands, Spain, 2008.

[iii] S. Pepper, "Topic Maps," *Encyclopedia of Library and Information Sciences, Third Edition,* 2010

[iv] R. Kannan, "Topic Map: An ontology framework for information retrieval," *arXiv preprint arXiv:1003.3530,* 2010.

[v] F. Alam, M. A. Khan, S. Rahman, S. Khusro and S. Ali. "Resource Description Framework and Topic Maps: Complementary or Competitive?" Proceedings of the Pakistan Academy of Sciences 52 (1), 1-14, 2015.

[vi] Oxford dictionaries Language Matters. (12-10-2014). Available: http://www.oxforddictionaries.com/

[vii] S. Decker, S. Melnik, F. Harmelen, D. Fensel, M. Klein, J. Broekstra, M. Erdmann, and I. Horrocks. "The Semantic Web: the roles of XML and RDF," *Internet Computing, IEEE,* vol. 4, pp. 63-73, 2000.

[viii] T. Berners-Lee, J. Hendler, and O. Lassila. "The Semantic Web," *Scientific American,* vol. 284, pp. 34-43, 2001.

[ix] K. Vanitha, K. Yasudha, M. S. Venkatesh, K. Ravindra, S. V. Lakshmi, and K. N. Soujanya. "The Development Process of the Semantic Web and Web Ontology," *(IJACSA) International Journal of Advanced Computer Science and Applications,* vol. 2, 2011.

[x] Y. Ding, D. Fensel, M. Klein, and B. Omelayenko. "The semantic web: yet another hip?," *Data Knowl. Eng.,* vol. 41, pp. 205-227, 2002.

[xi] D. Fensel, C. Bussler, Y. Ding, V. Kartseva, M. Klein, M. Korotkiy, B. Omelayenko, and R. Siebes. "Semantic web application areas," in *Proceedings of the NLDB Workshop*, 2002.

[xii] B. Kapoor and S. Sharma, "A Comparative Study of Ontology building Tools in Semantic Web Applications," *International Journal of Web & Semantic Technology,* vol. 1, pp. 1-13, 2010.

[xiii] World Wide Web Consortium. (17-10-2014). Available: http://www.w3.org/

[xiv] I. Horrocks, "Ontologies and the semantic web," *Commun. ACM,* vol. 51, pp. 58-67, 2008.

[xv] F. Alam, S. Rahman, S. Khusro and S. Ali: A Road Map for Killer Applications in Resource Description Framework (RDF) And Topic Maps. Sci.Int.(Lahore), 27(1), 185-190, 2014.

[xvi] A. Maedche and S. Staab, "Ontology Learning for the Semantic Web," *IEEE Intelligent Systems,* vol. 16, pp. 72-79, 2001.

[xvii] G. Antoniou and F. Van Harmelet, "A semantic web premier," *England: The MIT Press Cambridge,* 2004.

[xviii] I. Horrocks, B. Parsia, P. Patel-Schneider, and J. Hendler. "Semantic Web Architecture: Stack or Two Towers?," in *Proceddings of the Principles and Practice of Semantic Web Reasoning*

(PPSWR) 2005, pp. 37-41, 2005.

[xix] Nazmul. (1999). *Benefits of using XML*.[Online] Available:http://developerlife.com/tutorials/?p=31 [Access Date: 28/10/2014]

[xx] E. V. D. Vlist, "Relax NG: A Simpler Schema Language for XML.", *Beijing: O'Reilly,* 2004.

[xxi] F. Alam, N. *Rashid, M. Salam, M. R. Khan. Towards a Universal Query Language for Resource Description Framework (RDF) and Topic Maps. International Journal of Science and Technology Volume 3 No. 6, June, 2014.*

[xxii] W3School. Introduction to XML Schema. [Online] Available: http://www.w3schools.com/schema/schema_intro.asp [Access Date:22/08/2013]

[xxiii] A. Schätzle, A. Neu, G. Lausen, and M. Przyjaciel-Zablocki. "Large-scale bisimulation of RDF graphs," in *Proceedings of the Fifth Workshop on Semantic Web Information Management*, pp. 1-8, 2013.

[xxiv] R. Studer, S. Grimm, and A. Abecker. "Semantic web services: concepts, technologies, and applications," *Springer,* 2007.

[xxv] F. Alam, S. Ali, M. A. Khan, S. Khusro, A. Rauf, "A Comparative Study of RDF and Topic Maps Development Tools and APIs", BUJICT Journal, Volume 7, Issue 1, December 2014, pp. 1-12.

[xxvi] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquis.,* vol. 5, pp. 199-220, 1993.

[xxvii] D. L. McGuinness and F. v. Harmelen. (2004). OWL Web Ontology Language. [Online] Available: http://www.w3.org/TR/owl-features/ [Access Date: 05/09/2013]

[xxviii] G. Antoniou and F. V. Harmelen, "A Semantic Web Primer, 2nd Edition (Cooperative Information Systems)," *The MIT Press*, 2008.

[xxix] N. Matentzoglu, S. Bail, and B. Parsia. "A Corpus of OWL DL Ontologies," in *Description Logics*, pp. 829-841, 2013.

[xxx] T. D. Wang, B. Parsia, and J. Hendler. "A survey of the web ontology landscape," *Springer,* 2006.

[xxxi] G. Antoniou and F. V. Harmelen, "Web ontology language: Owl," in *Handbook on ontologies*, *Springer,* pp. 67-92, 2004.

[xxxii] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. "SWRL: A semantic web rule language combining OWL and RuleML," *W3C Member submission,* vol. 21, p. 79, 2004.

[xxxiii] C. Buil-Aranda, A. Hogan, J. Umbrich, and P.-Y.Vandenbussche. "SPARQL Web-Querying Infrastructure: Ready for Action?," in *The Semantic WebISWC 2013*, *Springer,* pp. 277-293, 2013.

[xxxiv] C. Wernhard and A. Persist, "Representing Proofs in the Semantic Web Draf," 2001.

[xxxv] T. Welsh. (2003). The semantic Web: Proof, Trust, and Security. [Online] Available: http://www.cutter.com/research/2003/edge030923.html [Access Date: 16/09/2013]

[xxxvi] K. S. Candan, H. Liu, and R. Suvarna. "Resource description framework: metadata and its applications," *SIGKDD Explor. Newsl.,* vol. 3, pp. 6-19, 2001.

[xxxvii] G. R. Librelotto, R. P. d. Azevedo, J. C. Ramalho, and P. R. Henriques. "Topic maps constraint languages: understanding and comparing," *International Journal of Reasoning-based Intelligent Systems,* vol. 1, pp. 173-181, 2009.