REVIEW:
Semantic Web; Ontology Specific Languages for Web Application Development

Zeshan Ahmed, Thomas Dandekar, Saman Majeed
Department of Bioinformatics
Biocenter, University of Wuerzburg
Am Hubland 97074, Wuerzburg, Germany

ABSTRACT: *This paper is based on the findings of a literature review on the field of World Wide Web. A list of some key publications and explicitly provided literature is briefly included in this paper; addressing the importance of the field of Semantic Web and describing its contributions as the mechanism for structuring the information over the web in a format so that machines can understand the semantic context. Highlighting the technological innovation of Semantic Web, this paper presents Ontology some domain specific languages for Ontology construction: eXtensible Mark-up Language, Resource Description Framework and Web Ontology Language; offering different ways of explicitly structuring and richly annotating Web pages. Furthermore this paper discusses how Ontology is contributing to the semantic based web system development with some example of real time applications and concludes with some existing limitations needing to be overcome.*

Keywords: Ontology, Semantic Web, World Wide Web

Received: 11 October 2011, Revised 3 December 2011, Accepted 8 December 2011

1. Introduction

World Wide Web is a global information sharing and communication system made up of three standards: Uniform Resource Identifier (URL), Hypertext Transfer Protocol (HTTP) and Hypertext Mark-up Language (HTML) developed by Tim Berners-Lee to effectively store, communicate and share different forms of information [1]. The information is provided over the web in text, image, audio and video formats using HTML, considered unconventional in defining and formalizing the meaning of the context. Currently most of the data over the web (or attached to web applications) is not well structured; it is easy to go for scattered extensive information by looking into bookmarked web pages but quite difficult to extract a piece of needed (particular) information. HTML documents are formatted such that these cannot be processed semantically because these are only available in unstructured readable format. This deficiency leads to the problems of *intelligently searching, extracting, maintaining, uncovering and viewing the knowledge based information over the web*. Moreover deficiency becomes the major cause of some semantic oriented problems e.g. Meta data extraction. There is a current need to have an approach which can publish data over the web not only readable but also in machine understandable and processable formats.

Most of the search engines are promising enough as they require excessive manual pre-processing e.g. designing a schema, cleaning raw data, manually classifying documents into taxonomy and manual post processing e.g. browsing through large result lists with too many irrelevant items [11]. Furthermore some search engines and screen scrapers are also there but are insufficient in creating a rich multi domain information environment [2]. Such search engines use full text query to search information but can only return unstructured contents not the actual structured information stored in the attached database whereas screen scraper extracts and reassembles fragments from the web pages.

To increase data integration and interoperability over the web the concept of "*Web Service*" was introduced [28, 51, 52]. Initially due to their dynamic nature the web services became very popular in industry in very little time but later on an enormous increase in the number of web services with end-to-end service authentication, authorization, data integrity and confidentiality problems were identified which are still alive and not handled by existing web technologies [53].

If the data will be structured over the web only then it will improve the process of search, extraction and maintenance of particular information over the web. As an advanced version of Web, Web 2 was introduced [1] to improve interactive information sharing, interoperability and user centred design web application development. Later on Web 2 was also updated to a new concept Web 3. The concept of Web 3 is to transform Web (online data) into a database to provide accessibility of the contents by multiple non browser applications [19]. Continuing the streak of advancement in existing web and to cope with the currently existing web problems: *Information filtration, security, confidentiality and augmentation of meaningful contents in mark-up presentation*, the concept of "*Semantic Web*" (SW) was proposed by Tim Berners Lee [3] (renowned as the modified version of Web3).

SW is a mechanism for presenting information over the web in such a form so that humans as well as machines can understand the semantic of the context. It is a linked mesh of information which could be processed [5]. The aim of SW is to produce technologies and domain specific languages capable of reasoning on semi structured information [4]. SW is an intelligent conception and advancement in World Wide Web to collect, manipulate and annotate information independently by providing effective access to the information. It provides categorization and uniform access to the resources and advances the transformation of World Wide Web into semantically modelled knowledge representation systems with a common framework which allows data to be shared and reused [7]. It also gives the concept of semantic based web services for dynamic composition of service based applications. SW research depends on a number of key methodologies: *Knowledge Representation Languages* and *Reasoning Algorithms* [32]. Currently SW is standing on a very important building block of Ontology [6], aims of structuring data into processable semantic models [8], as the collection of interrelated semantic oriented concepts (see section 2).

One of the most recent developments is the integration of SW concepts in agent and multi agent based (communication) system development. Some agent based SW systems have been developed using SW technologies. A new SW specific language i.e. *Meta-language of the agent* (AgentML), is also introduced to formulate the agent by discussing its (agent) components [54]. Agents together with SW can provide many practical online benefits e.g. web indexing agents can turn documents into formal knowledge, personal agents can be used for reservations (e.g. holiday's trips. doctor appointments etc.), multi agent system can be used to build and maintain additional Linked data sets etc [55].

The ultimate goal of semantic web is to structure the meaningful contents of unstructured published data over web to take advantage in improving the data extraction processes [3] and to involve knowledge management in creating an advanced knowledge modeled management systems. Without a doubt SW has contributed in the progress of web but still there are some limitations and due to them SW is not currently successful in attaining the actual goal of completely structuring the information over the web making advanced knowledge modelled system. The need is to enhance the existing semantic web technologies and proposition of new domain specific languages for better SW application development because all the theories can be fruitful if the implementation is possible.

The remainder of this review paper is organized as follows: targeting the challenges of implementing a SW application capable of providing semantic based search to extract desired information from attached repositories over the web; ontology is explored and discussed in section 2. Section 3 presents some ontology (domain) specific languages and section 4 describes some example of SW applications. Section 5 provides some limitations and section 6 conclusions.

2. Ontology

Ontology is the explicit representation and description of already available finite sets of terms and concepts used to make the abstract model of a particular domain. Ontology has become a favorite subject for different research communities e.g. computer science, philosophy, bioinformatics, knowledge engineering & management, natural language processing, information retrieval, cooperative information systems and information incorporation etc., because of its interdisciplinary nature [9]. With variability in its usage, ontology has different definitions with respect to the different fields e.g. in computer science it is defined as the combination of concepts and relationships for domain modelling [37], in philosophy it is known as the study of "what there is" [38], or a mathematical formulation of properties and relationships of certain entities [39], and so on.

Ontology is known as the major building block of SW to structure data in machine processable semantic based models for knowledge base systems implementation. Ontology has two kinds of representation schemes i.e. *informal* and *formal* [40]. Informal representation method provides broad range of entities and relations (object, attribute, value triples) whereas formal representation method is based on family of description logics [41]. Ontology produces the abstract modeled representation of already defined finite sets of terms and concepts involved in intelligent information integration and knowledge management [9]. It is basically categorized in three different categories: *Natural Language Ontology* (NLO), *Domain Ontology* (DO) and *Ontology Instance* (OI). NLP provides the relationships between generated lexical tokens of statements based on natural language [25]. DO models precise domains and OI is to generate automatic object based web pages [9].

Ontologies are constructed and connected to each other in a decentralized manner to clearly express semantic contents and arrange semantic boundaries to find out required needed information [10]. It is mainly the combination of following elements: *classes, properties, values, relations between classes, restrictions on properties* and *characteristics of slots* e.g. during the development of natural language search engines (in most of the cases) natural language based information (e.g. queries, grammar rules, vocabulary etc.) are treated as the input to the ontology construction process, which first parses the text in nouns and verbs. Nouns are represented as "*classes*" and verbs as "*properties*" containing values, relationships with other properties and some constraints. Classes are further divided in main and sub class categories maintained in taxonomical hierarchy. The size of ontology varies due to the increase in number of classes and instances.

Ontologies can be made manually from scratch e.g. by extracting information from web and by merging already existing ontologies into new ontologies. But this manual process sometimes becomes very complex and time consuming especially when dealing with a large amount of data. Moreover, to support the process of semantic enrichment reengineering for the building of the web consisting of Meta data depends on the proliferation of ontologies and relational Meta data. This requires production of Meta data at high speed and low cost. So in these cases machine learning approaches can be very helpful in generating ontologies automatically because they provide real time schemes like classification rules, instance based learning, numeric predictions, clustering, Bayesian networks and decision trees for the generation of ontologies.

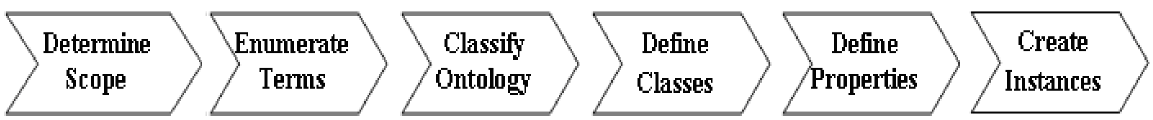


Figure 1. Ontology development activities [12]

Figure Legend. Six Ontology development activities i.e. Determine Scope, Enumerate Terms, Classify Ontology, Define Classes, Define Properties and Create Instances.

Ontology development is an iterative process based on six main activities: *Determine Scope, Enumerate Terms, Classify Ontology, Define Classes, Define Properties* and *Create Instances* as shown in Figure 1. In the beginning of an ontology development process it is very important to determine the scope otherwise it will be both time and effort consuming [12]. Then enumerated terms are needed to be identified to classify ontologies within their respective types. Classes and their respective properties along with their relationships and constraints are defined using identified enumerated terms. In the end only the instances are created and used. To implement ontology development process some experience, a powerful user friendly ontology supporting tool and communication between domain experts and developers is required e.g. LibraryWine [12].

3. Ontology Specific Languages

First step in building ontologies is to create the nodes and edges. Once the concepts (nodes) and relationships (edges) of graph based ontology are constructed then the next step is to quantify the strengths of semantic relationships [11]. Ontologies can be constructed manually and automatically by using some ontology supporting languages: *eXtensible Mark-up Language* (*XML*), *Resource Description Framework* (*RDF*) [5] and *Web Ontology Language* (*OWL*); all offer ways of more explicitly structuring and richly annotating Web pages.

• XML is one of the fundamental contributions towards middleware technologies [13]. It is a markup Meta language which allows sharing of information between different applications through markup, structure and transformation. As the major contribution towards semantic web, XML uses Data Type Definitions (DTD) and depends on data types, attributes, both internal and external elements structure documents and provide syntax serialization and abbreviation for data modelling [17]. The XML schema restricts the syntax to be only used for the structured documents, because of this XML has two main problems in process of information extraction; first it is without semantics and second is the arbitrary naming and structuring of elements [20].

• RDF, a URL based syntax data representation provides a secure and reliable mechanism for the exchange of metadata between web applications. RDF processes Meta data by making an abstract data model based on three object type attributes: *Resource, Property and Statement* [24]. Resource is an expression, property is an attribute to describe a resource and statement is a resource having some property and value. RDF uses three containers: *object bag, sequence, and alternative*, to keep multiple available and alternative values arranged in an order in resources and properties. The "object bag" contains resources, "sequence" contain resources along with their properties having single or multiple values arranged in order and "alternative" contains resources having alternate value(s) of a property [18]. RDF provides syntax serialization and abbreviation for RDF data modeling. Serialized syntax expresses the full capabilities of data modeling in a very regular fashion and abbreviated syntax includes additional constructs to provide a more compact form in representing a subset of the data model. RDF is more useful than XML in ontology construction because it provides semantic based features for data including domain independency, vocabulary and privileges in defining terminologies used in schema language. Furthermore it also provides syntax based on reification (statements about statements), data types, attributes, nesting, elements, element types, element container and no restrictions in structuring document like XML. RDF has its own grammar but it is not complete, it relies on the support of XML to fulfill its need. Moreover, the RDF modeling mechanism is insufficient in expressing various logical statements [17].

• The Web Ontology Language OWL was proposed by the W3C proposal in 2004 [35]. OWL is derived from American DARPA Agent Markup Language (DAML) [22]. OWL is based on ontology, inference and European Ontology Interchange Language (OIL) [23], and is intended to be an extension in RDF in expressing logical statements [21]. It provides an Application Programming Interface (API) for the development of Semantic Web application using ontology [31], influenced by XML Document Object Model (DOM) [36], can easily be used in any OWL supporting language editor e.g. Protégé 4. OWL API provides number of classes and interface for OWL based ontology modeling [34]. It is rich in vocabulary because it not only describes classes and properties but also provides the concept of namespace, import, cardinality relationship between the classes and enumerated classes. OWL has some specific limitations like only one "*Namespace*" per project is allowed, "*Import*" is not currently supported, no database backend and Multi-User support and a few OWL Language features are also missing [12].

Using above mentioned ontology specific languages, an extensible and customizable toolset is available i.e. *Protégé*, for the construction of ontologies. Protégé is with some excellent features towards automatic generation user-defined and modeled graphical interfaces for the acquisition of domain instances, extensible knowledge modelling and embedding standalone applications [12]. Furthermore Protégé provides plug ins acceptability to enhance its functionality e.g. Some new graphical user interface related features, new visualization libraries, import and export formats etc.

4. Semantic Web Applications

Residing in the domain of Semantic Web many products have been developed and and several approaches have been introduced by many researchers providing values in the implementation of semantic based applications with use of ontology [26]. Newly proposed approaches are helping in structuring data over the web to take advantage in implementing efficient web based information retrieval search mechanism. In this section we are discussing some of the recent SW based approaches e.g. Semantic Desktop [44], Reisewissen [30], Intelligent Semantic Oriented Agent based Search (I-SOAS) [27, 29] and Meta Data Search Layer[46].

Semantic Desktop; stepping into user's mid frame by implementing Personal Information Model (PIM). PIM is designed to improve the process for the identification of documents and retrieval of required document. The design is based on ontologies and classes. The relationships of classes and ontologies are predefined and the information can be accessed using RDF graphs. Four rules based on forward changing principle are defined to retrieve the information. This information is divided into three parts: *author* (single or team), *relevant project*, and *relevant solution*. The system works in the following way: first query runs aiming to find out the project and if project is found then it moves to find out the related documents of the project. The proposed architecture mainly consists of three main components: *receiver, interpreter and analyzer*. Receiver is used to provide index services and obtain the information about the structure of indexed files with the help of so called brainFiller. Interpreter first retrieves information (structure / unstructured) using full text search and then uses so called LiveLink. The contents of obtained information is structured with the help of manual annotation and meta data (based on their properties and preferences). As the last step, the analyzer queries (using Jena inference engine) the created RDF models to infer runs and uses F-Logic to integrate rules. Authors designed four case scenarios for proposed approach to share searched information: *Local Search, Group Search, Closed Community and Open Community*. Local search scenario only deals with the search mechanism and can only be applied to a personal desktop. Group search can be applied with *n* particular network domains. Closed community consists of a number of users having different roles but same topic where as open search consists of users with different roles and different topics.

Reisewissen is a hotel recommendation engine and travel information system. It provides quality services by semantically connecting, organizing and sharing the isolated pieces of information by transpiercing to data sources, caching & fetching of data, transforming data from heterogeneous to RDF models, mapping of ontologies between database and triples, matching RDF and non RDF based information [30]. Reisewissen is implemented by manually mapping ontologies (RDF models). Reisewissen identifies potential relevant knowledge sources and provides quality service by semantically connecting, organizing and sharing the currently isolated pieces of information in an online portal to anticipating customer behavior. The design of Reisewissen (Figure 2) is composed of three main components: *Data Connectors (DC), Evaluation Framework (EF)* and *Evaluation Engine (EE)*. Data connectors are used to provide transparency to data sources and transformation of data from heterogeneous to common data format (RDF and Java objects), moreover it also provides the caching and fetching of data. Evaluation Framework is a workbench to test the quality of data and rules by providing functions and filters to map resources and return result in decisive format (Boolean or float value). Evaluation Engine combines individual filters to rank and filter information by weighting and yielding the overall score. Information is obtained using Simple Object Access Protocol (SOAP) based web services and stored in both RDF and non RDF formats, which are then matched to find out the desired result. Data stored in RDF format is based on developed ontologies mapped between database and RDF triples. Moreover Reisewissen uses Prolog to capture ex-pert's knowledge which can be formalized and can generate new data by implementing the customer request in evaluator encapsulated rules. Data is matched semantically by combining data properties to ontology and similarities between two concepts are determined by distance reflecting their respective positions in hierarchy. As output a list of selected results are generated to customer.

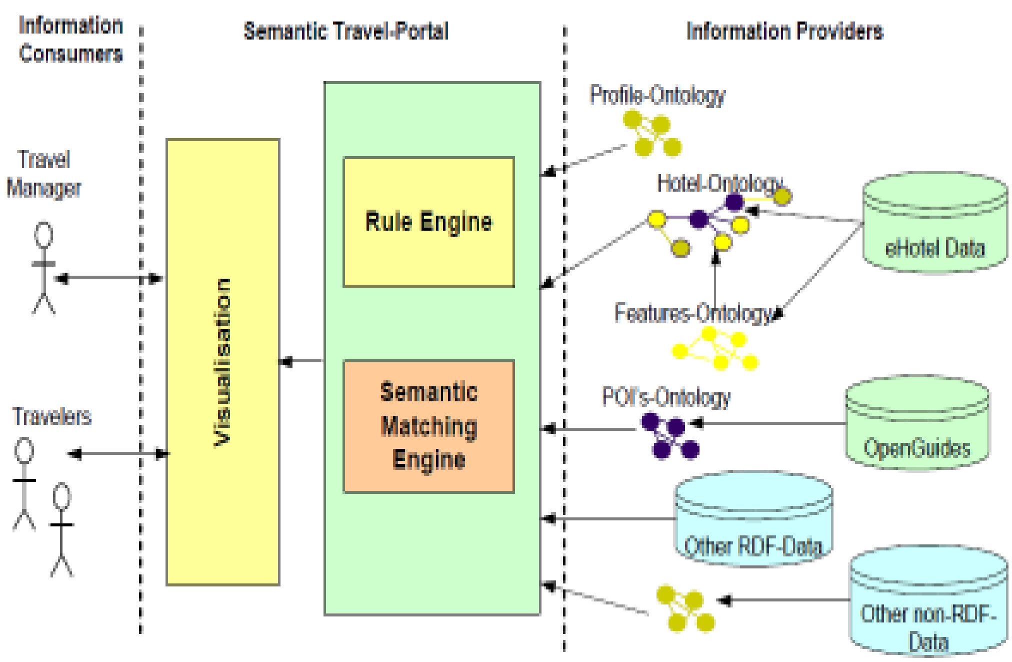


Figure 2. Reisewissen hotel recommendation engine [30]

Figure Legend. Reisewissen hotel recommendation engine consists of three major parts: The data connectors handling the transparent access to data sources and transformations, the evaluation framework providing functions (evaluators) to evaluate and filter resources and the actual evaluation engine combining the individual evaluators to rank and filter a set of hotels as well as yield evaluation results to a customer user interface.

I-SOAS is proposed to solve the problems of implementing semantic oriented information models, Meta data extraction and multiuser access in Product Data Management (PDM) Systems [48]. It provides a flexible multi user graphical interface [49], intelligent search [50], and knowledge management [45]. Without going into the details of all modules, we discuss only the most relevant one i.e. *intelligent search*. The overall job of intelligent search module is divided into five main iterative sequential steps i.e. *Data reading, Tokenization, Parsing, Semantic Modelling* and *Semantic based query generation* [47]. The main concept behind the organization of these five steps is to first understand the semantic hidden in the context of natural language based set of instructions and generate a semantic information processable models for the system's own understanding and information processing. At first the Data Reader reads and organizes input data from GUI into initial prioritized instructions list. Then the Data Tokenizer tokenizes instruction one by one, which are then treated by the Data Parser for parsing and semantic evaluation with respect to the grammar of used natural language. Then the Semantic Modeler filters the irrelevant semantic less data and generate Meta data based semantic model. Then in the last step the Semantic Based Query Generator is supposed to generate a new query used for further data storage and extraction of desired result. Following the concepts of semantic web and ontology construction, authors have created the ontology (Figure 3) using RDF with respect to the structure of proposed natural language (English) grammar of I-SOAS [42, 43].

Meta Data Layer; a process for Meta data search based on three questions for information extraction: *what user needs, where it lies,* and *how it can be retrieved*. The targeted objective is to identify the location from set of locations contained by a document and avoid looking into non-specific document. Scalability and efficiency of this approach is determined using simulation of documented Meta data keywords, location pointers, node connections and node knowledge. The whole process of identifying target location and search consists of nine procedural steps (Figure 4).

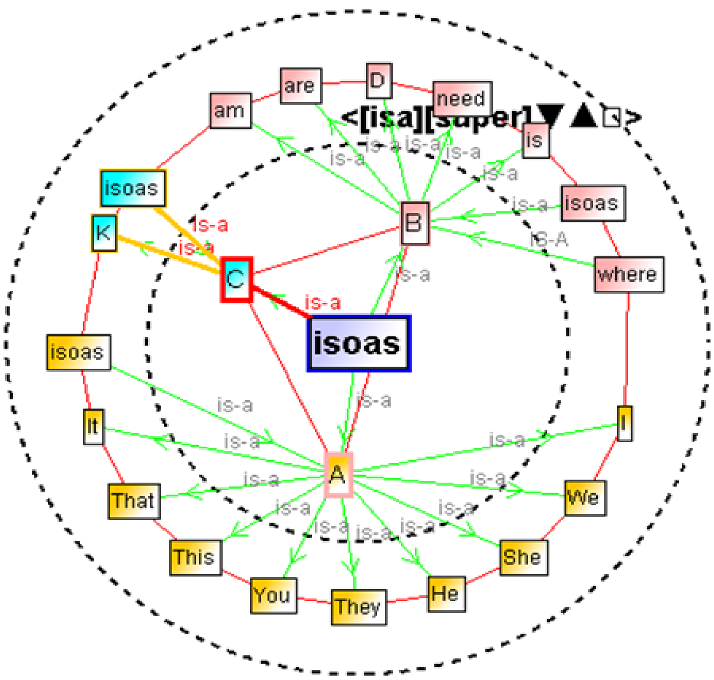


Figure 3. I-SOAS Ontology; Class relationships [43]

Figure Legend. I-SOAS Ontology; Class relationships consists of one main Class ISOAS, then three sub classes*: A*, *B* and *C*. All three subclasses contain their further subclasses and the relationships of this subclass with each other.

- Select target document from network having at least one keyword.
- Use keywords contained in document for the construction of a search query.
- Start with free node (not already containing any target document).
- Make a record of start node.

- Start node's knowledge treated as basic knowledge for the selection of other sub nodes.
- If number of nodes is equal to forwarding degree select those nodes.
- If number of nodes is less than forwarding degree select additional nodes.
- If number of nodes is more than equal to for-warding degree select subset of nodes.
- For each selected node, if node contains target document then update connectivity and if it doesn't then continue search using nodes.

Authors have explicitly mentioned that this search mechanism is good but there is still room for improvement in examining the path length of searches for different and same users characterized by their different query distributions. Moreover regarding the time to converge to a stable network, this can meet ambiguities and needs to have more realistic simulation using parameters and distributions.

Other than the mentioned ones, there are also some more beneficial ontology based approaches such as Google translate [54] join different language processing engines to predict, Bioinformatics related approaches e.g. protein interaction database such as i-Hop [55] and XplorMed [56].

5. Limitations of Semantic Web; Ontology

The development of ontology driven applications is difficult because of some disadvantages, limitations and principal problems which are as follows:

- Natural language parsers (used to parse the information to construct the ontology) are limited because they can only work over a single statement at a time [13].

- Not possible to define the boundaries of ontology based particular domain's abstract model.

- Not possible to automatically handle the increase in size of ontology (due to the increase in number of classes and instances).

- Creating ontologies manually is a time consuming process which becomes very complex when there is a large amount of data to create large number of ontologies. To take advantage in creating large number of ontologies by reducing the complexity and time, an automatic ontology creation mechanism is required. Some mechanisms are already proposed and implemented to create ontologies automatically but they are in-sufficient and less qualitative. While creating nouns based classes using existing automatic ontology creation mechanism is automatically possible now, it is quite impossible to identify the possible existing relationships between classes to draw the taxonomical hierarchy [14]. Furthermore it is also quite impossible to perform automatic emergence of ontologies to create new ontologies [16].

- Currently available ontology validators are restricted and not capable of validating all kind of ontologies e.g. based on complex inheritance relationship [53].

- Domain specific ontologies are highly dependent on the domain of the application and because of this dependency domain specific ontologies contain specific senses which are not possible to find in general purpose ontology [15].

- The process of semantic enrichment reengineering for web development consists of relational meta data required to be developed at high speed and in low cost depending on proliferation of ontologies, which is currently also not possible.

- Handling the dynamically raised calculations caused by the comparison of big complexities of similar ontologies is also not possible [16].

- Only one namespace per project is allowed during the ontology creation using OWL [12].

- Import is not currently supported during the ontology creation (using OWL) [12].

- No database backend support is available during the ontology creation (using OWL) [12].

- No multi user support is provided by any ontology supporting language [12].

Regardless of above mentioned limitations, using ontology is beneficial in and tantamount structuring data and implementing data extraction process for efficient information search.

We mention here furthermore, that as currently there is no fully automated ontology creation for the web possible, the method of highly user based structured web pages is also powerful, notably the Wiki movement (www.wikipedia.org) for instance to structure web pages of different bacterial genomes (e.g. Subti-Wiki, Staph-Wiki). Such approaches are complementary to automatic semantic web efforts and both profit from each other.

6. Conclusions

Introduction of World Wide Web brought new meaning to information sharing and communication. However advancement in Data formats made data accessibility much more complex. With the concept of Web Service, a new and innovative filed of Semantic Web came into being. A review research has been conducted in the field of Semantic Web and its importance has been elaborated in detail in this paper. In this paper, we have presented a major building block of Semantic Web i.e. *Ontology*, with some implementation technologies: *XML, RDF and OWL*, along with a brief concluding description of some Semantic Web applications: *Semantic Desktop, Reisewissen, I-SOAS and Meta Data Layer*. Furthermore, concluding the research review we have mentioned some ontology limitations needed to be overcome.

7. Acknowledgement

We (authors) are thankful to the University of Wuerzburg Germany for giving us the opportunity to work on this research. We are thankful to the blind reviewers and publishers for being publishing this manuscript. We thank DFG (TR34/Z1) for funding.

References

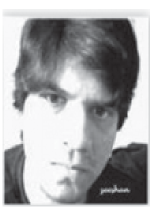
[1] Tim, B. L. (1994). Universal Resource Identifiers used in the World Wide Web. RFC 1630. Publ. Internet Society, Internet Engineering Task Force.
[2] David, H., Stefano, M., David, K. (2005). Piggy Bank: Experience the Semantic Web Inside Your Web Browser. Lecture Notes in Computer Science, V. 3729, Oct, p. 413 - 430.
[3] Tim, B. L., Hendler, J., Lassila, O. (2012). The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. Scientific American Special Online Issue. Publ. Scientific American. 24-30.
[4] Sebastian, R., Ryszard, K., Mariusz, C., Piotr, P., Krystian, S., Adam, W., Stefan, D. (2006). Building a Heterogeneous Network of Digital Libraries on Semantic Web, *In:* Proceedings of Semantic Systems from Visions to Applications 2006, Vienna Austria.
[5] Sean, B. P. (2007). The Semantic Web: An Introduction, last reviewed February, 28. <http://infomesh.net/2001/swintro>
[6] Wernher, B (2005). Ambient Intelligence Semantic Web or Web 2.0. *In:* Proceedings of Semantic Content Engineering.
[7] Witold, A., Tomasz, K., Krzysztof, Wêcel (2005).How Much Intelligence in the Semantic Web?. Volume 3528/2005, 1-6.
[8] Heiner, S. (2002) Approximate Information Filtering on the Semantic Web. *In:* Proceedings of the 25th Annual German Conference on AI: Advances in Artificial Intelligence, 114-128.
[9] Fensel, D. (2001). Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce. Springer-Verlag. ISBN 3540416021, p. 147.
[10] Okkyung, C., SeokHyun, Y., Myeongeun, O., San-gyong, H. (2003). Semantic Web Search Model for Information Retrieval of the Semantic Data. *In:* Proceedings of the 2nd international conference on Human.society@internet, 588-593.
[11] Gerhard, W., Jens, G., Ralf, S., Martin, T. (2004). Towards a Statistically Semantic Web. *In:* Proceedings of 23rd International Conference on Conceptual Modeling. 3-17.
[12] Holger, K., Mark, A. M., Natasha, F. N. (2003). Tutorial: Creating Semantic Web (OWL) Ontologies with Pro-tégé. *In:* 2nd International Semantic Web Conference.
[13] Grigoris, A., Frank, V. H. (2003). A Semantic Web Primer. The The MIT Press Cambridge, Massachusetts London, England
[14] José, S., Paulo, Q. (2004) A Methodology to Create Legal Ontologies in a Logic Programming Based Web Information Retrieval System. Artif. Intell. Law. 12(4) 397-417.
[15] Amalia, T., Laurent, R., Dalila, B. (2002). Vulcain – An Ontology-Based Information Extraction System, *In:* Proceedings of 6th International Conference on Applications of Natural Language to Information Systems-Revised. 64-75.
[16] Marc, E., York, Sure, (2004). Ontology Mapping - An Integrated Approach. *In:* Proceedings of 1st European Semantic Web Symposium. 76-91
[17] Klaus, T., Herman, M. (2006). Semantic technologies - An Introduction. Semantic Technologies Showcase the Austrian Situation. 15-20.
[18] Ora, L., Ralph R. S. (2007). Resource Description Framework (RDF) Model and Syntax Specification. reviewed February 2012. <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222>

[19] Sachin, R. (2009). Web2 to Web3 moving ahead with web technologies - lots of resources and articles on web2 and web3. reviewed 01 October. <http://sachinkraj.wordpress.com/2007/10/10/web2-to-web3-moving-ahead-with-web-technologies>
[20] The W3C Extensible Markup Language (XML). reviewed February 2012. <http://www.w3.org/XML>
[21] OWL; Web Ontology Language. reviewed February 2012. <http://www.w3.org/TR/owl-features>
[22] DAML. reviewed February 2012. <http://www.daml.org>
[23] Welcome to OIL. viewed February 2012. <http://www.ontoknowledge.org/oil>
[24] What Is An RDF Triple?. reviewed February 2012. <http://www.robertprice.co.uk/robblog/archive/2004/10/What_Is_An_RDF_Triple_.shtml>
[25] Borys, O. (2001). Learning of ontologies for the Web: the analysis of existent approaches. *In:* Proceedings of the International Workshop on Web Dynamics, held in conj. with the 8th International Conference on Database Theory.
[26] Galia, A. (2005). Language Technologies Meet Ontology Acquisition. *In:* Proceedings of the 13th International Conference on Conceptual Structures ICCS 2005, Springer. 3596. 367-380.
[27] Zeeshan, A. (2009). Intelligent - Semantic Oriented Agent Based Search (I-SOAS). *In:* proceedings of Doctoral Symposium at ACM International Conference on Frontiers of Information Technology.
[28] Zeeshan, A., Saman, M. (2011). Review Middleware Technologies, Chain Web Grid Services. *International Journal of Web Applications*, 3 (4) 197-205.
[29] Zeeshan, A. (2009). Proposing Semantic Oriented Agent and Knowledge base Product Data Management. *Information Management and Computer Security Journal*. 17(5) 360-371.
[30] Magnus, N., Malgorzata, M., Robert, T. (2006). Improving Online Hotel Search What Do We Need Semantic For? *In:* Proceedings of Semantic Systems from Visions to Applications.
[31] Horridge, M., Bechhofer, S. (2009). The OWL API: A Java API for Working with OWL 2 Ontologies. *In:* 6th OWL Experienced and Directions Workshop.
[32] Kathrin, D., Ronald, C., Annette, t. T., Nicolette, d. K. (2011). Comparison of Reasoners for large Ontologies in the OWL 2 EL Profile. Semantic Web. 2 (2) 71-87
[33] Pascal, H., Krzysztof, J. (2011). Semantic Web Tools and Systems. Semantic Web, 2 (2) 1-2.
[34] Matthew H., Sean, B. (2011). The OWL API: A Java API for OWL Ontologies. Semantic Web. 2 (1) 11-21
[35] Peter, F. P., Patrick, H., Ian, H. (2004). OWL Web Ontology Language semantics and abstract syntax. W3C Recommendation. 10 February.
[36] Bob, D. (2009). OWL 2 Web Ontology Language Document Overview. W3C Recommendation, World Wide Web Consortium, reviewed February. < http://www.w3.org/TR/ owl2-overview/>.
[37] Gruber, T. R. (1993). A translation approach to portable ontology specifications. Knowledge acquisition. 5(2) 199-200.
[38] Quine, O. (2004). On what there is. Cambridge Belknap Press, Harvard University.
[39] Hofweber, T. (2009). Logic and Ontology. Stanford Encyclopaedia of Philosophy.
[40] Schulz, S., Stenzhorn, H., Boeker, M., Smith, B. (2009). Strengths and limitations of formal ontologies in the biomedical domain. RECIIS Rev Electron Comun Inf Inov Saude. 3 (1) 31–45.
[41] Baader, F., Calvanese, D., Mcguinness, D., Nardi, D., Patel, S. P. (2007). The Description Logic Handbook Theory, Implementation, and Applications (2nd Edition). Cambridge University Press.
[42] Zeeshan, A., Thomas, D., Saman, M. (2012). Role of Ontology in NLP Grammar Construction for Semantic based Search Implementation in Product Data Management Systems., *International Journal of Management, IT & Engineering*. 2(2) 1-40.
[43] Zeeshan, A., Saman, M., Thomas, D. (2010). Towards Design and Implementation of a Language Technology based Information Processor for PDM Systems. International Science & Technology Transactions of Information Technology- Theory and Applications, 1 (1) 1-7.
[44] Leo, S., Gunnar, A.G., Thomas, R. (2008). The Semantic Desktop as a foundation for PIM research. *In:* Proceedings of the Personal Information Management Workshop.
[45] Zeeshan, A. (2011). Designing Knowledge Base towards PDMS, *International Journal of Information Technology and Engineering*, 2(1) 9-12.
[46] Sam, J. (2003). P2P MetaData Search Layers. *In:* Proceedings of Second International Workshop on Agents and Peer-to-Peer Computing. 101-112.
[47] Zeeshan, A., Detlef, G. (2009). Design Implementation of I-SOAS IPM for Advanced Product Data Management. *In:* Proceedings of The Second IEEE International Conference on Computer, Control & Communication, 1-5.
[48] Zeeshan, A., Detlef, G. (2007). Contributions of PDM Systems in Organizational Technical Data Management. *In:* Proceedings of The First IEEE International Conference on Computer, Control & Communication, November.
[49] Zeeshan, A. (2011). Designing Flexible GUI to Increase the Acceptance Rate of Product Data Management Systems in Industry. *International Journal of Computer Science & Emerging Technologies*, 2(1) 100-109.

[50] Zeeshan, A. (2010). Proposing LT based Search in PDM Systems for Better Information Retrieval. International Journal of Computer Science & Emerging Technologies. 1(4) 86-100.
[51] Wainewright, P. (2002). Web Services Infrastructure, The global utility for real-time business. A white paper.
[52] Thomas, M., Stefan, T., Isabelle, R. (2002). Transactional Attitudes: Reliable Composition of Autonomous Web Services. *In:* Proceedings of Workshop on Dependable Middleware-based Systems, International Conference on Dependable Systems and Networks. IEEE, ISBN: 0-7695-1597-5, p. 792.
[53] Grit, D., Son, N., Andrew, T. (2004). OWL-S Semantics of Security Web Services: a Case Study. The Semantic Web: Research and Applications, Publ. Springer, V. 3053, 240-253.
[54] Visit, H., Vuong, T. X. (2005). Semantic Web Agent Communication Capable of Reasoning with Ontology and Agent Locations. World Academy of Science, Engineering and Technology. 10.
[55] Daniel, J. L. (2008). Intelligent agents and the Semantic Web; Developing an intelligent Web. IBM Technical Library.
[54] Google Translate. Last reviewed 1st March 2012, <http://translate.google.com/translate_tools>.
[55] iHOP. Last reviewed 1st March 2012, < http://www.ihop-net.org/UniPub/iHOP/ >.
[56] Perez-Iratxeta, C., Pérez,AJ, Bork, P., Andrade, M. A. (2003). Update on XplorMed: a web server for exploring scientific literature. Nucleic Acids Research. 31, 3866-3868.

Authors Bibliography

Prof. Thomas Dandekar is the Chair of the Department of Bioinformatics, Biocenter, Am Hubland, University of Wuerzburg Germany, and interested in combining genome-based bioinformatics with systems biological modelling.



Zeeshan Ahmed is the software engineer, researcher at the Department of Bioinformatics, Biocenter, Am Hubland, University of Wuerzburg, Germany, and has interestes in developing intelligent software systems towards product data analysis, visualization and management.

Saman Majeed is the doctoral scientist the Department of Bioinformatics, Biocenter, Am Hubland, University of Wuerzburg, Germany.