

Ontology Languages For The Semantic Web

An Updated Review

C. Ramesh¹, Dr. K. V. Chalapati Rao² and Dr. A. Goverdhan³

¹ CVR College of Engineering, Department of CSE, Ibrahimpatan, R.R.District, A.P., India
Email: hmcr.ramesh@gmail.com

² CVR College of Engineering, Department of CSE, Ibrahimpatan, R.R.District, A.P., India
Email: chalapatiraokv@gmail.com

³ JNTUH College of Engineering, Department of CSE, Jagityala, Karimnagar, A.P., India
Email: govardhan_cse@yahoo.co.in

Abstract—The phenomenal growth of content on the Web has made it difficult to locate, organize, and retrieve information. One way to cope with this problem is to automate these tasks. But because of the complexity of natural-language processing, we still do not have machines that can understand and analyze the content on the Web as humans do. To handle this problem, many new research initiatives and commercial enterprises have been set up to enrich available information with machine processable semantics. Such support is essential for “bringing the Web to its full potential”. Tim Berners-Lee, known as the inventor of the World Wide Web (WWW), has a vision for the future of the World Wide Web, which he calls “The Semantic Web”, which aims to provide an intelligent access to heterogeneous and distributed information, enabling software products (agents) to mediate between user needs and the information sources available. Ontologies have proven to be an essential backbone technology in realizing the vision of Semantic Web. This paper summarizes ongoing research in the development of Ontology Languages for realizing the Semantic Web to its full potential.

Index Terms—WWW, Semantic Web, Ontology Languages, Ontology, Semantic Web development tools.

I. INTRODUCTION

The World Wide Web, the largest repository of information ever assembled, continues to grow at exponential rate. It contains information on almost every imaginable subject, and this information is instantaneously available to anyone with an internet connection. However, it is difficult for machines to process and integrate this information meaningfully. Semantic Web [1], envisioned by Tim Berners - Lee, is the initiative taken by W3C to solve the problems in meeting the challenge faced by current Web.

The full vision of the Semantic Web has yet to be fully realized, but there has been considerable progress in the development and use of standards, languages, technologies and applications. Ontologies have proven to be an essential element for representing and sharing knowledge across agents in various areas, such as natural language processing, bioinformatics, formal languages and e-commerce [2]. Ontologies are corner stone in realizing the vision of Semantic Web. In recent years, several ontology languages have developed for realizing the Semantic Web. A summary of the recent approaches

for the development of ontology languages for the Semantic Web is the focus of our paper.

The rest of the paper is organized as follows: in section II, an overview of Semantic Web and the role of Web ontology languages is presented and requirements for Ontology languages are explained; in Section III the recent developments in Ontology languages for the Semantic Web are presented; Section IV describes RDF; Section V describes Semantic Web development tools and at end, the conclusion and future enhancements are discussed.

II. OVERVIEW OF SEMANTIC WEB AND THE ROLE OF ONTOLOGY LANGUAGES

A. Semantic Web

Semantic Web is the next stage in the evolution of the Web [3], where information is given well-defined meaning, enabling better coordination among computers and people. Currently information found on the Web is mainly for human consumption and is not machine-understandable. It is quite difficult to automate things on the Web and the enormous volume of information on the Web makes it even more difficult to manage it manually.

Semantic Web is not a replacement for the Web, but extension to the current Web and it is about making the Web more understandable by machines [4]. The impact of the Semantic Web, when it is realized, will be of enormous use, both for people and business, because it will achieve interoperability of information between web applications (agents, web services, etc.). Fig.1 presents the layered architecture of Semantic Web.

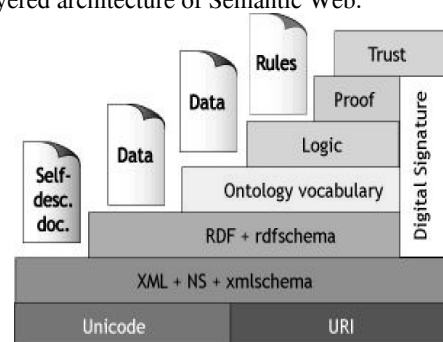


Figure1. Semantic Web Architecture

B. The Role of Ontology Languages

Although the notion of ontologies is independent of the web, ontologies play a special role in the architecture of the Semantic Web. The key challenge of the Semantic Web is to ensure a shared interpretation of information. Related information sources should use the same concepts to reference the same real world entities or at least there should be a way to determine if two sources refer to the identical entities, but possibly using different vocabularies. Ontologies and ontology language are the key enabling technology in this respect. An ontology, by its most cited definition in AI, is a shared, formal conceptualization of a domain, i.e. a description of concepts and their relationships [5]. Ontologies are domain models with two special characteristics, which lead to the notion of shared meaning or semantics:

1. Ontologies are expressed in formal languages with well-defined semantics.
2. Ontologies are built upon shared understanding within a community, both of people and software agents.

Ontologies come in different flavors; from flat lexicons with very few relationships to very expressive ontologies, which attempt to capture every possible aspect of the domain and have broad support for axioms [6]. Web ontology languages allow us to formally express ontologies. The expressiveness of an ontology is limited by the ontology language, which is used for the specification of the ontology. Many ontology languages have been developed, both with limited and with high expressivity. We put our focus on the development of Ontology languages for the Semantic Web.

C. Requirements for Ontology Languages

Ontology languages allow users to write explicit, formal conceptualizations of domains models. In recent years, several markup languages have been developed for realizing the Semantic Web.

The main requirements¹ of an Ontology language are [7]:

- (1) Have a compact syntax.
- (2) Be highly intuitive to humans.
- (3) Have a well-defined formal semantics.
- (4) Be able to represent human knowledge.
- (5) Include reasoning properties.
- (6) Have the potential for building the knowledge bases.
- (7) Have a proper link with existing Web standards to ensure interoperability.

Unlike some existing markup languages, specifically HTML, a Semantic Web language must describe meaning in a machine readable way. Therefore an Ontology language needs not only to include the ability to specify vocabularies but also the means to formally define it in such a way that it will work for automated reasoning.

III. RECENT APPROACHES FOR THE DEVELOPMENT OF SEMANTIC WEB ONTOLOGY LANGUAGES

In this section some recent Ontology languages, particularly needful for the development of Semantic Web are presented, namely eXtended Markup Language, Resource Description Framework, Simple HTML Ontology extensions, Ontology interchange language, DAML, Web Ontology Language.

A. eXtended Markup Language

The eXtended Markup Language (XML) was the first language to separate the markup of Web contents from the Web presentation, facilitating the representation of task-specific and domain specific data on the Web. It provides a uniform framework for exchanging data between applications. Document Type Definition (DTD) and XML Schemas, were introduced, to enforce constraints on which tags to use, how they should be nested within a document. An XML Schema is a defined as grammar or definition language that constrains conforming XML documents to a specific vocabulary and a specific hierarchical structure. Unfortunately XML lacks semantics, software agents cannot be guaranteed to determine the intended interpretation of its tags. It is designed to describe the structure of the document, not the content.

B. Resource Description Framework

RDF², developed by the W3C for describing web resources, allows the specification of the semantics of data based on XML in a standardized, interoperable manner. It also provides mechanisms to explicitly represent services, processing and business models, while allowing recognition of non explicit information. RDF is becoming a widely recognized language and a representation formalism, that can serve as a worldwide interlingua for information interchange. The RDF description model uses object –attribute –value triples also known as statement. Its goal is to add formal semantics to the web and provide a data model and syntax convention for representing the semantics of the data in a standardized manner. It provides a means of describing the relationships among resources in terms of the named properties and values. RDF has significant advantages over XML. The object – attribute structure provides natural semantic units because all objects are independent entities. RDF played an important role as a basis for DARPA Agent Markup Language (DAML), whose layers of logic are built on the top of the basic RDF framework [8]. The descriptive power of RDF is minimal that in practice, it is always used in combination with RDF Schema. RDF Schema is a simple extension of RDF defining a modeling vocabulary with notion of classes and properties.

1. <http://w3.org/DesignIssues/Logic.html>

2. <http://w3.org/TR/REC-rdf-syntax/>

IV RDF AND RDF SCHEMA LIMITATIONS

Though RDF and RDF Schema form the basis for building Semantic Web, together they still lacked sufficient expressive power. For example, they cannot define: (1) the properties of properties, (2) necessary and sufficient conditions for class memberships, or (3) equivalence and disjointness of classes. In addition, the only constraints expressible are domain and range constraints on properties. As a result, the semantics have remained weakly specified.

A. Simple HTML Ontology Extensions

Giving the authors the ability to embed knowledge directly into HTML pages, making it also simple for user agents and robots to retrieve and store knowledge, was the goal of the so-called *Simple HTML Ontology Extensions* (SHOE). This approach allows authors to add semantic content to web pages, relating the content to common Ontologies that provide contextual information about the domain [9]. Most web pages with SHOE annotations tend to have tags that categorize concepts, therefore there is no need for complex inference rules to perform automatic classification [10]. This approach extends HTML with a set of object-oriented tags to provide structure for knowledge acquisition. It associates meaning with content by committing web pages to existing ontologies. These ontologies permit the discovery of implicit knowledge through the use of taxonomies and inference rules, allowing information providers to encode only the necessary information into their web pages. An ontology tag delimits the machine-readable portion of the ontology. Some other tags³ complement the definition of Ontologies. SHOE focuses on the problem of maintaining consistency as the ontologies evolve. In "Ref. [11]" the use of SHOE in a real world internet application is described. Tools for annotating pages, information gathering tasks and querying are provided.

B. Ontology Interchange Language

Though RDF Schema is quite simple compared to a full-fledged knowledge representation languages, it lacks sufficient expressive power. Hence to specify the meaning of data more precisely, richer languages are necessary.

OIL(Ontology Inference Layer or Ontology Interchange Language), was developed in the Onto Knowledge project⁴ with an aim to combine the best features of frame and DL(Description Logic) based knowledge representation systems, while at the same time maximizing compatibility with emerging web standards. It is compatible with the RDF syntax [12]. OIL has a frame-like syntax, which facilitates tool building, yet can be mapped onto an expressive description logic (DL), which facilitates the provision of reasoning services. OIL, built on top of RDF(S) has following layers: Core OIL groups the OIL primitives that have a direct mapping to RDF(S) primitives; Standard OIL is the complete OIL model, using more primitives than the ones defined in RDF(S); Instance OIL adds instance of

concepts and roles to the previous model; and Heavy OIL is the layer for future extensions of OIL. OilEd is one of the popular ontology editing tool in OIL.

C. The DARPA Agent Markup Language

The DARPA Agent Markup Language (DAML) is a US government – sponsored endeavor aimed at providing the foundations for the next Web evolution, the Semantic Web. DAML [13] consists of two portions, the ontology language and a language for expressing constraints and adding inference rules. It also includes mappings to other Semantic Web languages such as SHOE, OIL, KIF, XML and RDF. Building on top of the RDF and RDFS and with its root in description logics, the Ontology Language (DAML+ OIL) [14] has a well defined model - theoretic semantics as well as an axiomatic specification that determines the language intended interpretations. This makes it an unambiguously computer interpretable language, thus making it amenable to agent interoperability and automated – reasoning techniques. The Inference Language (DAML-L) is a logical language with well defined semantics and the ability to express at least propositional Horn clauses, which enable compact representation of constraints and rules for reasoning. The language ties the information on a page to machine-readable semantics and allows communities to extend simple ontologies for their own use. In addition, it provides mechanisms for the explicit representation of services, processes and business models so as to allow non- explicit information to be recognized [15]. DAML + OIL and DAML – L together provide a markup language for the Semantic Web with expressive power and a well-defined semantics for reasoning. The DAML family of markup languages enables web service providers to develop semantically grounded, rich representations of web services that a number of different agent architectures and technologies can exploit to a variety of different ends [16].

D. Web Ontology Language (OWL)

Built on top of RDFS, the OWL provides a more expressive vocabulary along with a formalism based on predicate logic and descriptive logic. It uses RDFS/ XML – based syntax⁵. It is regarded as a W3C standard ontology language for Semantic Web. It is compatible with an early ontology languages, including SHOE, DAML + OIL, and provides the engineer with more power to express semantics. It includes four concepts that form the basis of an OWL document: (1) classes, (2) relationship between classes, (3) properties of classes, and (4) constraints on relationships between classes and properties of classes. Reasoning engines can make use of these to carry out logical inferences and derive knowledge. OWL is considered one of the fundamental technologies underpinning the Semantic Web, and has attracted both academic and commercial interest.

E. Three species of OWL:

The W3C has defined OWL to include three different sublanguages (OWL FULL, OWL DL, OWL Lite) in order

3. <http://cs.umd.edu/projects/plus/SHOE/ontologies.html>

4. <http://ontoknowledge.org/OIL>

5. <http://w3.org/TR/owl-features>

to offer different balances of expressive power and efficient reasoning.

OWL Full: The entire language is called OWL Full and it uses all the primitives and allows their combination with RDF and RDFS. OWL Full supports maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. For example, in OWL Full a class can be treated simultaneously as a collection of individuals and as an individual in its own right.

OWL Full allows an ontology to augment the meaning of the pre-defined (RDF or OWL) vocabulary. It is unlikely that any reasoning software will be able to support complete reasoning for every feature of OWL Full. The advantage of OWL Full is that it is fully upward compatible with RDF, both syntactically and semantically: any legal RDF document is also a legal OWL Full document, and any valid RDF/RDFS Schema conclusion is also a valid OWL Full conclusion. The disadvantage of OWL Full is that the language has become so powerful as to be undecidable, dashing any hope of complete (let alone efficient) reasoning support.

OWL DL: OWL DL supports those users who want the maximum expressiveness while retaining computational completeness (all conclusions are guaranteed to be computable) and decidability (all computations will finish in finite time). OWL DL includes all OWL language constructs, but they can be used only under certain restrictions (for example, while a class may be a subclass of many classes, a class cannot be an instance of another class). OWL DL is so named due to its correspondence with description logics.

The advantage of this language is that it permits efficient reasoning support. The disadvantage is that we lose full compatibility with RDF. An RDF document will in general have to be extended in some ways and restricted in others before it becomes a legal OWL DL document. Conversely, every legal OWL DL document is still a legal RDF document.

OWL Lite: OWL Lite is the restricted subset of OWL DL. It supports the users primarily needing a classification hierarchy and simple constraints. For example, while it supports cardinality constraints, it only permits cardinality values of 0 or 1. It has a lower formal complexity than OWL DL. The advantage of this language is that, it is both easier to grasp (for users) and easier to implement (for tool builders). The disadvantage is of course a restricted expressivity.

V. SEMANTIC WEB DEVELOPMENT TOOLS

The unique needs of Semantic Web requires tools for ontology development⁶, content generation and content analysis [17]. In particular, we need the following elements:

- Editors and semi-automatic construction tools to build new ontologies.
- Annotation tools to link unstructured and semi-structured information sources with meta data.
- Reasoners for reasoning support.
- Reusing and Merging Ontologies: Ontology library systems.
- Ontology Environments to create new ontologies by reusing existing ones.

In the following, we will briefly describe examples for these technologies.

A. Common Editors Used For Building Ontologies

- ✓ DAG-Edit provides an interface to browse, query and edit vocabularies with a DAG data structure.
- ✓ Protégé⁷ is the most widely used tool for creating ontologies and knowledge bases.
- ✓ SMORE⁸
- ✓ WebOnto is a java applet coupled with a Web server that allows users to browse and edit knowledge models.
- ✓ OilEd
- ✓ OntoStudio⁹ is the most widespread commercial modeling environment for creating and maintaining ontologies.

B. Annotation tools

The best known tools around annotation and authoring are:

- Annotea
- Annozilla

C. Reasoning Service

Inference engines process the knowledge available in the Semantic Web by deducing new knowledge from already specified knowledge.

- Jena¹⁰ provides a programmatic environment for RDF, RDFS and OWL, SPARQL and includes a rule-based inference engine.
- The FaCT (Fast Classification of Terminologies) can be used to automatically derive concept hierarchies. It is a Description Logic (DL) classifier that makes use of the well-defined semantics of OIL.
- RACER and Ontobroker are other examples of Reasoners.

D. Ontology Merging

Ontology mapping enables interoperability among different sources in the Semantic Web. It is required for combining distributed and heterogeneous ontologies. Ontology mapping transforms the source ontology into the target ontology based on semantic relations.

Glue and OntoMorph are some of the popular ontology mapping tools.

E. Ontology Libraries and Environments

Examples of Ontology library systems are:

- Web Onto
- Ontolingua ontology library

6. <http://w3.org/wiki/SemanticWebTools>

7. <http://protégé.stanford.edu/>

8. <http://mindswap.org/2004/SMORE>

9. <http://ontoprise.de/en/products/ontostudio/>

10. <http://jena.sourceforge.net/>

- DAML Ontology library
- SHOE
- Ontology Server
- IEEE Standard Upper Ontology
- (IEEE), Sesame24, OntoServer25, and ONIONS.

CONCLUSIONS

Semantic web, a vision of Tim Berners – Lee, aims at providing intelligent access to distributed information, so as to enable users to obtain more directly the information needed. This is achieved through appropriate structuring of the information in various sources, with sets of inference rules facilitating automatic reasoning. The current state of the art in the field of ontologies and ontology languages, that form the backbone for extracting semantic content of information, has been reviewed in this paper. An overview of the current Semantic Web technologies is also a highlight of our paper. The field is wide open and is currently an active area of research, with potentially, for significant advances.

REFERENCES

- [1] T. Berners-Lee, J. Hendler, O. Lassila, "The Semantic Web", *Scientific American*, May 2001.
- [2] Gómez-Pérez, Oscar Corcho, "Ontology Languages for the Semantic Web", *IEEE Intelligent Systems*, vol.17, no.1, 2002, pp.54-60.
- [3] T. Berners-Lee, "Weaving the Web". *Orion Business Books*, 1999.
- [4] J. Heflin, J. Hendler, "A portrait of the Semantic Web in action," *IEEE Intelligent Systems*, vol. 16, no. 2, 2001, pp. 54–59.
- [5] Tom R. Gruber, "Towards Principles for the Design of Ontologies Used for Knowledge sharing". in *N.Guarino, et al., (eds.), Formal Ontology, Conceptual Analysis and Knowledge Representation, Deventer*, The Netherlands, 1993. Kluwer Academics Publishers.
- [6] D. Fensel, "Ontologies: a silver bullet for Knowledge Management and Electronic Commerce", *Springer-Verlag*, Berlin, 2003
- [7] D.Fensel, et al., "OIL: an ontology infrastructure for the semantic web", *IEEE Intelligent Systems*, vol.16, 2001, pp.38-45.
- [8] D. Alferes, et al., "Semantic web logic programming tools", in: *F. Bry, et al. (Eds.) , Principles and Practice of Semantic Web Reasoning; International Workshop, PPSWR 2003,LNCS, Springer*, Vol. 2901, 2003, pp. 16-32.
- [9] S.Luke, et al., "Ontology – based web agents", in: *First International Conference on Autonomous Agents, ACM*, New York, 1997, pp. 59-66.
- [10] J.Heflin, J. Hendler, "Dynamic ontologies on the web", in: *American Association For Artificial Intelligence Conference, AAAI Press*, California, 2000, pp. 251-254.
- [11] J.Heflin, et al., "Applying ontology to the web: a case study", *Engineering Applications of Bio-Inspired Artificial Neural Networks*, 1607.
- [12] Harmelen, F. and Horrocks, I., "FAQs on OIL: The ontology Inference Layer", *IEEE Intelligent Systems*, 15(6), 69-72. (Nov./Dec. 2000).
- [13] Hendler, J. and McGuinness, D., "DARPA Agent Markup Language.", *IEEE Intelligent Systems*, 15(6): 72-73 (2001).
- [14] I.Horrocks, "DAML + OIL : a reasonable web ontology language", in: *Lecture Notes in Computer Science (LNCS)*, vol. 2287, Springer-Verlag, Berlin, 2002, pp. 2-13.
- [15] J. Hendler, "Agents and the Semantic Web", *IEEE Intelligent Systems*, 16 (2), 2001, pp.30-37.
- [16] S McIraith, et al., "Semantic Web Services", *IEEE Intelligent Systems*, 16 (2), 2001, pp.46-53.
- [17] D. Fensel and M. Musen, "Special Issue on Semantic Web Technology", *IEEE Intelligent Systems*, 16(2), 2001.