Attempting to build the Semantic Web: The Ontological Approach

Erhardt Graeff

Rochester Institute of Technology

December 14, 2004

Attempting to build the Semantic Web: The Ontological Approach

When father of the Internet Tim Berners-Lee first envisioned the World Wide Web, he imagined it as "an information space, with the goal that it should be useful not only for human-human communication, but also that machines would be able to participate and help." (1998, Introduction, para. 1) However, what amassed was a mess of poorly formed HTML documents boasting animated GIFs and information displayed without regard for meaning or context. What Berners-Lee was wishing for, and continues to wish for, is a better World Wide Web—a Semantic Web. This ultimate realization of the Internet's potential is something that Berners-Lee and the World Wide Web Consortium (W3C) are still working on. With millions of users and billions of documents, the web is constantly growing and evolving. The W3C hopes that it evolves into the Semantic Web—and that hope lies in something called an ontology. (Clark, 2002)

What is the Semantic Web?

To begin, we must have an understanding of what the Semantic Web is, or rather what it will hopefully accomplish. Obviously, the premise has something to due with linguistic meaning given the use of the adjective "semantic." Berners-Lee coined the term Semantic Web, and defines it thusly in an article for *Scientific American*, "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation." (W3C, 2001, para. 2) The key words in his statement are "extension" and "meaning." In "meaning," we are given the connection to semantics—a context for our information and data, essentially: metadata. Berners-Lee and the W3C began working on the element of "extension" before the term "Semantic Web" was ever used. This comes in the form of eXtensible Markup Language or XML.

In its initial draft, XML was described as a simple dialect based on Standard Generalized Markup Language, or SGML, created for the purpose of making streamlined web data more generic, and fully compatible with the existing technologies of SGML and HTML. (W3C, 1996) Already embodying the Semantic Web's tenet of interoperability and flexibility, XML quickly became the foundation of the W3C's approach to the Semantic Web. What XML enables is the creation of document definitions or schemas. Schemas provide a vocabulary and organizational structure for creating customized document templates. (Geroimenko, 2004) However, XML only provides this syntactic interoperability. It is not capable of ensuring that the markup elements in one particular document will convey any meaning to a document's recipient. The templates do however form the basis for defining customized languages; and herein is the potential for adding conveyable "meaning" to the data, through the virtually unlimited "extensibility" offered by XML. What was lacking was a way for both parties sharing XML documents to know and understand what each element means. (Daconta, Obrst, & Smith, 2003) The solution to this issue was an ontology.

What is an Ontology?

Ontologies used to be a term relegated to the field of Philosophy. But with the advancement of computers throughout the 1990's it has taken on new applications. First, it was employed in artificial intelligence, or AI, research to "facilitate knowledge sharing and reuse." (Fensel, 2004, p. 3) Considering the computer-automation desired of the Semantic Web, it was a small step for ontologies to make there way from AI into the W3C's plan of action. In relation to our linguistic definitions in the previous section, Vladimir Geroimenko defines an ontology as "an explicit representation of the MEANING of terms in a VOCABULARY, and their interrelationships." (2004, p. 109) As defined, an ontology finally gives us the ability to create context for, and extract meaning from, various web documents. Dieter Fensel gives the following two explanatory notes on how an ontology makes this possible:

> Ontologies define formal semantics for information, thus allowing information processing by a computer. Ontologies define real-world semantics, which makes it possible to link machine-processable content with meaning for humans based on consensual terminologies. (2004, p. 4)

Thus, an electronic ontology provides a consistent way of cataloguing data and metadata for a body of information. And the cataloguing metaphor works well for parallels in the realm of Library Science, where there are already examples of interoperable metadata ontologies at work in projects such as the Dublin Core Metadata Initiative. (Dumbill, 2000)

In order to serve its purpose effectively, an ontology should conform to the following five principles as outlined by T. R. Gruber: clarity, coherence, extendibility, minimal encoding bias, and minimal ontological commitment.

- Clarity: an ontology should make sure its defined terms are completely and objectively defined to avoid miscommunication of meaning.
- Coherence: an ontology should ensure that no definition contradicts itself or any other definition.
- Extendibility: an ontology should allow for new terms based on existing vocabulary to be defined for special cases.

- Minimal encoding bias: an ontology should strive to reduce symbol-level encoding of definitions to allow for differing implementations of the vocabulary.
- Minimal ontological commitment: an ontology should provide definitions that are as specific as possible for the given term's purpose to reduce the need of altering the vocabulary later. (Gómez-Pérez, Fernández-López, & Corcho, 2004)

Despite these guidelines, just the idea of creating a standard ontology for web-based data raises several questions, such as: Who decides how the ontology is formed? And how do you make sure it is truly universal? To address these questions, the W3C formed a group, commonly known as WebOnt, to focus on the task of engineering a flexible web ontology.

What is WebOnt?

WebOnt is short for Web-Ontology Working Group. Chartered in November of 2001, WebOnt was specifically commissioned under the W3C's Semantic Web Activity to develop a new ontological language that extended the pre-existing metadata technologies/standards. (W3C, 2001) WebOnt completed its mission in February of 2004 by issuing a finished product for W3C recommendation. (W3C, 2004b) This finished product was OWL, a Semantic Web-savvy ontological update to RDF/RDFS and DAML+OIL.

What are RDF and RDFS?

RDF stands for Resource Description Framework. This language was the first bridge between XML and an ontology. Predating WebOnt by nearly 4 years, RDF's first working draft was completed in 1997. (Powers, 2003) This first draft introduced the idea of using Uniform Resource Locators or URIs to identify grammatical components within a statement. URIs are split by a color into two parts: the URI scheme and the scheme-specific resource. (Geroimenko, 2004) A common example of this is a web address where "http" is the URI scheme and the "//www.website.com/" is the resource location.

Resources are one of the basic elements of RDF's description structure, often referred to as a "triple." The other two major pieces are Properties and Values. Altogether, they form Statements. Statements are often broken up using the terms subject, predicate, and object though, which aligns better with the linguistic approach to ontological grammar. (Bray, 2001) Here is an example of an RDF description code block:

```
<rdf:Description about='http://www.strobotron.org'>
  <Author>Erhardt Graeff</Author>
</rdf:Description>
```

In this example the statement reads, "The Author of http://www.strobotron.org/ is Erhardt Graeff" where the subject or resource is http://www.strobotron.org/, the predicate or property is Author, and the object or value is Erhardt Graeff.

The RDFCore Working Group was chartered by the W3C in 2001 to refine RDF into the modeling and syntactic language demonstrated above. The major work necessary for RDF to be a success directly relates to many of the specifications for a quality ontology outlined earlier—these include: interoperability, scalability, and overall flexibility in its vocabulary. (Bray, 2001) To allow for this functionality, a class system was needed to provide structure to the description statements. The RDF Schema or RDFS was that solution.

RDFS sits like a taxonomical and higher logic layer atop RDF, and essentially provides a backbone for using the language as a simple ontology. Implementing Object-

Oriented design methods, RDFS creates a classification system for RDF tags. A schema document is then able to group RDF statements into classes, and arrange them into subclasses of one another to form a taxonomy. RDFS tags are based on logic axioms which restrict the interpretation of the information, some examples are: subClassOf, isDefinedBy, and domain. (Fensel, Hendler, Lieberman, & Wolfgang, 2003) RDF documents are then considered instances of the each RDFS document, which allows for greater interoperability and retainable semantics over plain XML.

RDF and RDFS are limited however in their ability to create more complex definitions, ontological logic, and class structures. This led to RDFCore and OntoWeb working together toward resolving the more complete Semantic Web framework of RDF and OWL in 2004. To finish OWL required a more robust ontological language for the analysis—this was DAML+OIL. (W3C, 2001)

What is DAML+OIL?

DAML (DARPA Agent Markup Language) and OIL (Ontological Interchange Language) actually started life as two separate ontology projects. DAML was a project sponsored by DARPA in order to create a more rigorous ontology for use in smart "agents"—referring back to the AI origins of ontologies—and future Semantic Web applications. Essentially, the government sponsored the DAML-ONT project, a committee staffed with W3C members such as Tim Berners-Lee, for creating a language which contained advanced implementations of logic axioms (i.e. predicate calculus) for use in domains RDF and RDFS could not handle. (Fensel et al., 2003)

OIL, sometimes referred to as "Ontology Inference Layer," was a European-based initiative before its inclusion in the DAML+OIL project. The original missions of OIL were very similar to DAML from its inception, foreshadowing their later mergence.

The program was striving for a maximally expressive ontology which would be fully compatible with the existing technologies of XML and RDF, including an ability to interpret information from a wide variety of domains. (van Harmelen & Horrocks, 2000)

After seeing the related projects working concurrently, but separately, the DAML and OIL groups merged to pool their resources and form a better overall ontology for the Semantic Web. While the W3C was never formally a part of the project, they still have authored several notes outlining the combined efforts of the DAML+OIL language release. (Ouellet & Ogbuji, 2002a)

Each language, DAML and OIL, brought contributions to the final product. DAML provided strong axiomic logic, while OIL contributed well-defined datatyping as part of the ontology. Much of the flexibility provided by DAML+OIL comes from its extension of the RDFS class system with a more complete Object-Oriented approach. DAML+OIL allows collections and new levels of abstraction for classes and objects, as well as complicated logic relationships between classes. (Ouellet & Ogbuji, 2002b) Here is an example:

```
<daml:Class rdf:ID="CoffeeShop">
  <rdfs:label>Coffee Shop</rdfs:label>
  <rdfs:comment>A place offering light fare and occasional
entertainment</rdfs:comment>
  <daml:intersectionOf parseType="daml:collection">
    <daml:Class rdf:about="#Restaurant"/>
    <daml:Class rdf:about="#EntertainmentVenue"/>
  </daml:intersectionOf>
</daml:Class>
```

In this situation, a class for coffee shops is being defined. Coffee shops can be both places to eat and places to enjoy music and poetry readings; thus, coffee shops are not just a restaurant or just a venue for entertainment. So a DAML+OIL collection is formed based on an intersecting union between the classes of Restaurant and EntertainmentVenue for the class CoffeeShop.

The previous example showed that RDF/RDFS is still used for description, but how DAML+OIL's extensions allow for a more expressive portrayal of the semantics involved in a particular domain of knowledge. But as flexible as DAML+OIL is for interpreting situations on a wide gamut of applications, it is not designed in a manner to bridge all gaps associated with the World Wide Web. Something engineered solely for the problems facing Semantic Web development is needed to make Berners-Lee's visions into realities—that is why we now have OWL.

What is OWL?

OWL or Web Ontology/Ontology Web Language was the first completely W3C sponsored Semantic Web ontology project. WebOnt was the group for the project and they began by evaluating the DAML+OIL model and its potential for being a web ontology language candidate. This process enabled WebOnt to better understand the requirements of a comprehensive Semantic Web ontology through analysis of DAML+OIL's issues and shortcomings. (Daconta et al., 2003)

This analysis produced one of the six documents submitted for W3C recommendation in February of 2004, specifically the "OWL Use Cases and Requirements." (W3C, 2004b) This document defined exactly what OWL must be in order for it to fully realize itself as a web ontology. Among the requirements is a redefinition of what makes a well-designed ontology, extending Grubin's five principles outlined earlier to eight new specific goals. The goals are:

- using established ontologies
- changing established ontologies
- integrating established ontologies
- detecting inconsistencies across ontologies and instances of use
- providing a balance between expressivity and scalability when creating ontologies
- avoiding unnecessary complexity which may discourage widespread adoption
- maintaining compatibility with other standards
- supporting internationalization (W3C, 2004b)

A large focus in this list is thematically related to ensuring OWL works with all other existing web technologies. For the Semantic Web to work, an ontological framework must exist that can read and employ various protocols and standards already in use. Another important focus is on having the language as accessible to users as HTML is, thus they have worked toward lessening the amount of formal sounding tags.

This transition to a more user-friendly ontology forms the basis for the transition from DAML+OIL to OWL. One example of this is simply rewording tags like daml:equivalentTo to owl:sameAs. Other measures taken to simplifying the language refer to the analysis of DAML+OIL, during which they found many synonymous tags shared between DAML and RDF/RDFS; thus, such redundancies were eliminated. Another obligatory improvement was the addition of changes to RDF/RDFS since the publishing of DAML+OIL, propagating OWL's cutting edge interoperability. (Daconta et al., 2003)

To make OWL easier to implement for a variety of users, it has been defined in three different versions: OWL Lite, OWL DL, and OWL Full. "OWL Lite enables a user to define an ontology of classes and properties and the instances (individuals) of those classes and properties." (Daconta et al., 2003, Chapter 8, OWL Lite, para. 1) This allows you to quickly construct a simple description framework with minimal logic—assuming complex constructions are not necessary. Here is an example:

```
<owl:Class rdf:ID="Bongo">
  <rdfs:subClassOf rdf:resource="Drum" />
</owl:Class>
```

This simple application defines that Bongo is a subclass of the class Drum, something possible in earlier version of both RDFS and DAML+OIL. OWL DL (Description Logic) extends OWL Lite by adding complex cardinality and Boolean relationships, as well as relational restrictions by value. This improves the flexibility of a user's ontological design while still guaranteeing computational processing of your code. Here is an example:

```
<owl:Class rdf:ID="Bongo">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#madeOfWood" />
      <owl:hasValue rdf:resource="#WoodShell" />
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

In this application, a Bongo is said to have a "wood shell" and it will only be a subclass of madeOfWood if that class contains a value of WoodShell somewhere in it. Finally, OWL Full has all the same language constructions as OWL DL but differs by extending flexibility in logic and relationships to whatever the user desires. A designer can freely mix in RDF and RDFS, as well as let collections act as individuals. It is engineered to maximize a user's expression of a domain, but at the cost of assured computation. (W3C, 2004c)

Where is the Semantic Web?

Alas, the world is still waiting for the Semantic Web. There are small sprouts of hope though found in groups like the W3C, commercial research labs, and academic institutions, who are working toward slowing building applications using the ontological framework laid out by RDF and OWL. A few examples are Ontaria, Jena, and Longwell.

Ontaria is a W3C project principally investigated by Tim Berners-Lee. It aims to be a browsable directory of Semantic Web data. Well-formed RDF vocabularies, especially those using OWL ontologies, are added to the directory on a continual basis for public reference. Hopefully, this resource will inspire new semantic entrepreneurs, so that more and more information on the web is encapsulated in semantically-navigable ontological frameworks. (W3C, 2004a)

Jena is an opensource project derived from initial research by HP Labs Semantic Web Programme. Jena acts as programming API and framework for interfacing Java with RDF and OWL. The beauty in this pairing is in how fundamentally similar Java and OWL are in their pursuits, considering Java's platform-independency and OWL's domain-independency and overall interoperability. Hopefully, the widespread use of this API among Java developers will lead to them more readily adopting Semantic Web ontologies in dealing with data. ("Jena – A Semantic Framework," n.d.)

Finally there is Longwell, an excellent representative of a Semantic Web "killer app." This application is the output of the SIMILE project at MIT which brings together MIT's CS labs, the W3C, HP, and MIT Libraries. Essentially, Longwell is the first web-based Semantic Web browser. It is written in Java using the Jena API. Its implementation is as a portal to the digital libraries hosted by MIT, which makes a natural choice for ontological testing grounds given the normally-careful metadata construction which goes into library records. ("Longwell," 2004)

Though these projects offer impressive applications of ontological technology, they are obviously not the Semantic Web—perhaps forerunners—but not the Semantic Web. Berners-Lee, ye olde pioneer, gave an address at the recent WWW2004 "'People ask,' he said, 'so what's the Semantic Web killer app going to be? That's not the right question. The real proof of the Semantic Web,' he said, is when new connections are made, and new links between information emerge.'" (Ford, 2004) For these essential information links to become a reality, there needs to be a common push from academia, industry, and web users. Academia needs to continue to work on researching Semantic Web implementations. Industry needs to take academia's research, realize its value in their company, and apply it; thereby, forcing employees to work with technologies like OWL. And web users need to keep evolving as web users. Whether they are building their own sites, playing with XML technologies, or just wishing that search engines would get it right the first time, users need to make their desire of a Semantic Web ever-more visible. Ontologies have laid a formidable framework. The next step is having the world, and its web, use it.

References

Berners-Lee, Tim (1998). *Semantic web road map*. Retrieved on December 12, 2004 from http://www.w3.org/DesignIssues/Semantic.html.

Bray, Tim (2001). *What is RDF?*. Retrieved on December 12, 2004 from http://www.xml.com/pub/a/2001/01/24/rdf.html.

Clark, K.G. (2002). *If ontology, then knowledge: catching up with WebOnt*. Retrieved on December 12, 2004 from http://www.xml.com/pub/a/2002/05/01/webont.html.

Daconta, M.C., Obrst, L. J., & Smith, K. T. (2003). *The semantic web: A guide to the future of XML, web services, and knowledge management*. Indianapolis: Wiley Publishing, Inc.

Dumbill, E. (2000). *The semantic web: a primer*. Retrieved on December 12, 2004 from http://www.xml.com/pub/a/2000/11/01/semanticweb/.

Fensel, D. (2004). *Ontologies: a silver bullet for knowledge management and electronic commerce*. Berlin: Springer.

Fensel, D., Henderl, J., Lieberman, H., & Wahlster, W. (2003). *Sprinning the semantic web: Bringing the world wide web to its full potential*. Cambridge: The MIT Press.

Ford, P. (2004). *Berners-Lee keeps WWW2004 focused on semantic web*. Retrieved on December 12, 2004 from http://www.xml.com/pub/a/2004/05/20/www-timbl.html.

Geroimenko, V. (2004). *Dictionary of XML technologies and the semantic web*. London: Springer.

Gómez-Pérez, A., Fernández-López, M., & Corcho, O. (2004). *Ontological engineering: With examples from the areas of knowledge management, e-commerce and the semantic web*. London: Springer.

Jena – A semantic web framework for java. (n.d.). Retrieved on December 12, 2004 from http://jena.sourceforge.net/.

Longwell. (2004). Retrieved on December 12, 2004 from http://simile.mit.edu/longwell/index.html.

Ouellet, R. & Ogbuji, N. (2002a). *Introduction to DAML: Part 1*. Retrieved on December 12, 2004 from http://www.xml.com/pub/a/2002/01/30/daml1.html.

Ouellet, R. & Ogbuji, N. (2002b). *Introduction to DAML: Part 2*. Retrieved on December 12, 2004 from http://www.xml.com/pub/a/2002/03/13/daml.html.

Powers, S. (2003). *Practical RDF*. Sebastopol: O'Reilly & Associates, Inc.

van Harmelen, F. & Horrocks, I. (2000). *Questions and answers on OIL: The ontological inference layer for the semantic web*. Retrieved on December 12, 2004 from http://www.ontoknowledge.org/oil/oil-faq.html.

World Wide Web Consortium (1996). *Extensible markup language (XML)*. http://www.w3.org/TR/WD-xml-961114.html.

World Wide Web Consortium (2001). *Semantic web*. Retrieved on December 12 from http://www.w3.org/2001/sw/.

World Wide Web Consortium (2004a). *About Ontaria*. Retrieved on December 12, 2004 from http://www.w3.org/2004/ontaria/static/about.

World Wide Web Consortium (2004b). *OWL web ontology language: Use cases and requirements*. Retrieved on December 12, 2004 from http://www.w3.org/TR/webont-req/.

World Wide Web Consortium (2004c). *OWL web ontology language guide*. Retrieved on December 12, 2004 from http://www.w3.org/TR/2004/REC-owl-guide-20040210/.