The Semantic Web: Fundamentals and A Brief State-of-the-Art

Luis Sánchez-Fernández and Norberto Fernández-García

In this article we present an overview of the Semantic Web. We look at the some of the problems facing the Web today and by way of a possible solution we propose an evolution of the current Web whereby the content available on the Web would have associated formal, machine-readable descriptions. We describe the fundamental components of the Semantic Web and its state-of-the-art: semantic annotation, ontologies, and logical reasoning.

Keywords: Metadata, Ontology, RDF, RDF Schema, OWL, Semantic Annotation, Semantic Web.

1 Introduction

The World Wide Web was created as recently as 15 years ago and yet it in that short time it has brought about an astonishing revolution. It has allowed people all over the world to exchange information to a degree that had previously been impossible.

In spite of its great success, the Web as it is today has a number of limitations which we would like to address in order to achieve a higher level of automation in its use. A typical example is the use of search engines. The most common way to find information on the Web is by using search engines such as Google; we can type some words in a box and it will return a number of relevant web pages. Although search engines usually work well, anyone who regularly uses one will know that they do not always deliver the goods. On occasion either or both of the following will occur:

- We find web pages that are not related to the object of the search. For example, we wanted to find information on the Web about the "International Semantic Web Conference" to take place in Galway (Ireland) in November 2005 and so we queried Google for "ISWC" (the query was made on October 27, 2005). The first hit referred to the 2004 "International Semantic Web Conference" which was not what we were looking for, while the second was for the Web page for the 2005 "International Symposium on Wearable Computers". The first page relevant Web page related to what we were actually looking for (in fact it was the home page of the conference) came 6th in the list of hits returned by Google.
- The search engine does not find some Web important pages related to what we are interested in. For example, we also queried Google for "Computer Architecture" + "Carlos", (the query was made on November 2, 2005) to try and find information about a course forming part of the Telecommunications Engineering degree studies at the *Universidad Carlos III de Madrid* (UC3M). The two main pages about this course posted on UC3M's Web server of the UC3M were not among the first 50 hits first shown by Google.

There are also some things that you simply cannot do automatically on today's Web. For example, it would be nice to be able to ask a search engine, say, "tell me the name of the President of France". All we can do at the moment is search for "President of France" in Google, which will give us a number of web pages where we can find the answer to our question, but will not specifically answer the question. It is somewhat annoying to have to go through this two-step procedure. After all, the information is there; we can't it be retrieved by the search engine directly? At times we need to make a number of searches and look through several web pages before we find what we are looking for.

2 Syntax vs. Semantics

Search engines like Google retrieve documents that contain the we have put in the query box (Google itself also uses links pointing to a particular Web page to decide whether or not to retrieve that page). For example, in our

Luis Sánchez-Fernández graduated as a telecommunications engineer from the *Universidad Politécnica de Madrid*, Spain, in 1992 and received his doctorate in Telecommunications Engineering, from the same university in 1997. In October 1997 he joined the *Universidad Carlos III de Madrid* where he is currently a full professor in the Dept. of Telematic Engineering, holding the post of Assistant Director. He is Director at Laboratorio de Tecnologías Web, <http://www.it.uc3m.es/infoflex/techweb/es/>, which forms part of the research group *Grupo de Aplicaciones y Servicios Telemáticos* (Telematic Applications and Services Group) of the *Universidad Carlos III de Madrid*. He has participated and/or led a number of national research projects and one European project related to web technologies, including Semantic Web technologies, and has authored more than 50 publications in national and international conferences and journals as well as a number of chapters in scientific books. His current research activities are focused on the Semantic Web (semantic annotation, ontologies, semantic Web services). He is also interested in other technologies related to Web applications, such as XML. He is a member of the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*) and a frequent contributor to its journal ***Novática***. <luiss@it.uc3m.es>

Norberto Fernández-García graduated in Telecommunications Engineering from the *Universidad de Vigo*, Spain,in 2002. He currently works as an assistant lecturer at the Higher Politechnical School of the *Universidad Carlos III de Madrid*, Spain, where he is studying for his doctorate. His research is currently centred on matters related to the Semantic Web and, in particular, semantic annotation, knowledge management in peer-to-peer systems, and the application of Semantic Web technologies to journalism. <berto@it.uc3m.es>

query for "ISWC" we found web pages containing the term "ISWC". This technique (together with a great many technical details that we will not go into here) works quite well for retrieving documents related to a query. However, as we have seen, in some cases the results of the query are not correct (we found web pages that were unrelated to what we were looking for or we failed to find some particularly relevant pages). There are at least three kinds of situations that can lead to such errors:

- Polysemy: we search for a term and find web pages containing that term but with a different meaning from the one we are interested in.
- Synonymy: we search for a term and instead of finding pages related to what we are interested in we find web pages containing a synonym of the term (but not the term itself).
- Multilingualism: we search for a term in English and we find pages related to what we are interested in but they are written in Spanish or some other language.

In all these cases, the problem is that the query does not precisely identify what we are interested in but only identifies a term which in a given language has amongst its various meanings the meaning we are looking for. A search procedure in which the result of the search is a set of documents containing the terms we typed into the query box is called a "syntax search".

If web pages had formal annotations attached to them which unambiguously identified the main concepts and entities they contain, a (semantic) search engine could avoid making the three types of errors described above and only find documents precisely related to what we were actually interested in when we made the query.

The Semantic Web is a vision of a future Web in which information that in the current version of the Web is understandable only by humans will also be machine-readable. If this vision comes to fruition in the future, it will open the door to new Web-based applications, ranging from semantic search engines to intelligent agents capable of browsing the Web and making a doctor's appointment for us [1].

3 Fundamentals of The Semantic Web

As we have just seen, under the Semantic Web, pages will have annotations (inside the pages themselves or elsewhere) which formally describe the information (knowledge) contained in those pages. Semantic Web applications will make use of that knowledge to help in the automation of Web tasks that are currently performed with heavy user interaction (following links, form-filling, combining data from various sources, deciding on the next step to take, making searches, etc.).

Knowledge-based systems of the kind that will be developed for the Semantic Web have been studied by researchers into Artificial Intelligence for some years. As readers will know, these systems have two basic components: a knowledge base of facts known by the intelligent system and an inference engine. As we will see, knowledge-based applications for the Semantic Web will also have these two elements although there will be some differences with respect to classical knowledge-based systems.

The knowledge base for a Semantic Web application will be built from annotations collected by the application as it browses (semantic) web pages. The open nature of Web architecture advises against the use of traditional knowledge representation languages, because they presuppose the existence of a common vocabulary shared by all the users of the system. However, in an open environment such as the Web such a supposition cannot be made because different user groups must be able to use their own vocabularies. On the Semantic Web, anyone will be able to annotate web pages and these annotations will not be limited to a fixed vocabulary.

This requirement has prompted the World Wide Web Consortium (W3C) [2] to define a standard recommendation for representing knowledge on the Semantic Web: the RDF (Resource Description Framework) [3].

An RDF model is made up of a set of triples in which each triple comprises a subject (the resource being described: a Web page, a person, a location or anything else that may be of interest), a predicate identifying the characteristic defined for that resource (the author of the Web page, his work, the coordinates of the location, ...) and an object indicating the value of the predicate for the specified subject (for example the Web page author's name is "Luis Sánchez"). Resources in RDF are identified by Universal Resource Identifiers, URIs (URLs are the most common type of URIs). The subject and predicate of an RDF triple are URIs. Anyone can identify new resources to describe and new properties of resources simply by assigning a new URI to them. This fits in very well with the open architecture of the Web. Objects can be URIs, but they can also be constants (a string of characters, an integer, ...).

Semantic Web applications may need more than semantic annotations. Most of these applications will need a model of the domain they are operating in, which includes a vocabulary of the concepts relevant to that domain and probably the properties relating the different concepts and the rules governing the domain. This model will enable the system to draw conclusions and/or take decisions by processing the annotations extracted from browsed web pages. These models are defined by means of the second essential ingredient needed to make the Semantic Web a reality: ontologies.

Although there are a many definitions of what an ontology is, we will borrow a definition coined by Studer et al. [4] based on the previous definitions by Gruber [5] and Borst [6]: "*An ontology is a formal and explicit specification of a shared conceptualization*".

The elements of this definition are also explained in [4] as follows:

- **Conceptualization** refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon.
- **Explicit** means that the types of concepts used and the constraints on their use are explicitly defined.

■ **Formal** refers to the fact that the ontology should be machine-readable.

■ **Shared** reflects the notion that an ontology captures consensual knowledge; that is, it is not private to some individual but accepted by a group.

It may seem a little confusing that we are now talking about Semantic Web applications that store vocabularies describing concepts and properties in their knowledge bases while previously we had said that anyone can create new terms (URIs) to identify resources (remember that resources can be the subject, object, or predicate of an RDF triple). How can a Semantic Web application 'understand' the meaning of an annotation if it does not use a vocabulary known by the application? The answer is simply that it cannot. This is where the idea of "shared conceptualization" of ontologies comes in. A useful ontology will be shared by a group of people (or a group of organizations) and will be used to annotate their web pages. The Semantic Web application storing that ontology in its knowledge base will be able to understand the annotations of such web pages. This actually is not so very different from the state of the Web today. A Spanish person will find it difficult to understand web pages written in German (unless he or she happens to speak German) and a doctor can easily understand web pages related to medicine but may have problems understanding web pages related to complex electronic technologies.

With regard to this, one of the most active lines of research into the Semantic Web today is concerned with what is known as ontology mapping. Ontology mapping addresses the problem of detecting that two concepts defined in two ontologies are related to one another in some way or even that they are actually the same concept. There are several techniques that can be used to do this: by analysing the ontology structures, by natural language processing techniques if the concepts have associated labels, etc. Thanks to ontology mapping we can discover relationships between the vocabulary (the ontology) used by our knowledge-based system and other vocabularies, thereby enabling us to extend the range of web pages that our system is capable of processing. Unfortunately, the problem of ontology mapping is far from being solved (at least not automatic mapping).

As in the case of RDF, the W3C has developed two standard recommendations for the development of ontologies for the Semantic Web: RDF Schema [7] and OWL (*Web Ontology Language*) [8].

The third fundamental element of the Semantic Web is the development of inference engines to implement the semantics of the ontology and annotation languages used. In order to be of use, these inference engines will need to combine powerful reasoning capabilities with scalable performance.

The W3C has proposed an architecture (**Figure 1**) for the Semantic Web based on the three fundamental elements we have just described: annotations, ontologies and inference engines.

The lower levels of Semantic Web architecture reflect the fact that the RDF models can be encoded as XML

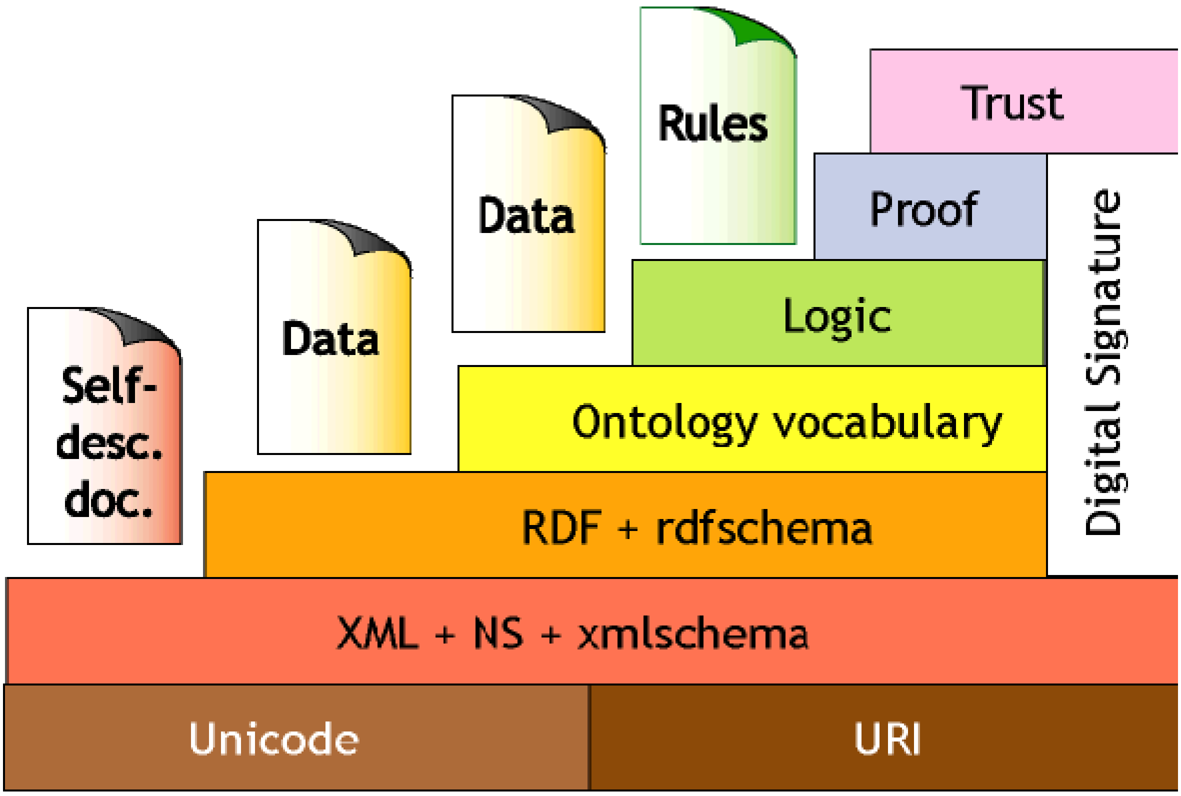

Figure 1: Semantic Web architecture.

(eXtensible Markup Language) documents and that the RDF Schema and OWL ontology languages are RDF models. It is interesting to note that digital signature technologies also appear in the figure. Due to the open nature of the Web, annotations will follow the philosophy of: "*anyone can say anything about anything*". A Semantic Web application will need to know how reliable the annotations it collects are. New models for security and trust on the Semantic Web will need to be developed.

The next three sections of this article deal with the state-of-the-art in research activities related to annotation, ontologies, and reasoning for the Semantic Web. The great variety of research activities being conducted today, the zeal with which the research community is addressing the challenge of the Semantic Web, and the obvious constraints of space of an article of this kind, means that our overview will be of necessity incomplete. For each topic we will try and mention some of the most important papers on the subject, but others will inevitably be left out. We will finish off with some conclusions and our view of how the Semantic Web will develop in the near future.

4 Introduction to Semantic Annotation

As we have suggested in previous sections, the Semantic Web [1] can be seen as an extension of the current day Web in which the information contained on web pages must be represented formally (as well as in natural language) to enable machines to *understand* that information and process it intelligently.

In order to give web resources a machine-readable formal description and thereby realize the Semantic Web vision, we need to add *semantic metadata*[1] to those web resources. The process of adding semantic metadata to web resources is commonly known in the state-of-the-art as *semantic annotation*[2] .

Generally speaking, the semantic annotation of a web resource (a web page, an image, an email, …) needs to associate its entire content or part of it (for example, a particular word) with a given identifier. This identifier (normally a URI) unambiguously identifies a concept mentioned in the content of the resource. The identifiers of the concepts to be used in the annotation process are obtained from conceptualizations of certain domains of knowledge, typically in the form of ontologies.

Since annotations are the key to automatic processing of web pages, semantic annotation is of vital importance if we are to make the Semantic Web a reality. As a result, annotations have been, and will continue to be, an important field of Semantic Web research. In the state-of-the-art of semantic annotation we can find dozens of approaches that can be broken down in various ways, such as by the degree of automation of annotation tasks, or by the kind of web resources that can be annotated. Looking first at the degree of automation, we can break the proposed tools and systems down into:

Manual annotation systems: these include systems such as Annotea [9], SHOE Knowledge Annotator [10], SMORE (*Semantic Markup, Ontology and RDF Editor*) [11] or CREAM (*Creating RElational, Annotation-based Metadata*) [12]. These systems typically provide a user interface allowing human annotators to view and browse both ontologies and web resources simultaneously, using the knowledge modelled in the ontologies to add annotations to web resources. On the subject of approaches featuring manual annotation we should also mention proposals that aim to extend content authoring tools to include semantic annotation capabilities, such as Semantic Word [13].

(Semi-)automatic annotation systems: these include systems such as AeroDAML [14], SemTag [15], S-CREAM [16], PANKOW [17], C-PANKOW [18], KIM [19] or MnM [20]. These systems basically make use of NLP (Natural Language Processing) techniques to extract references in the text to certain concepts described in ontologies. These systems generally require the input of seed patterns or corpuses of documents in order to train the system.

With regard to the kind of resources that can be annotated, most of the state-of-the-art approaches are centred on the annotation of text resources. This is true of most of the systems mentioned previously, especially those using NLP techniques. However, recently there ahs been a growing interest in the annotation of non-text resources, and so it is no surprise to find among our semantic annotation state-of-the-art approaches proposals for the annotation of images [21], audio [22] and multimedia [23]. We can also find approaches where the information to be annotated is not the content of a web resource, but a service [24]. Thus we could say that semantic annotation is following the trends in Semantic Web research by not only covering aspects of the classical Web but also more innovative aspects such as Web Services or GRID services (*Global Resource Information Database*).

5 Introduction to Semantic Web Ontologies

An ontology is made up of the following parts:

- Classes and instances: for example, an ontology modelling the political structure of the European Union may contain classes such as "Country" or "Political Party". Instances are used to model elements or individuals. Usually instances belong to classes. For example, the instance "Spain" belongs to the class "Country". Classes are usually organized in a hierarchy of subclasses. If class A is a subclass of class B, instances of class A also belong to class B.
- Properties: they establish relationships between the concepts of an ontology. For example, the property "isPartyMember" associates a person with the party he or she belongs to.
- Rules: They model logical sentences that are always true. They are commonly used for modelling knowledge that cannot be represented using the other three ontology elements. Rules can be used for three purposes:

[1] From the Greek *semantikos*, meaning, from the Greek *meta*, 'beyond', and the Latin *data*, 'data': data about the meaning of other data
[2] From the Greek *semantikos*, meaning, and from the Latin *annotare*, 'to add a note': to add a note to explain the meaning of a symbol

- Knowledge creation: to obtain new logical sentences based on information stored in the knowledge base.

- Constraint definition: define properties that the model should met. They are used to detect inconsistencies.

- Reactive rules: determine actions to be taken by a knowledge-based system as a result of certain conditions being met.

Ontologies having no rules are called lightweight ontologies. The simplest type of lightweight ontology are called taxonomies and they are made up of a hierarchy of classes representing the relevant concepts in the domain.

The decision of whether or not to use rules should not be taken lightly, since while rules provide us with high expressiveness they also make for more complex reasoning with the ontology, which can give rise to scalability issues.

The development and maintenance of complex ontologies requires new techniques and tools to be developed. This is the purpose of what is known as "ontology engineering". Within the scope of "ontology engineering" there are currently a number of research activities in progress, among which are:

- Methodologies for the development of ontologies. There are a several different proposals for defining appropriate methods and methodologies for the development of ontologies. Among the most important are METHONTOLOGY [25], and On-To-Knowledge [26].
- Tools supporting the development of ontologies. Most of these tools include an ontology editor with a user-friendly graphical interface that allows users to view the class hierarchy of the ontology and insert/modify/delete classes, properties, instances and/or axioms. Some of them also provide support to other phases of the ontology life-cycle, such as evolution, documentation, evaluation, etc. Among the most widely used ontology editors that support RDF, RDF Schema and/or OWL are Protegé [27], InferEd [28], WebODE [29] and OilEd [38].
- Evaluation and quality measures of ontologies. As in other types of engineering, in ontology engineering we need to evaluate the ontologies that we are going to use. This evaluation includes a number of different aspects, such as checking that the ontology is consistent (that it cannot reach contradictory conclusions), that it models the domain properly, that it contains no redundancies, that it is easy to maintain, etc. Some relevant approaches to ontology evaluation are to be found at [30] and [31].
- Support for ontology maintenance and evolution. As in any other software system, Semantic Web-based applications will evolve over time, as will the ontologies that they use.

We should also mention research activities related to ontology learning, whereby ontologies are built (semi-)automatically using various techniques (for example, NLP). The results obtained by these systems normally need to be debugged manually before they can be used.

6 Reasoning Tools and Rule Languages for The Semantic Web

One of the activities currently being carried out in the field of the Semantic Web is the development of inference engines that support the languages being defined for the Semantic Web. Examples of these are Jena [32], a development environment for Semantic Web applications that includes for RDF Schema and OWL Lite reasoners (OWL Lite is an OWL subset). Another RDF development environment that incorporates an inference engine is SESAME [39].

A very important subset of OWL is OWL-DL, in which DL means *Description Logics,* a type of logic with less expressiveness than first-order predicate logic, but one over which it is possible to perform reasoning tasks more efficiently. Pellet [33] is an example of an OWL-DL reasoner.

The RDF Schema and OWL ontology languages have limited capabilities for logical reasoning. For this reason, a number of proposals for rule languages for the Semantic Web have been put forward. One of the first proposals was TRIPLE [34]. TRIPLE is a rule language based on Horn clauses on top of RDF and an inference engine capable of reasoning about models defined in TRIPLE.

Another very important initiative is RuleML (Rule Markup Language) [35]. RuleML is an XML language for defining rules intended for the Semantic Web RuleML contains aspects logical programming, functional programming (the subject of an article in this monograph) and object orientation. The designers of RuleML have submitted a language called SWRL (Semantic Web Rule Language) [36] for consideration as a W3C standard. SWRL is a combination of OWL-DL (an OWL subset) and a subset of RuleML.

Regardless of the languages and tools being developed, an important issue when designing a system based on Semantic Web technologies is exactly how logical reasoning capabilities are to be used. As readers will know, most logical reasoning languages with high expressiveness are characterized by exponential complexity (as is the case of OWL-DL) or are undecidable (as in the case of first-order predicate logic). As a result, the use of inference engines for large scale knowledge bases is not feasible today, nor does it seem likely that it will be feasible in the future.

To overcome this problem a number of approaches are open to us, all of which involve limiting the use of inference engines in our systems. One frequently adopted solution is to use lightweight ontologies combined with a reasoner to expand class hierarchy queries. The expanded query is finally processed by a conventional relational database which stores the instances. This type of reasoning can be implemented efficiently and therefore has no scalability issues.

Other approaches are based on "approximate reasoning"; that is, reasoning algorithms that do not return 100% of the results but perform much better.

7 Conclusions

In this article we have presented an overview, necessarily incomplete for reasons of space, of the state-of-the-art of the Semantic Web. It is incomplete in two aspects: 1) we have omitted references to some important work in this field and 2) we have left out a number of important Semantic Web related research activities. A truly exhaustive overview would, in any case, be impossible due to the huge volume of research work going on in this area. However, we did not want to conclude this article without at least mentioning two important activities: Semantic Web Services and Semantic Web query languages.

Web Services is a middleware technology promoted by the W3C for communication between applications using the Web. The Web Services standards that are being developed by the W3C provide mechanisms for defining the interfaces provided by servers to clients wishing to use a given Web Service. However, they do not provide any way to formally define either the tasks that can be performed with a particular Web Service, or the semantics of the data to be exchanged between client and server. The idea behind Semantic Web Services is to provide formal descriptions to services to enable automatic discovery, access and composition of Web Services. References to some important work in the field of Semantic Web Services are to be found at [40][41][42][43][44].

Query languages are used to formally represent the information we wish to obtain from a knowledge base (or a conventional database). In the case of the Semantic Web, there are a number of proposals for RDF query languages, the most promising of which is probably SPARQL [37], which is currently a W3C Working Draft.

The Semantic Web vision is a source of great controversy among its supporters and detractors. The latter believe that it will never become a reality. However, applications and prototypes based on Semantic Web technologies are beginning to appear.

At this point in time it is hard to gauge to what degree this vision will eventually be implemented or if and when its use and deployment will become widespread. Clearly the first requirement for this to happen is for a significant number of web pages to be annotated. We believe that before this requirement is met, existing semantic annotation systems need further development and more applications promoting their use need to appear.

The development of a Semantic Web with all its possibilities, as described in [1], is probably still very far off, if indeed it ever happens. However, we are convinced that in the near future the technologies being developed will make for a more automated, more user –friendly, and more efficient use of the Web.

*Translation by **Steve Turpin***

References

[1] T. Berners-Lee, J. Hendler, O. Lassila. "The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities". *Scientific American*, May 2001.

[2] World Wide Web Consortium (W3C), <http://www.w3.org> [2-11-2005].

[3] Resource Description Framework (RDF). W3C Semantic Web Activity. <http://www.w3.org/RDF/> [2-11-2005]. Contains links to all W3C Recommendations on RDF.

[4] R. Studer, V.R. Benjamins, D. Fensel. "Knowledge Engineering: Principles and Methods". *IEEE Transactions on Data and Knowledge Engineering* 25(1-2), pp 161-197, 1998.

[5] T.R. Gruber. "A Translation Approach to Portable Ontology Specification". *Knowledge Acquisition* 5(2), pp 199-220, 1993.

[6] W.N. Borst. Construction of Engineering Ontologies. Centre for Telematica and Information Technology, University of Twente. Enschede. The Netherlands, 1997.

[7] RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation 10 February 2004. <http://www.w3.org/TR/rdf-schema/> [2-11-2005].

[8] Web Ontology Language (OWL). W3C Semantic Web Activity. <http://www.w3.org/2004/OWL/> [2-11-2005]. Contains links to all W3C Recommendations on OWL. .

[9] J. Kahan, M-R. Koivunen, E. Prud'Hommeaux, R.R. Swick. Annotea: An Open RDF Infrastructure for Shared Web Annotations. In WWW10 Conference, Hong Kong, May 1-5, 2001.

[10] SHOE Knowledge Annotator. Available at <http://www.cs.umd.edu/projects/plus/SHOE/KnowledgeAnnotator.html>.

[11] A. Kalyanpur, J. Hendler, B. Parsia, J. Golbeck. SMORE - Semantic Markup, Ontology, and RDF Editor. Available at <http://www.mindswap.org/papers/SMORE.pdf>.

[12] S. Handschuh, S. Staab. Authoring and Annotation of Web Pages in CREAM. Proceedings of the 11th International World Wide Web Conference, WWW 2002, Honolulu, Hawaii, May 7-11, 2002. ACM Press.

[13] M. Tallis. Semantic Word Processing for Content Authors. Second International Conference on Knowledge Capture (K-CAP 2003), Sanibel, Florida, USA, 2003.

[14] P. Kogut, W. Holmes. AeroDAML: Applying Information Extraction to Generate DAML Annotations from Web Pages. First International Conference on Knowledge Capture (K-CAP 2001). Workshop on Knowledge Markup and Semantic Annotation, Victoria, B.C. October, 2001.

[15] S. Dill, N. Eiron, D. Gibson, D. Gruhl, R. Guha, A. Jhingran, T. Kanungo, S. Rajagopalan, A. Tomkins, J.A. Tomlin, J.Y. Zien. SemTag and Seeker: Bootstrapping the semantic Web via automated semantic annotation. WWW2003 Conference, Budapest, Hungary, May 2003.

[16] S. Handschuh, S. Staab, F. Ciravegna. S-CREAM: Semi-automatic CREAtion of Metadata Proceedings of the European Conference on Knowledge Acquisition and Management - EKAW-2002. Madrid, October 1-4, 2002.

[17] P. Cimiano, S. Handschuh, S. Staab. Towards the Self-annotating Web. In the 13th International World Wide Web Conference, WWW 2004, pp 462-471, New York, USA, May 17-22, 2004.

[18] P. Cimiano, G. Ladwig, S. Staab. Gimme' the Context: Context-driven Automatic Semantic Annotation with C-PANKOW. In the 14th International World Wide Web Conference, WWW 2005, Chiba, Japan, May 10-14, 2005.

[19] B. Popov, A. Kiryakov, A. Kirilov, D. Manov, D. Ognyanoff, M. Goranov. KIM, Semantic Annotation Platform. Proceedings of Second International Semantic Web Conference, ISWC 2003, LNCS 2870, pp 835-849.

[20] M. Vargas-Vera, E. Motta, J. Domingue, M. Lanzoni, F. Stutt, F. Ciravegna. MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup. In 13th International Conference on Knowledge Engineering and Management (EKAW 2002), Springer Verlag, 2002.

[21] J. Wielemaker, A.Th. Schreiber, B.J. Wielinga. Supporting Semantic Image Annotation and Search. In Annotations for the Semantic Web. Handschuh, S. and Staab, S. (Eds.), pp 147-155, IOS Press, 2003.

[22] P. Cano, M. Koppenberger. Automatic sound annotation. Machine Learning for Signal Processing, 2004. Proceedings of the 2004 14th IEEE Signal Processing Society Workshop, pp 391-400, Sept 29 – Oct 1, 2004.

[23] S. Bloehdorn, K. Petridis, C. Saathoff, N. Simou, V. Tzouvaras, Y. Avrithis, S. Handschuh, Y. Kompatsiaris, S. Staab, M.G. Strintzis. Semantic Annotation of Images and Videos for Multimedia Analysis. In Proceedings of the Second European Semantic Web Conference, ESWC 2005, LNCS 3532, pp 592-607, May 2005.

[24] A. Patil, S. Oundhakar, A. Sheth, K. Verma. METEOR-S Web service Annotation Framework. In Proceedings of the 13th International World Wide Web Conference, WWW 2004, pp 553-562, New York, USA, May, 2004.

[25] M. Fernández-López, A. Goméz-Pérez, N. Jurista. METHONTOLOGY: From Ontological Art Towards Ontological Engineering. Spring Symposium on Ontological Engineering of AAAI. Stanford University, California, 1997.

[26] S. Staab, H.P. Schnurr, R. Studer, Y. Sure. Knowledge Processes and Ontologies. IEEE Intelligent Systems 16(1), pp 26-34, 2001.

[27] Protegé, <http://protege.stanford.edu/> [22-12-2005].

[28] InferEd, <http://www.intellidimension.com/pages/site/products/infered/default.rsp> [22-12-2005].

[29] WebODE, <http://webode.dia.fi.upm.es/WebODEWeb/index.html>[22-12-2005].

[30] A. Gómez-Pérez. Evaluation of Ontologies. International Journal of Intelligent Systems 16(3), pp 391-409, 2001.

[31] C. Welty, N. Guarino. Supporting Ontological Analysis of Taxonomic Relationships. Data and Knowledge Engineering 39(1), pp 51-74.

[32] Jena, <http://jena.sourceforge.net/index.html> [22-12-2005].

[33] Pellet OWL-DL Reasoner, <http://www. mindswap.org/2003/pellet/index.shtml>, [22-12-2005].

[34] Michael Sintek, Stefan Decker: TRIPLE—A Query, Inference, and Transformation Language for the Semantic Web. International Semantic Web Conference (ISWC), Sardinia, June 2002

[35] RuleML, <http://www.ruleml.org/> [22-12-2005].

[36] SWRL: Semantic Web Rule Language. Proposal sent to W3C. <http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/> [22-12-2005].

[37] SPARQL Query Language for RDF. W3C Working Draft 23-11-2005. <http://www.w3.org/TR/rdf-sparql-query/>.

[38] OilEd, <http://oiled.man.ac.uk/> [23-12-2005].

[39] SESAME, <http://www.openrdf.org/> [23-12-2005].

[40] OWL-S: Semantic Markup for Web Services, <http://www.daml.org/services/owl-s/1.1/overview/> [23-12-2005].

[41] David Martin, Massimo Paolucci, Sheila McIlraith, Mark Burstein, Drew McDermott, Deborah McGuinness, Bijan Parsia, Terry Payne, Marta Sabou, Monika Solanki, Naveen Srinivasan, Katia Sycara, "Bringing Semantics to Web Services: The OWL-S Approach", Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition (SWSWPC 2004), July 6-9, 2004, San Diego, California, USA.

[42] Web Service Modeling Ontology, <http://www.wsmo.org> [23-12-2005].

[43] Dumitru Roman, Uwe Keller, Holger Lausen, Jos de Bruijn, Rubén Lara, Michael Stollberg, Axel Polleres, Cristina Feier, Christoph Bussler, and Dieter Fensel: Web Service Modeling Ontology, Applied Ontology, 1(1), pp 77 -106, 2005.

[44] R. Akkiraju, J. Farrell, J.Miller, M. Nagarajan, M. Schmidt, A. Sheth, K. Verma, "Web Service Semantics - WSDL-S, " A joint UGA-IBM Technical Note, version 1.0, April 18, 2005.