Mansoura Engineering Journal, (MEJ), Vol. 35, No. 3, September 2010. E. 1

Ongoing Trends Of Knowledge Representation For Semantic Web: Review Study

دراسة استعراضية للاتجاهات الحديثة في تمثيل المعرفة في الويب المفاهيمي

A. M. Riad
Head of information system dep.
Faculty of computers and information sciences.
Mansoura university, Egypt
amriad2000@yahoo.com

Hamdy K. Elminir
Head of communication dep
Misr Academy for Engineering & technology.
hamdy_elminir@hotmail.com

Mohamed Abu ElSoud
Computer Science Dept.
Faculty of computers and information sciences. Mansoura university, Egypt
mohamed_hossieny@yahoo.com

Sahar. F. Sabbeh
Assistant Lecturer
Alzarka Higher institute for administration & computer sciences
m_sabbeh@yahoo.com

ان شبكة الويب هي أغنى المصادر المعلوماتية لما تحتويه من معلومات ومصادر متنوعة يمكن الوصول إليها إلا أن قوتها الحالية تعتمد على قدرة المستخدم على التصفح والوصول لما يحتاجه من معلومات كما أن تنظيم المعلومات بصورة تسهل عملية البحث عنها والوصول إليها يعتبر أمرا غاية في الصعوبة. ويعد تصور الويب المفاهيمي أحد الحلول التي يتوقع أن تساعد على التغلب على هذا العيب وذلك من خلال تحويل محتوى الويب الحالي إلى محتوى يمكن فهمه من قبل الآلات وأجهزة الحاسب. ولدعم هذا التصور يجب أن تأتي صفحات الويب مع وصف لمحتواها مما يسهل عملية تفسيرها بشكل منطقي ويعد جمع هذه المعرفة وترتيبها بشكل يمكن تشغيله أتوماتيكيا أحد التحديات الكبرى ومن هنا تستخدم تقنيات تمثيل المعرفة وتحديدا وسوم الوصف "annotations" وخرائط المفاهيم "ontology" من أهم الوسائل التي تستخدم لدعم مشاركة المفاهيم بين المستخدمين والحاسبات. ولقد تم تطوير الكثير من الأدوات التي تساعد على توصيف المحتوى "annotation" وانتاج خرائط المفاهيم "ontology" في هذا البحث نقدم استعراض لتقنيات تمثيل المعرفة والأدوات التي تم تطويرها لدعم توصيف الويب وبناء خرائط المفاهيم موضحين التقدم والبحث الحالي.

Abstract:

The Web is the richest information repository, but most of the information it holds is unstructured. Moreover, its power is limited by the ability of human users to navigate the various sources for the information they require. To overcome this defect, semantic web vision is anticipated to be a promising path towards transforming the present web content into a machine understandable. To support this vision, augmenting web pages with description about the content they hold is necessary to make it possible to reason about it. A major challenge is to capture knowledge and structure it in a format that can be processed automatically. Therefore, knowledge representation techniques, namely annotation and ontologies, support sharing common understanding of a domain among people and computers. A large number of tools have been developed to help web content annotations and ontology creation. This paper presents a generic review of knowledge representation techniques and tools that have been developed to support annotation as well as ontology building showing advances and ongoing research.

Keywords:

Semantic web, knowledge representation, annotation, ontology

1. Introduction

Humans are capable of using the Web to carry out information retrieval tasks. However, a computer cannot

Accepted September 30, 2010.

accomplish the same tasks without human direction as web pages are designed to be read by people, not machines. Semantic web is a vision of information that is understandable by computers, so that they can perform more of the tedious work involved in finding, sharing, and combining information on the web. Trying to overcome usual defects concerning traditional web's failure to enable machine understandable processing of data, better representation of web data was a must to achieve semantic web vision. Knowledge representation techniques are used to provide an organized description and structured representation of web content identifying entities and relationships to better enable meaningful manipulation and retrieval of information. Annotation and ontologies are both used to support this notion forming the foundation for semantic web vision. Thus, markup languages and tools development was a must rather than a necessity.

In the next section we'll try to articulate the technologies upon which semantic web vision is based, section 3 reviews the advances in knowledge representation for semantic web exploring semantic annotation and ontology as a core component of semantic web technology. Conclusion is presented in section 4.

2. Semantic web stack

The development of Semantic Web proceeded in steps, each step building a layer on top of another. Tim Berners-Lee's vision based semantic vision on a five layer architecture [1]. Semantic web stack includes *XML as a syntax layer*, taking advantage of xml as a tool that allows users to add structure to their documents. But yet XML still says nothing about what the structures mean [2]. Thus, a data layer had to be added. *RDF* represents the data layer, utilizing RDF's ability to allow the encoding, exchange and reuse of structured metadata. Principally, information is represented by very generic means, i.e. directed partially labeled graphs [3]. But yet improvements to RDF had to be made as RDF does not make assumptions nor does it define semantics about any particular domain. Thus, is it up to the user to do so in RDF Schema (RDFS) which enables people to create their own RDF vocabularies employing object oriented programming model. However, RDF and RDFs both sustain the limitation that two systems may use different identifiers for what is in fact the same concept limiting the reuse of data among different applications which was a basic aim of the semantic web vision. Ontology came out as an accepted solution to the defects encountered by RDF and RDFs. Ontologies describe formal, shared conceptualizations of a particular domain of interest [1]. This common definition of domain theories and vocabularies help both people and machines to communicate concisely, supporting the exchange of semantics and not only syntax [4]. In order to make effective machine understandable manipulation of data, Logic has to be a part of semantic web vision to enable inferences, e.g. to choose courses of action and answer questions. And finally proof allows the explanation of given answers generated by automated agents.

3. Knowledge representation

A unified representation of web data and web resources is absolutely necessary to support semantic web vision in which resources could be annotated with machine-processable metadata providing them with background knowledge and meaning. Moreover annotation could be exploited by users' agents that can carry out sophisticated tasks and searches roaming from page to page. Ontologies as well, playing a main role towards this direction of web technology. In this section, annotation and ontologies tools are to be reviewed.

3.1 semantic annotation

An important condition for achieving semantic web goal is the ability to annotate web resources and associate metadata to content. Semantic Annotation is about assigning to entities in the text links to their semantic descriptions by adding emphasis (e.g., underlining, highlighting, special markings) and/or critical or explanatory notes to a text [5]. In order to do this, users need representation languages, ontologies and support tools. In order to support semantics for semantic web, markup languages, among which RDF, RDFs , OWL, ...etc [6] have been widely used. However, there will always be the question whether regular users can use these languages. In order to avoid the burden that regular users face in annotating web pages, support tools have been developed to help user in annotation process. Generally, there are three methods of acquiring semantic annotations, manual, semi-automatic and automatic annotation systems.

3.1.1 Manual annotation

Manual annotations follow the idea to provide users with useful visual tools for manual annotations and navigation with capabilities to stamp a web document. One of the first attempts made to provide an annotation tool was ANNOTEA., a Web-based shared annotation system based on an RDF infrastructure. Annotations are modeled as a class of metadata and are viewed as statements made by an author about a Web document. Annotations are external to the documents and can be stored in one or more annotation servers enabling the reuse of annotations. ANNOTEA aimed to the re-use of existing W3C technology as possible combining RDF with XPointer, XLink, and HTTP [7]. However, as stated earlier, RDF encountered defects that's why there was a need for tools that were based on a richer markup language. Thus, based on ANNOTEA, an OWL-based tool SAHA was developed. SAHA eases the process of creating ontological descriptions of documents and supports collaborative creation of metadata by centrally storing annotations, which can be viewed and edited by different annotators. Concepts defined in external ontologies can be used in annotations by connecting SAHA to ontology servers [8,9]. These two systems share the advantage of being browser-based which imposes little requirements to end user computational environment. Nevertheless, there have been the everlasting shortcoming of manual systems, as manual annotation depends on annotator's experience in the subject domain which is time consuming and expensive having millions of documents to be annotated. So, semi-automatic annotation tools have been targeted.

3.1.2 Semi-automatic annotation

Semi-automatic solutions are based on creating semantic metadata for computer processing. CREAM one of the first introduced frameworks for annotating web documents, it enables web authoring and annotation blurring the boundary between authoring and annotation processes. CREAM makes use of an RDF crawler that gathers interconnected fragments of RDF from the Web and builds a local knowledge base from this data and construct relational metadata that comprises class instances and relationship instances based on a domain ontology[10]. CREAM lacked the usage of information extraction techniques which had been fulfilled in later development resulting in S-CREAM (Semi-automatic CREAtion of Metadata), a framework that integrated a learnable information extraction component which learns information extraction rules from manually marked-up input. S-CREAM aligns conceptual markup, which defines relational metadata[11].

An implementation of the S-CREAM proposed Ont-O-Mat which is a component-based, ontology-driven markup tool. Ont-O-Mat provided users with capabilities for creating and maintaining ontology-based DAML+OIL markups enabling annotation through an easy to use graphical user interface [11].

A later development of semi-automatic tool based on ONT-O-mat was SMORE, they both share the notion of blurring the boundaries between web authoring and annotation. SMORE extended the capabilities of earlier tools by combining content creation and annotation giving users the ability to mark up parts of images. SMORE enables users to collect information from web, including ontology search capability as well as making use of multiple ontologies [12].

Since text is not the only data that can be subject to annotation and seeing that multimedia content becoming an important component of today's web content. It was necessary to have annotation tools that enable annotation of multimedia content. Based on ONT-O-mat, M-ONT-O-mat was proposed as a development of the former tool, this development included the support of multimedia content making use of MPEG-7 standards. MPEG-7 provides color, texture, shape and motion descriptors that extract visual, low-level, non-semantic information from images and videos. OntoMat-Annotizer links MPEG-7 visual descriptions to conventional Semantic Web ontologies and annotation creating ontologies that include instances of high-level domain concepts together with specification of visual descriptors allowing multimedia content analysis [13,14].

BAsAS, a semi-automatic architecture that makes it easy for users to start annotating and its named for its main characteristics: Browser, Annotation server, AJAX and SPARQL. BAsAS offered a SPARQL interface to access annotations stored on server. Annotation server is responsible receiving information from annotation interface and updating requests for stored data. Annotation server accessed using an AJAX interface which helps to keep the client side of the annotation as lightweight as possible [15].

All of the aforementioned semi-automatic annotation tools share the common characteristic of being based on ontologies in order to enhance the annotation process, nevertheless, semi-automatic solutions requires a certain amount of manually annotated pages on top of which the system can be trained. However, even with machine assistance, this is a tough, time consuming and error-prone task. This led to the emergent need for automatic solution for annotation process.

3.1.3 Automatic annotation

Automatic annotation tools make use of ontologies as well as information extraction tools in order to identify all entities in web documents. ONTEA, an automatic annotation platform that works on text in a particular domain based on domain ontology, detecting metadata from text and prepares documents structure and creates new class for each application. ONTEA analyzes a document or text using regular expression patterns and detects equivalent semantic elements according to the defined domain ontology creating a semantic version of the text which can serve as a basis for further computer processing [16,17].

Being dependent only on ontology may provide unreliable results, thus Kim addressed this shortcoming by providing facilities of indexing and retrieval making use of ontology in combination of a knowledge base. KIM came with embedded upper-level ontology and a knowledge base providing extensive coverage of entities of general importance. Ontologies and knowledge bases involved are handled using Semantic web technology and standards, including RDF(S) repositories, ontology middleware and reasoning. The platform allows KIM-based applications to use it for automatic semantic annotation, content retrieval, and querying and modifying the underlying ontologies and knowledge bases.[18].

Giving variety of options for users to use was the main objective of MnM. This tool supports both automated and semi-automated annotation of web pages as well as making use of multiple ontologies. MnM integrates a web browser with an ontology editor and provides open APIs to link to ontology servers and for integrating information extraction tools, besides its support for large-scale automatic markup of web pages [19].

As previously stated, both XML and RDF although having capabilities of providing structure to web documents but it lacks the logic that supports information reuse and help agents infer answers to user queries. Thus basing annotation tools on OWL which is a W3C standard language for providing logic is an added value and that was the base for NOMOS annotation tool. It was designed for use by annotators, corpus developers, and corpus consumers, emphasizing configurability for a variety of specific annotation tasks. Its features include synchronized multi-channel audio and video playback, compatibility with several corpora, platform independence, and mixed display of temporal, non-temporal, and relational information [20].

Automatic annotation may encounter a problem when dealing with different languages. As different languages may give a completely different meaning for a certain sentence, thus taking language attribute in to consideration is a necessity. In this context, APLODA (Automated Processing of Ontologies with Lexical Denotation Processing of Annotation), adopts this view point [21].

EXCOM, an XML-based automatic text annotator that doesn't make use of ontologies whereas its full focus is on the linguistic processing in order to facilitate information extraction process[22].

3.2 Ontology

The success of semantic web depends on the availability of ontologies in much the same degree of metadata markup tools. As previously indicated, ontologies have a great potential to describe the semantics of domain in both a human-understandable and computer-processable form. Building ontology can be of great benefit as it enables sharing common understanding of the structure of information among people or software agents and enabling reuse of domain knowledge resulting in an enhanced search and retrieval of web resources [23].

The process of building an ontology is an iterative design process that goes through the entire life cycle of ontology development. The first step involves the determination of domain and scope of the ontology. Afterwards, a consideration may be given to the reuse of existing ontologies instead of building a new one from scratch. The reuse of existing ontologies may involve merging multiple ontologies. What ever the decision was - building, merging or reusing existing ontology- the third step must involve the enumeration of important terms of the domain .so that an ontology can be reliably built or existing ontology can be chosen. In the fourth step, classes and class hierarchy must be defined. The next step includes the definition of the properties of classes (slots). Define the facets of the slots to describe values, allowed values, cardinality . etc... he last step in the life cycle employs the creation of instances of classes in hierarchy[24].

The process of building ontology can be tedious and troublesome if performed manually by users [25]. Thus, the automatic derivation of ontologies from Web sources without human review is to date a challenging research issue. Therefore, many editing tools have been developed to help users to create and edit ontology. Requiring users to be trained to knowledge representation and predicate logic, XML as markup language is not suited to describe the interrelationships of resources. Therefore, W3C has suggested the resosurce description framework (RDF), RDF schema (RDFS), DAML+OIL and OWL. Based on these languages many ontology engineering tools have been developed.

The first attempts for creating ontology editors came up with platforms that supported browsing and editing ontologies in specific languages. The first ontology editing tool was Ontolingua Server[26], based on Ontolingua language. Ontolingua server aimed to ease the development of ontologies with a form based web application. Including many embedded tools that allowed editing, merging besides the capability of connecting to remote or local ontologies.

Afterwards, WebOnto was developed proving superiority over Ontolingua in supporing collaborative browsing, creation and editing of ontologies represented in the knowledge modeling language OCML besides being scalable to large ontologies. WebOnto uses an easy to use Java based client designed to provide a direct manipulation interface displaying ontological expressions using a rich medium [27].

But yet, both Ontolingua and WebOnto were developed to support ontologies built using specific languages besides being oriented to research activities and mostly were developed as standalone tools with no extensibility facilities. As anatural consequence, another generation of ontology-engineering environments was developed aiming to support to most of the ontology lifecycle activities.

One famous editing tool in this context is Protégé -2000, Protégé allows the construction and reuse of domain ontology. It can be easily extended to access other knowledge based embedded applications as well as providing the ability to edit OWL-based ontologies as well as generating graphical representations[28]. However, Protégé lacked the support of knowledge base access from the web.

Being able to support ontologies represented by different languages was the motive for the development of OntoEdit, as it supports data import and export via different formats, such as RDF/RDFS, OWL and other formats [29]. the KAON (Karlsruhe Ontology) tool suite has been developed as the successor of OntoEdit. KAON targeted for business applications by providing a framework for building ontology-based applications [30].

OilEd, came to fulfill the need for an editor that allows ontology creation and editing based on DAML+OIL. The basic design has been heavily influenced by Protege and OntoEdit, but OilEd provides extended expressivity as well as reasoning capabilities [31]. However, OilEd was not intended as a full ontology development environment it didn't support the development of large-scale ontologies and much more functionalities provided in former tools.

SWOOP (Semantic Web Ontology Overview and Perusal) is a simple, scalable, hypermedia inspired OWL ontology browser and editor written in Java. SWOOP has reasoning support as well as the facility of supporting multiple ontology environments. Ontologies can be imported from OWL, XML, RDF and text formats. These formats could be used to save the edited ontologies [32].

4. Conclusion

Today, World Wide Web is a crucial resource and Semantic Web, with unified metadata is vital for facilitating information-based tasks. In order for this to be achieved, a meaningful representation of web data is needed to provide flexible and reusable content. Knowledge representation techniques are used to enhance web-based applications with embedded user interfaces and application logic trying to reach their full potential. Knowledge representation languages and tools have been developed and extended to support well defined organization and structure of web content. Markup languages as well as tools were developed to help annotating web content as well as creating ontologies. Thus, the tools described in this paper are examples of some of the basic technologies that are needed to consolidate the foundation of semantic web technology. These include tools for annotating and creating marked up pages easily and tools for building ontologies showing advances and ongoing research.

Refernces:

[1] T. Berners-Lee, J. Hendler, O. Lassila, "The Semantic Web," Scientific American, p: 35–43, 2001.

[2] S. Decker, S. Melnik, F. van Harmelen, D. Fensel, M. Klein, J. Broekstra, M. Erdmann, I. Horrocks. "The Semantic Web: the roles of XML and RDF". Internet Computing, IEEE Vol 4, Issue 5 P: 63-73, 1089-7801. 2002.

[3] K. Selcuk Candan, H. Liu, R. Suvarna. "Resource Description Framework: Metadata and Its Applications", ACM SIGKDD Explorations Newsletter, VOL Issue 1, July 2001

[4] A. Gómez-Pérez , O. Corcho,"Ontology Languages for the Semantic Web", IEEE Intelligent Systems and their Applications. Vol. 17 Issue 4, 2002.

[5] A. Kiryakov, B. Popov, I. Terziev, D. Manov, and D. Ognyanoff, "Semantic annotation, indexing, and retrieval". Journal of Web Semantics, VOL 2, Issue 1, (2005).

[6] Grigoris Antoniou, E. Franconi, F. van Harmelen, "Web Ontology Language: OWL ", Handbook on Ontologies in Information Systems, Springer-Verlag: p. 76-92, 2003.

[7] J. Kahan, M. Koivunen, E. Prud'Hommeaux, R. Swick. "Annotea: An Open RDF Infrastructure for Shared Web Annotations", Computer

Networks, Vol 39, Issue 5, P: 589-608. 2002.

[8] O. Valkeapää, E. Hyvönen. "An Adaptable Framework for Ontology-based Content Creation on the Semantic Web". J. UCS , Vol. 13 , Issue. 12, p: 1835-1835. 2007.

[9] O. Valkeapää, E. Hyvönen "A Browser-based Semantic Annotation Tool for Distributed Content Creation". 1st Asian Semantic Web Conference (ASWC2006), 2006.

[10] S. Handschuh and S. Staab, "Authoring and annotation of web pages in CREAM", in Proc. WWW, 2002, p:462-473.

[11] S. Handschuh, S. Staab, F. Ciravegna. "S-CREAM -- Semi-automatic CREAtion of Metadata". Expert Update, Vol. 5 Issue. 3 (ISSN 1465-4091), 2002.

[12] V. Uren, P. Cimiano, S. Hanndschuh, M. Vargas-Vera, E. Motta, F. Ciravegna. "Semantic Annotation Systems for Knowledge Management: A Survey of Requirements and State of the Art", Journal of Web Semantics, Vol 4, Issue 1, p: 14-28, Elsevier, 2006.

[13] K. Petridis, D. Anastasopoulos, C. Saathoff, N. Timmermann, Y. Kompatsiaris, S. Staab. "*M-OntoMat-Annotizer: Image Annotation Linking Ontologies and Multimedia Low-Level Features*". Springer Verlag. 10. p: 633-640. Knowledge-Based Intelligent Information and Engineering Systems, Proceedings of the 10th International Conference, KES 2006.

[14] S. Bloehdorn, K. Petridis, C. Saathoff, N. Simou, V. Tzouvaras, Y. Avrithis, S. Handschuh, I. Kompatsiaris, S. Staab, and M. G. Strintzis: "Semantic Annotation of Images and Videos for Multimedia Analysis", 2nd European Semantic Web Conference, ESWC 2005, Heraklion, Greece, May 2005.

[15] V. Zacharias, "The BAsAS Architecture For Semantic Web Annotations". Poster at the 1st Semantic Authoring and Annotation Workshop (SAAW 2006), at the ISWC 2006

[16] M. Laclavík, M.Šeleng, E.Gatial, Z.Balogh, L.Hluchý. "Ontea: Semi-automatic pattern based text annotation empowered with information retrieval methods". proceedings in informatics and information technologies. - Kosice : Vydavatelstvó STU, ISBN 978-80-227-2716-7, part 2, P. 119-129. 2007.

[17] M. Yasrebi, M. Mohsenzadeh. "Semi-Automatic Approach for Semantic Annotation". PWASET VOL. 38 ISSNB:2070-3740 P: 869:872, 2009.

[18] B. Popov, A. Kiryakov, A. Kirilov, D. Manov, D. Ognyanoff, M. Goranov, "KIM – a semantic platform for information extraction and retrieval", Natural Language Engineering Vol 10, Issue 3-4 P: 375 - 392 ISSN:1351-3249, 2004.

[19] M. Vargas-Vera, E. Motta, J. Domingue, M. Lanzoni, A. Stutt, F. Ciravegna. "MnM: Ontology Driven Tool for Semantic Markup". In Proceedings of the Workshop Semantic Authoring, Annotation & Knowledge Markup (SAAKM 2002) with ECAI 2002,2002.

[20] J. Niekrasz, A. Gruenstein. "NOMOS: A semantic web software framework for annotation of multimodal corpora". In: 5th Conference on Language Resources and Evaluation 2006.

[21] C. Wartena, R. Brussee, L Gazendam, W. Huijsen. "Apolda: A Practical Tool for Semantic Annotation". dexa, p: 288-292, 18th International Conference on Database and Expert Systems Applications (DEXA 2007), 2007.

[22] B. Djioua, J. J. García Flores, A. Blais, J.Pierre Desclés, G. Guibert, A. Jackiewicz, F. Le Priol, L. Nait-Baha, B. Sauzay, "EXCOM: An Automatic Annotation Engine for Semantic Information". FLAIRS Conference p: 285-290, 2006.

[23] I. Horrocks, " Ontologies and the semantic web". Communications of the ACM Vol 51 , Issue 12 P: 58-67 , ISSN:0001-0782, 2008.

[24] B. Richardson, L.J. Mazlack, "Approximate ontology merging for the semantic Web", Fuzzy Information, 2004. Processing NAFIPS '04. IEEE Annual Meeting Vol. 2, P: 641- 646 (ISBN: 0-7803-8376-1), 2004.

[25] S. C. Buraga, L. Cojocaru, O. C. Nichifor. "Survey on Web Ontology Editing Tools", Proceedings of the 7th International Conference On Technical Informatics – CONTI 2006, Timisoara, 2006.

[26] A. Farquhar, R. Fikes, J. Rice, "The Ontolingua Server: A Tool for Collaborative Ontology Construction", in: Proc. 10th Knowledge Acquisition for Knowledge-Based Systems Workshop (KAW96) Banff, p 44.1–44.19, 1996.

[27] E. Motta, "An Overview of the OCML Modelling Language". In Proceedings of the 18th Workshop on Knowledge Engineering Methods & Languages (KEML), 1998.

[28] N. F. Noy, M. Sintek, S.Decker, M. Crubézy, R. W. Fergerson, M. A. Musen. "Creating Semantic Web Contents with Protégé-2000", IEEE Intelligent Systems archive, Vol. 16 , Issue 2 P: 60 – 71, 2001.

[29] Y. Sure, J. Angele, S. Staab. "OntoEdit: Multifaceted Inferencing for Ontology Engineering." J. Data Semantics I. p: 128-152. 2003

[30] E. Bozsak, M. Ehrig, S. Handschuh, A. Hotho, A. Maedche, B. Motik, D. Oberle, C. Schmitz, S. Staab, L. Stojanovic, N. Stojanovic, R. Studer, G. Stumme, Y. Sure, J. Tane, R. Volz, V. Zacharias, "KOAN : towards a large scale semantic web", E-Commerce and Web Technologies, Third International Conference, EC-Web 2002, Aix-en-Provence, France, 2002.

[31] Sean Bechhofer, Ian Horrocks, Carole Goble, Robert Stevens. OilEd: a Reason-able Ontology Editor for the Semantic Web. Proceedings of KI2001, Joint German/Austrian conference on Artificial Intelligence, Springer-Verlag LNAI Vol. 2174, p: 396-408. 2001

[32] A. Kalyanpur, B. Parsia, E. Sirin, B. Cuenca-Grau, J. Hendler. "Swoop: A 'web' ontology editing browser". In Journal Web Semantics: Science, Services and Agents on the World Wide Web Vol 4, Issue 2 , P: 144-153, 2006.