International Journal of Computer Applications (0975 – 8887)
Volume 54– No.13, September 2012

Publishing Knowledge and Transformation of Conventional Web into Semantic Web using Metadata

Khandaker Tabin Hasan
Department of Computer Science and Engineering (CSE)
American International University of Bangladesh(AIUB)

Md. Saddam Hossain Mukta
Department of Computer Science and Engineering (CSE)
Islamic University of Technology (IUT)
Dhaka, Bangladesh

Mir Tafseer Nayeem
Department of Computer Science and Engineering (CSE)
Islamic University of Technology (IUT)
Dhaka, Bangladesh

Sharmin Afsah
Department of Electrical and Electronic Engineering (EEE)
American International University of Bangladesh(AIUB)

ABSTRACT
In Semantic Web research, most of the work has focused around the OWL ontology, natural language processing, word sense disambiguation and semantic matching. Few efforts have been made to provide a publishing methodology for metadata-rich Web content for end users. This work encompasses all possible aspects of publishing pure digital content, such as HTML documents, images, video, and other media documents free from metadata. Afterwards the content could be integrated with metadata to provide optimal capability and scalability for document organization, search and navigation in the aim of explicit knowledge publishing. The effort is made on our specific framework to publish documents which are enriched with metadata for better machine reasoning. The first and the second version of the prototype is a Mozilla Firefox extension called Semantic Web (SWeb) browser. SWeb is intended for managing and publishing knowledge and information, as "publishing knowledge" was missing in the system where we got the opportunity to contribute.

General Terms
Semantic Web, Knowledge Transformation, Metadata Visualization.

Keywords
Knowledge, Ontology, OWL, metadata

1. INTRODUCTION
Tim Berners-Lee *et al.* first proposed the idea of semantic Web at the very first International World Wide Web Conference, at CERN, Geneva, in September 1994. The World Wide Web Consortium (W3C) was first formed at that conference. Since then extensive research has been done under many different projects. The definition of semantic Web as provided by W3C is "*An extension of the current Web in which information is given well-defined meaning, a place where machines can analyze all the data on the Web, even a Web in which machine reasoning will be ubiquitous and devastatingly powerful"* [27]. The Semantic Web has two primary properties toward its definition. First, common formats for integration and arrangement of data drawn from diverse sources, whereas the original Web mainly concentrated on the interchange of documents. Second, a language for recording how the data relates to real world objects.

Recently, as in the AI approach, working on Semantic Web heavily depends on deductive reasoning which is problematic since "*It must be the case that a significant portion of the inferences we want [to make] are deductions, or it will simply be irrelevant how many theorems follow deductively from a given axiom set."* [8]. However the process is evolutionary and our focus here is to produce sufficiently metadata-rich Web documents for machines to extract knowledge for users or machines and how to publish them in a specific semantic Web framework where metadata is defined as "*Structured information that describes and/or enables finding, managing, controlling, understanding or preserving other information over tim*e" [19].

To provide advanced knowledge [3] services, we need efficient ways to access and extract knowledge from Web documents. Although Web page annotations could facilitate semantic knowledge gathering, annotations are hard to find and will probably never be rich or detailed enough to cover all the knowledge these documents contain. Manual annotation is unfeasible and unsalable and automatic annotation tools are still largely unreliable. Specialized knowledge services therefore require tools that can search and extract specific knowledge directly from unstructured or semi-structured text on the Web, guided by an ontology that details what type of knowledge to harvest [9]. As a complement to knowledge extracting tools, we use a technique that allows users and publishers to specify their own set of metadata to best describe the content which is referred to as *attributes* in this document. An attribute is a pair of *name* and *value.* The novel tool we use here for publishing Semantic Web (SWeb) documents, converting conventional Web documents into SWeb documents, classifying and managing them accordingly is a Semantic Web browser (SWeb), developed in the project "KnowDive" at the University of Trento [10]. Our paper covers the publishing methodology of Semantic Web documents using our existing tools and framework.

Knowledge is reproducible and sharable [12] as long as it is not tacit (like feeling of happiness). Predictive answers to those fundamental questions refer to *Wisdom*.

In this paper, we have proposed a mechanism to use metadata to publish knowledge which provides an optimal process to convert the traditional web into semantic web. We are inspired by numerous semantic web major works on semantic world. In our system we have used semantic web technologies and algorithms, i.e. OWL [31], WSD [42] which have more expressive power as a result it can conform semantic consistency. The rest of the paper is organized as follows. We make a brief focus on some methods used so far i.e. OBO-Edit, SIMILE in section 2. Major principles used in so called web development technologies in Section 3**.** Several technical and semantic inconsistencies and flaws are presented in Section 4**.** In section 5, we describe different steps to translate the document into semantic web, for instance document organization, interpolation, CUD, Finally a comprehensive discussion and comparison between semantic and traditional web.

2. STATE OF THE ART

Flexible information Access using Metadata in Novel Combinations (FLAMENCO) is a research supported by a CAREER grant awarded to Prof. Marti Hearst from the National Science Foundation, NSF IIS-9984741 [13]. Flamenco provides a search interface framework that has a primary design goal of user to move through large information spaces with maximum flexibility of navigation and control. The interface uses hierarchical faceted metadata with explicit exposure of category to guide the user toward possible choices and to organize the results of keyword searches. The interface uses metadata in a manner that allows users to both refine and expand the current query, while maintaining a consistent representation of the collection's structure. This use of metadata is integrated with free-text search, allowing the user to follow links, and then add search terms, then again follow more links without interrupting the interaction flow.

SIMILE [29] is a joint project conducted by the MIT Libraries and MIT Computer Science and Artificial Intelligence Laboratory. As the project objective described in [15], it seeks to enhance interoperability among digital assets, schemata/vocabularies/ontology, metadata and services. SIMILE will leverage and extend DSpace [14], enhancing its support for arbitrary schemata and metadata, primarily through the application of RDF [28] and semantic Web techniques. The project also aims to implement a digital asset dissemination architecture based on Web standards. The dissemination architecture will provide a mechanism to add useful "views" to a particular digital artifact (i.e. asset, schema, or metadata instance), and bind those views to consuming services. A key challenge is that the collections which must inter-operate are often distributed across individual, community, and institutional stores. The objective is set to be able to provide end-user services by drawing upon the assets, schemata/vocabularies/ontology, and metadata held in such stores.

Several other semantic Web projects are running under SIMILE and different semantic tools are being developed under them. A very recent of their work is a Mozilla Thunderbird mail client [30] extension called Seek for faceted navigation of mails.

OBO-Edit is an open source ontology editor developed by the Berkeley Bioinformatics and Ontologies Project [18]. This is a desktop application with necessary features to create, edit, visualize, and specially optimized for OBO biological ontology file format. So far it offers no integrated features for publishing metadata rich web.

Protégé is a desktop application for ontology creation and editing, and a knowledge-base framework developed at Stanford University [17]. Protégé supports two ways of modeling ontologies e.g., Protégé-Frames and Protégé-OWL. The ontologies being created with Protégé can be exported to RFD(S), OWL[31] and XML Schemas[32]. However, it does not support publishing documents with the ontologies that has been created with it. Apart from different interaction pattern, the visualization is similar to OBO-Edit.

SWeb is a Semantic Web application that has been developed in the KnowDiveproject [43] that uses semantic organization of user resources. It uses classification encoded into lightweight ontology [16] and other powerful linguistic tools that provides an excellent framework for document organization and metadata management. The key components that integrated in SWeb framework are –SWeb Classification, Attribute Management System, eTypes, Semantic Enrichment, SMatch [23], Get-Specific [25]. Most of our work has been accomplished on top of this architecture.

3. PRINCIPLES OF WEB DEVELOPMENT

3.1 Navigation

The Web documents are the resources organized onto a single system or distributed over the globe and therefore, needs an intuitive form of navigation in order to explore through the complex network of information space. The number of indexed Web contents as of August 2012 is over 8.09 billionpages[33] and this sheer size is the clear indication of the potential challenge for navigation and search on the Web [4]. Without a proper navigation model, user may become lost into the space of a single site. Navigating between sites is something that allows user to look for information on a global scale, let us call it **External Navigation** [34]. On the other hand, navigating within a site is something very intrinsic and well-thought approach by most of the Web developers; let us call it **Internal Navigation.**

3.2 Search

Search is the user's lifeline when navigation fails. Even though advanced search can sometimes help, simple search usually works best, and search should be presented as a simple box, since that's what users are looking for [1]. Provision of search is the last hope when we are lost in the information space or simply do not know where to look for the desired information. However, there are few questions are to be addressed first: When to search (the time), what to search (the knowledge), Where to search (the space) and finally, How to search (the method).

The time specifically describes the situation when the user needs to search something which is not apparently available. Knowledge is something more than information that is being conceptualized by the user. The search could range from a word on a page to an idea in the web. Both time and knowledge are deeply involved into the user's cognitive state and we, the developers, by recognizing user requirements, provide a clue to space and a method to perform the operation.

3.3 Access Control/Authentication

Authentication provides a mean to protect documents from unauthorized access and the kind of HTTP transactions must be determined for which authentication is required beforehand. Client authentication is a common requirement for modern Web sites as more and more personalized and access-controlled services move online [1]. There are two types of authentications:

- Basic authentication: It is supported by most HTTP server daemon and browser and provides virtually no security as it sends password in a clear text over the network.
- Digest authentication: It provides a secure means of transaction in an encrypted format.

The order of topics as, will be discussing in this paper, is **Navigation, Search and Authentication** with comparisons respectively

4. CONVENTIONAL METHODOLOGY FOR WEB BROWSING PROCESS

4.1 External Navigation

There is no novel way for external navigation till now except the address bar on the browser where we need to put the exact URL for the document.

However, there are some services exist e.g., DMoz[35], Google Directory[37] or Yahoo directory[36] which provides an incomprehensive means of external navigation. These services use simple directory of classification structure of rooted tree.

4.2 Internal Navigation

Simplest of the navigation tools is a list of links pointing to the documents. This technique is still frequently used in parallel with modern navigation bars that are being found in today's most internet sites. Others are well decorated vertical or horizontal navigation bar commonly referred as 'menu bar' often with multi-level popup.

Fig 1: List of links for navigation.

4.3 Search

Searching for key words in a document is usually a built-in feature of the browsers. But searching for a document based on key words or phrases in or out side of a site is the function of a search engine. Featuring search capability in a site may well be developed with the site itself or developers may use a third party search engine like Google. Whatever the search engine is being used, not only the search efficiency in terms of precision and recall, but the visualization of the search results is also very important. The most popular search engine Google presents a list of document title with a summary pointing to the document URL. Following are some examples of conventional search interface and visualization.

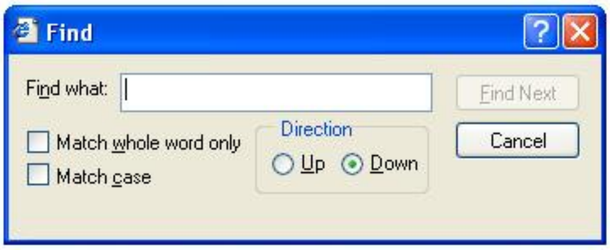

Fig 2: Searching for word in document, Windows Internet Explorer.

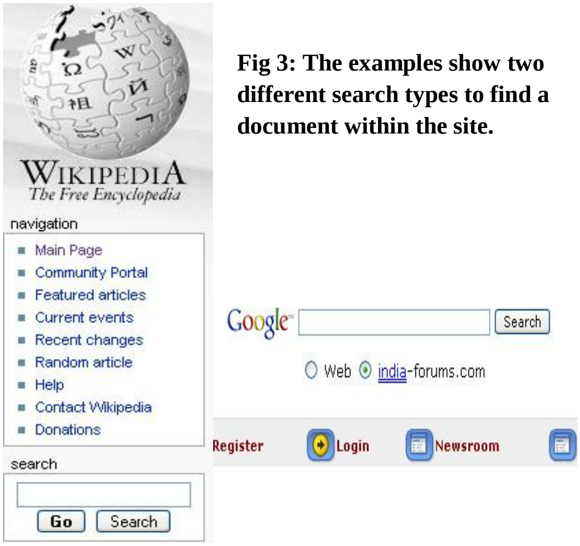

Fig 3: The examples show two different search types to find a document within the site.

To search for document as of a global resource several engines exist. They provide both a linear search and a directory like tree structured search and as usual the visualization of the search result is often a long list of links. Although, their advanced search option provide elaborate filtered search, but how efficient it is, since very few Web documents are built in compliance with the document description, still remains a question.

4.4 Authentication

By authenticated access, users are provided with personalized access to available resources and scope of interaction by security policies. A generic login mechanism (Fig 4) is provided usually on a page that works as an access to authentic users. Some of the applications prompt for authentication as a popup to provide access to restricted resources. i.e. Windows resource access attempts pop up being made.

Fig 4: Common Login frame on Yahoo mail home page.

5. TRANSLATING THE DOCUMENT INTO SEMANTIC WEB

The primary objective was to separate data from metadata of Web documents. Therefore, we first define some unfamiliar forms of metadata and remove them from the data.

5.1 Handling links on the web page

Any links embedded in the document is regarded as metadata [13][19][20][22] and Semantic Web provides different means to handle them. Link types are common in conventional Web pages such as Navigation (Internal and External), Fragment identifier, Inline links in the content, E-mail links, Sponsored links. Conventional Web pages also contain some metadata [13][19][20][22]defined in "*meta*" elements, but this information is not viewable to the user, instead widely used by the search engines. While designing SWeb, the key issues were to decide first what are the elements go as metadata and what will be left as data only.

The whole navigation map can be structured as SWeb tree, maintaining proper semantic alignment. But inline links, fragment identifier and e-mail links are embedded into the document itself and needs some special methods to separate them as metadata while retaining their intended usefulness. The issues are addressed as follows for SWeb publication.

For the case of fragment identifier, to have more control on data, SWeb publishing splits the document into small convenient sizes and integrates the fragment identifier in the SWeb navigation tree. The advantages being offered are - each fragment will have their own attribute value set and will allow easy CRUD operations as independent documents by "part-of" [38] relationship. For inline links, Since SWeb publishing aims for almost a new way of document publishing on the web, inline links are considered a decoupled attribute in the meta-document and are internally coupled with their references on the document. These inline links appear the way it appears now on the conventional Web documents for backward compatibility and the dereferencing is done by pointing to the corresponding attribute of a meta-document for each link.

5.2 Translating document data

Some available embedded Web documents data (document title, URL, date of creation, date of last modification, document size, etc.) has been extracted and transformed into SWeb metadata as document attributes. The collection of attributes associated with a document is called a meta-Document (mDoc) of the document. The documents are organized into a hierarchy of nodes that forms a formal classification and the relations for document-to-node, document-to-document and facets [38] definition, and facets hierarchies are maintained by using Propositional Description Logic[39].Automated process is fully automated where the self-describing data are extracted from the document to form the *meta-document*, and the result depends on the quantity and quality of data being structured in the document. The very common source metadata are found in "*title*" and "*meta*" element of conventional Web pages. The additional attributes are added those are found from the semantic relationship between node to node and the document's position in them.

5.3 Methodology for semantic web browser

This sub section describes the navigation, search and authentication for SWeb specification. A working prototype SWeb interface is shown below in fig 5. Frame 1 for navigation tree of classification[38], frame 2 represents the *meta-documents* classified into the selected node; frame 3 shows the selected document and frame 4 shows the attributes of the document. First three frames are visually dynamic in size and position.

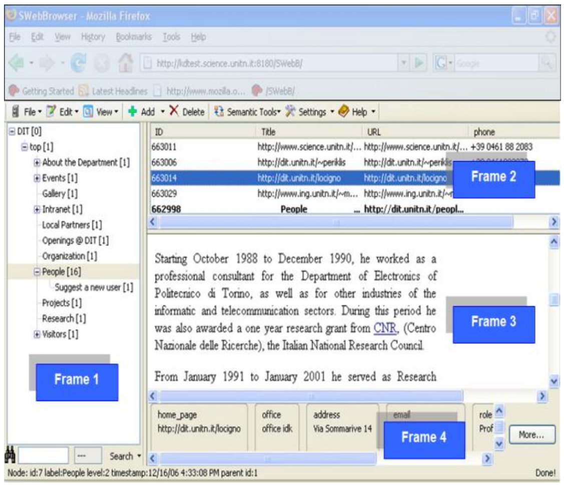

Fig 5: SWeb interface as it appears on the Mozilla FireFox browser.

Semantic enrichment is the user chosen action on the tree, sub-tree or a node which generates propositional DL [39] formula after disambiguating the object(s) on the web document hierarchy. This operation is necessary to establish semantic relations for node to node, and node to documents when classified under a particular node. It gives the full potential of semantic based search and match within a classification or between two classifications. For instance, a document is classified under a node *car* which is the child of node *vehicles*, the concept at the node describes the properties of the document and understandably answers iteratively twice the question "*what*". The more general answer about the document is that the content is about "vehicle" and more specifically is about "car".

Link discovery is the SWeb term for Semantic Schema Matching [23]. It is necessary to perform semantic enrichment before we can perform a kink discovery operation. The operation provides user options for node to node, node to sub-tree (includes all descendants of the node), sub-tree to node and sub-tree to sub-tree. The first object is the target object and the second object is the source object and the match is based on formula expressed in propositional DL.It also provides users to specify whether to look for equivalent match or more specific match or more general match or all.

5.3.1 Navigation

SWeb navigation tree is a visual representation of a directed acyclic graph (DAG) of classifications where a node may have multiple parents.

5.3.1.1 External navigation

Unlike DMoz or Google directory, user can browse directly from one internal site to any external node following a link.

SWeb supports two types of external links e.g. **Adopted-child** and **Context Link (CLink)**. An Adopted-child link is logically internal, but it may be physically external, and the CLink behaves physically external and could be internal on the other hand. Both types of external links can be added

manually or using Link discovery.

Fig 6: SWeb navigation tree. Blue and yellow arrows of the left tree points to external links of different nodes of the tree presented at the right.

5.3.1.2 Internal navigation

SWeb supports feature for local navigation for a site taking away the trouble of menu designing while maintaining the semantic relationship between node-node, node-document and document-document. Developers need only to write and design the document pages and develop a logical navigation model for the site. These navigation structures are *Normalized Formal Classification (NFC)* formulated by Fausto Giunchiglia et al [16] trees. A named classification is created for the site and the nodes are added with desired structure.

5.3.1.3 CUD (Create, Update and Delete) on classification

Since the semantic relationship between nodes is to be retained, the following scenarios may take place on user action for create, update and delete nodes by the use of Controlled Vocabulary [40] (this complex intelligent system behavior is still under test). Every operations described in this subsection causes change to necessary change in document attributes.

5.3.1.3.1 Interpolation and Extrapolation

There exists a sub-tree where ‘carnivore’ is the child of ‘animal’.

Case 1: (Interpolation) User attempts adding ‘mammal’ under ‘carnivore’. System prohibits the operation and finds ‘mammal’ is more specific to ‘animal’ and more general to ‘Carnivore’, therefore, ‘Mammal may well fit in between ‘Animal’ and ‘carnivore’. System suggest user to interpolate ‘mammal’ between ‘animal’ and ‘carnivore’.

Case 2: (Extrapolation) There exists a sub-tree where ‘reptile’ is the child of ‘animal’ and ‘frog’ is child of ‘reptile’. User tries to add ‘dog’ under ‘animal’. System determines ‘dog’ is a hyponym for ‘mammal’ where ‘mammal’ and reptile has same semantic distance from ‘animal’ (possible siblings) which can be put under ‘animal’ and position ‘dog’ under ‘mammal’ by extrapolation.

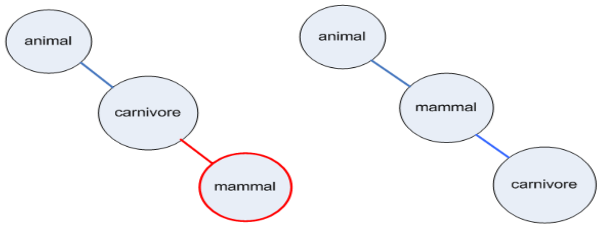

Fig 7: Interpolation of new node.

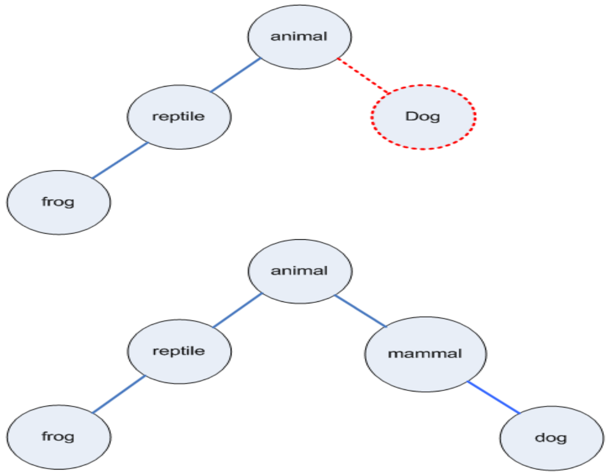

Fig 8: Extrapolating the new node by interpolating a system suggested node.

The question is how much extrapolation the system should offer? If we want to add ‘dog under ‘animal’, then ‘chordate’, ‘vertebrate’, ‘mammal’, ‘placental’, ‘carnivore’ and ‘canine’ are all candidates to be added or suggested for adding before ‘dog’. Choosing “extrapolation all” will extend the sub-tree further by adding all missing nodes in between. “Repositioning all”, reconstructs the whole tree depending on the highest existing semantic details that exists in the tree.

5.3.1.3.2 Deletion by Contraction

Deleting a node usually deletes the whole sub-tree under it. However, another option is also considered. The following tree shows the hierarchy from ‘animal’ to ‘carnivore’. Deleting the node ‘vertebrate’ deletes the whole sub-tree while user is provided with the option to delete the intended node only and rest of its children are added to the parent of the deleted node as an operation performed by contraction.

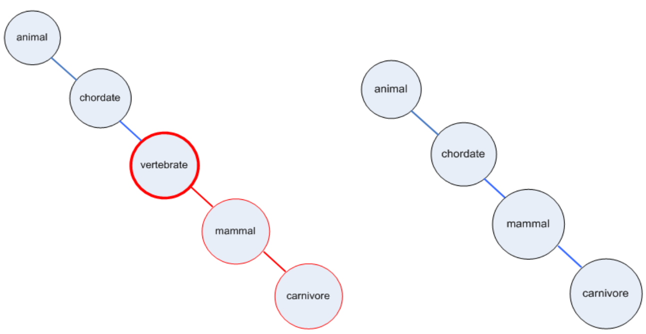

Fig 9: Contraction by removing intermediate node only.

5.3.2 Document organization

Once the navigation structure is created, user now can classify the documents that do not have any embedded navigation structure. For instance, DIT Semantic Web (SWeb) site which is hosted on a different machine, every document of http://www.dit.unitn.it is accessed by the SWeb navigation tree. The documents contain no embedded navigation menu while the same menu is integrated with SWeb as the classification tree. Important issue of using this technology is the backward compatibility where user would like to browse the site in non SWeb enabled browsers and developer should keep an option to resolve the issue that we have not put any effort on it for this moment.

Fig 10. presents the full expanded tree of classification named "DIT" that we have created during our experiment. The numbers in the square brackets of the node labels indicates the number documents being classified under the corresponding node.

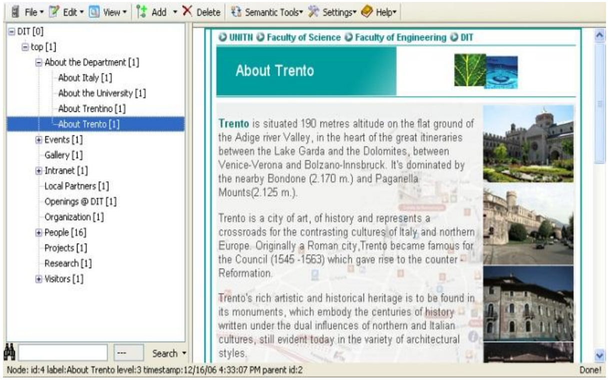

Fig 10: ADIT internal page. Navigation map is transformed into SWeb navigation tree.

Advantage over conventional web navigation

The most remarkable change here is the visual clarity and the semantic relationship. Tree structured navigation provides more intuitive means to navigate within the site. It also allows user to add, rename and remove any node or link conveniently which extends the scope of navigation manageability far beyond the site itself. Finally developers are relieved from designing a conventional navigation menu which requires substantial user study and attention.

5.3.3 Search

In SWeb, we have options for both simple search and advanced search. In this step, we have enough semantics to look for what we want and we made the best possible use of them.

Fig 11: A simple search option with a button for advanced search.

5.3.3.1 Simple Search

A simple search enables a quick search with key words or phrases within a node or a sub-tree. The search can be performed on all classifications or on a selected node and the search result is added as child node of its search scope (Fig 11). As described before, documents are classified in their relevant position in the classification where semantic relationship for node to node and node to document are affluently maintained. A disambiguated node label is the first indication where the desired documents could be found. A phrase "*Venn diagrams*" may well be found in classification for mathematics. However, the search result could be big set depending on target space that is being searched.

5.3.3.2 Advanced Search

Advanced search let users take the full advantages of semantic Web technology by specifying the attributes with values which is associated with the documents. It narrows down the search results and generates more relevant output. In SWeb, attribute name and values are also disambiguated and provides the most powerful means to find the relevant documents. For instance, filtered with the attribute subject having a value soil (*soil#4*) may also return the results on territory (*territory#3*) as they have same sense in the WordNet.

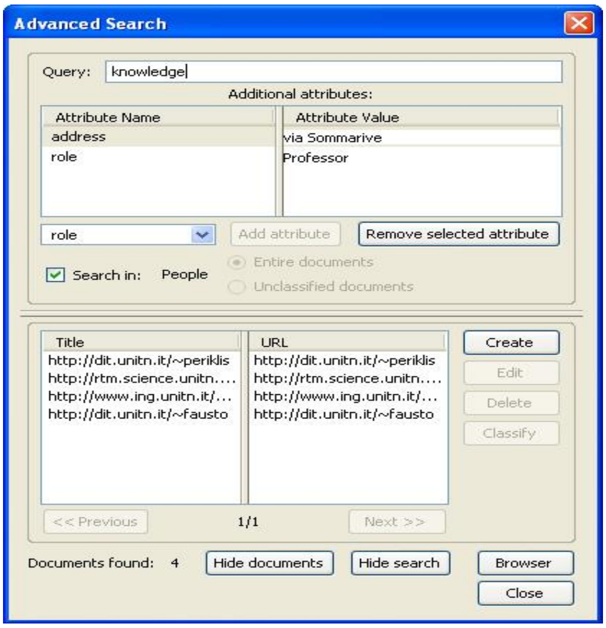

Fig 12: SWeb advanced search interface provides filtering options by document attributes.

Advantage over conventional Search

The advances of SWeb search on a metadata rich Web is numerous. The search can be performed on any specific part of Web site unlike the conventional search, which allows search only in the whole site. The search scope may range from an end node of a tree to the whole classifications in the repository or even on a fragment of a document.

5.3.4 Authentication

Current SWeb authentication is implemented with Role Based Access Control (RBAC) [41] that allows different access rights (i.e. CRUD on objects such as nodes, mDocs, classifications, etc.) for different user groups. This is also very non-trivial task to ensure security where multiple applications that serves multiple users with specific access rights to specific data and resources. Roles are essentially a semantic construct that forms the basis of access control policy [21]. In addition to our system, it also controls performing various semantic operations like semantic enrichment (which computes concepts at node and label for disambiguation with WSD algorithms)[42] and semantic match. Access to sites and resources are also integrated with SWeb authentications. With SWeb login, classified resources are available to users for which they are authorized to access.

Fig 13: Current Login option at SWeb toolbar under File menu.

Developers can control access with a single authentication system by verifying users from SWeb database. In SWeb, restricted resources are invisible to the user instead of denying access to them. Fig 14 shows DIT SWeb in SWeb navigation structure for both registered and non-registered users.

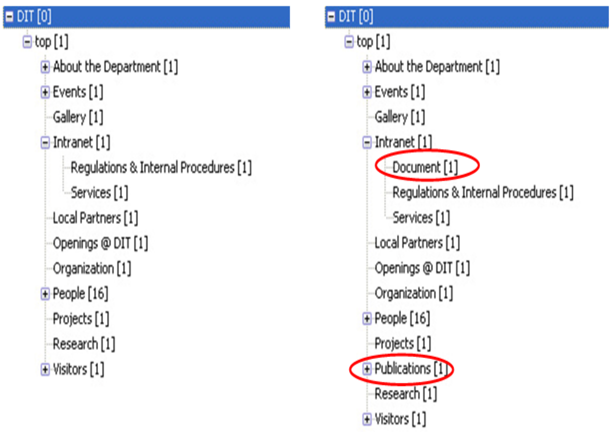

Fig 14: The nodes in the red ovals are not visible to the unauthorized users.

Advantages over conventional authentication

It provides a uniform built-in authentication to site resources, and other administrative and management operations. This authentication allows user to customize and organize resources by adding different internal or external links, and save into their profiles. To support this authentication, SWeb has an Access Control tool that allows user, role and access right management. This tool has not been discussed in my work.

6. CONCLUSION

This work has proved its usefulness for the end users in a laboratory environment as it has been tested with SWeb's first prototype. Our work, on top of the first generation of SWeb tools and components, was a formidable challenge. The system was developed as a FireFox plug-in using XUL (XML User Interface Language) and JavaScript. Absence of a suitable IDE made our work more difficult and time consuming. For every single change, the code had to be recompiled, and new plug-in had to be reinstalled. However, it was always a fun to discover or invent some methods of doing things faster and that experience later helped me in other works.

The work presented in this paper was not very user-friendly due to the constraint of existing visualization and interaction methods used in the project which made the usability of the interfaces questionable. What we tried to do was to demonstrate its potential for alternative use that was not anticipated as a concrete idea in the project agenda. The work was completed, tested and accepted in the project for a full-scale implementation and integration in future version, certifying its usefulness and quality

7. REFERENCES

[1] **JakobNielsen, "***Top Ten Mistakes in Web Design",* **1996.**

[2] Kevin Fu, Emil Sit, Kendra Smith, Nick Feamster, "*Dos and Don'ts of Client Authentication on the Web*", MIT Laboratory for Computer Science, August 2001.

[3] Jonathan Hey, "The Data, Information, Knowledge, *Wisdom Chain: The Metaphorical link*", published at Intergovernmental Oceanographic Commission - OceanTeacher: a training system for ocean data and information management, 2004.

[4] Daily Estimated Size of the World Wide Web. See at: http://www.worldwidewebsize.com

[5] Ackoff, R. L., *"From Data to Wisdom"*, Journal of Applied Systems Analysis, Volume 16, 1989 p 3-9.

[6] Clark, R.C. (1999), *Developing Technical Training (2nd edition).* Washington D. C.: International Society for Performance Improvement.

[7] Thomas R. Gruber, "*A Translation Approach to Portable Ontology Specifications*", Appeared in Knowledge Acquisition, 5(2):199-220, 1993.

[8] Drew McDermott, "*A critique of pure reason",* Computational Intelligence, 3:151-237, 1987.

[9] Alani, Harith and Kim, Sanghee and Millard, David and Weal, Mark and Hall, Wendy and Lewis, Paul and Shadbolt, Nigel,, "*Automatic Ontology-Based Knowledge Extraction from Web Documents*",IEEE Intelligent Systems18(1):pp. 14-21, 2003.

[10] Fausto Giunchiglia, KnowDive Project, University of Trento, See at: <http://dit.unitn.it/~knowdive>

[11] Aristotle, "*Posterior Analytics*", Book 1, part 2, 350 B.C., Translated by G. R. G. Mure, 2004.

[12] T. D. Wilson, "*The nonsense of 'knowledge management*"', Information Research, Vol. 8 No. 1,

[13] Yee, K.P., Swearingen, K., Li, K., and Hearst, M., "*Faceted metadata for image search and browsing*". In Proceedings of CHI 2003 (Fort Lauderdale, FL, Apr. 2003).

[14] Smith, M., Bass, M., McClellan, G., Tansley, R., Barton, M., Branschofsky, M., Stuve, D., and Walker, J., "

DSpace: An Open Source Dynamic Digital Repository". *D-Lib Magazine*, Volume 9, Issue 1, pp 2.

[15] R. Lee. "*Scalability report on triple store applications"*. Technical report, Massachusetts Institute of Technology, July 2004.

[16] Fausto Giunchiglia, M. Marchese, and I. Zaihrayeu: "*Encoding classifications into lightweight ontologies"*. In Proceedings of ESWC'06, Budva, Montenegro, June 2006.

[17] Noy, N. F., Sintek, M., Decker, S., Crubézy, M., Fergerson, R.W., & Musen, M.A. "*Creating Semantic Web Contents with Protégé-2000"*. IEEE Intelligent Systems 16(2), 60-71, 2001.

[18] Day-Richter J, Harris MA, Haendel M, Lewis S, "*OBO-Edit – An Ontology Editor for Biologists"*. Bioinformatics, Volume 23, Number 16, pp2198-2200.

[19] Weibel SL, Koch T. "*The Dublin Core Metadata Initiative"*. D-Lib Magazine December 2000. See at: http://www.dlib.org/dlib/december00/weibel/12weibel.html

[20] National Archives of Australia, AGLS Metadata Element Set, Part 1: Reference Description, Version 1.3, 2002.

[21] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D.R. Kuhn, and R. Chandramouli, "*Proposed NIST Standard for Role-Based Access Control"*, ACM Transactions on Information and System Security, vol. 4, no. 3, pp. 222–274, August 2001.

[22] Lars Marius Garshol, *Metadata? Thesauri? Taxonomies? Topic Maps! Making sense of it all*, Journal of Information Science, Vol. 30 No.4, pp.378-91, 2004.

[23] Fausto Giunchiglia, Pavel Shvaiko, and Mikalai Yatskevich, "*Semantic Schema Matching"*, Appear in proceedings of the 13th International Conference on Cooperative Information Systems (CoopIS 2005), pp 347-365

[24] Bernardo Magnini, Luciano Serafini, and Manuela Speranza, "*Making explicit the semantics hidden in schema models"*. In A. Cappelli and F. Turini, editors, *8th Congress of the Italian Association for Artificial Intelligence*, Pisa, Italy, LNAI 2829, pages 436–448. Springer, 2003.

[25] George Miller. "*WordNet: A Lexical Database"*. Communications of ACM, Vol. 38: Issue 11, pp 39-41, 1995.

[26] Fausto Giunchiglia, Ilya Zaihrayeu and Uladzimir Kharkevich, "Formalizing the Get-Specific Document Classification Algorithm". Technical Report DIT-07-013, Informatica e Telecomunicazioni, University of Trento, 2007.

[27] Berners-Lee, T., Hendler, J., & Lassila, O. (2001, May), "The Semantic Web". *Scientific American* 284, 34–43.

[28] RDF Homepage. See at: http://www.w3.org/RDF/

[29] SIMILE project home page. See at: http://www.simile-widgets.org/

[30] Mozilla Thunderbird mail client. See at: http://www.mozilla.org/en-US/thunderbird/features/

[31] OWL Web Ontology Language. See at: http://www.w3.org/TR/owl-features/

[32] XML Schema. See at: http://www.w3schools.com/schema/

[33] The size of the World Wide Web (The Internet). See at: http://www.worldwidewebsize.com/

[34] Antoon Verhoeven and Kai Warendorf, "External Navigation Control and Guidance for Learning with Spatial Hypermedia" Journal of Interactive Media in Education (JIME), 1999 (1).

[35] Open directory project. See at: http://www.dmoz.org/

[36] Yahoo directory. See at: http://dir.yahoo.com/

[37] Google directory. See at: http://directory.google.com/

[38] Fausto Giunchiglia, Biswanath Dutta and Vincenzo Maltese, "Faceted Lightweight Ontologies," Conceptual Modeling: Foundations andApplications 2009: 36-51.

[39] Benjamin N. Grosof,Ian Horrocks ,Raphael Volz, "Description logic programs, combining logic programs with description logic", WWW '03 Proceedings of the 12th international conference on World Wide Web, Pages 48 - 57

[40] Controlled vocabulary. See at: http://www.boxesandarrows.com/view/what_is_a_controlled_vocabulary_

[41] D. F. Ferraiolo, R. S. Sandhu, S. I. Gavrila, D. R. Kuhn, and R. Chandramouli,"Proposed NIST standard for role-based access control,"Information and System Security, vol. 4, no. 3, pp. 224–274, 2001.

[42] M Stevenson, Y Wilks, "Word sense disambiguation", The Oxford Handbook of Computational Linguistics, 2003.

[43] Knowdive Project. See at: http://disi.unitn.it/~knowdive