International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 05, May-2015

VM-SEMWEB: A Semantic Web for Vietnamese Mathematical Documents

Cao Xuan Tuan
Ministry of Education and Training, Hanoi, Vietnam

Vo Trung Hung
University of Danang, Danang, Vietnam

***Abstract* — Along with the development of the Internet, the data on the Internet is growing very fast. The searching demand on the Internet is also increasing rapidly and the quality of searching results returned is a challenge. In this paper, we would like to propose the solutions to build a storage system and search for Vietnamese mathematical documents based on semantic web. We propose the utilization of ontologies to store and represent the relationships between documents and the interaction with users through web environment. The system allows users to easily find the mathematical documents suitable for the purpose of the search.**

Keywords* — *Semantic web; mathematical document; content management; description logic; ontology

I. INTRODUCTION

Along with the rapid development of Internet, the document on the network is growing rapidly. We can easily found science, mathematics documents but the number of results returned too big to make the user takes a long time to select the appropriate document. For example, Google will return about 2,340,000 links to the documents when we search the phrase "Cauchy formula". One other example is zbMATH which isa service that is widely used in the fields of mathematics. This service maintains a database of more than 1.6 million mathematics documents and annual growth rate of 80,000 articles [4].

With huge amounts of data thus no mathematician can remember or have time to read all the documents. We should study to find a appropriate way to organize, store and utilize the knowledge most effectively. In addition, we also observed an increase in the complexity of the mathematical content, more and more the interdependence between different areas in and outside of mathematics.

Like the normal Web (World Wide Web), mathematics Semantic Web not allow the author published their online documents but also accumulate mathematical knowledge in the giant, decentralized and dynamic database. The author can note semantic annotations of their work in a special form, namely description logics. It allows the computer to understand the practical knowledge in fact there. Based on the semantic annotations, the server analyzes the content of mathematical knowledge and provides services on the mathematical semantic web. Mathematicians can access and use of more efficient mathematical data warehouses.

We propose a semantic web to support mathematicians in the effective management and mathematical knowledge obtainment by using Internet and computer systems.

In this paper, we present overview about the semantic web and propose a semantic web model for Vietnamese mathematical documents. We propose a general architecture and process for authors/users to upload and exploit mathematical documents on semantic web system. In the next sections, we introduce our experiment, some evaluations, and conclusion.

II. SEMANTIC WEB

A. Introduction to semantic web

Semantic web is formed from the idea of Tim Berners-Lee, the inventor of the WWW, URIs, HTTP, and HTML. Tim Berners-Lee has defined: *"Semantic Web is an extension of the current web in which information is clearly defined so that humans and computers can work together more effectively"* [1]. He has launched two issues of the semantic web is to make a better Web with collaboration environment and the computer can understand and automatically process the information on the Web.

As defined by World Wide Web Consortium (W3C), *"The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries"* [2]. Thus, the semantic web is a network of linked information so that they can be easily processed by computers on a global scale.

B. Semantic Web architecture

Semantic Web architecture consists of 7 layers [3]. In particular, the current system of Web (World Wide Web) is on the second layer. All layers of the semantic web are used to ensure the safety and becoming value of the best information.

Overall system architecture of Semantic Web as follows:

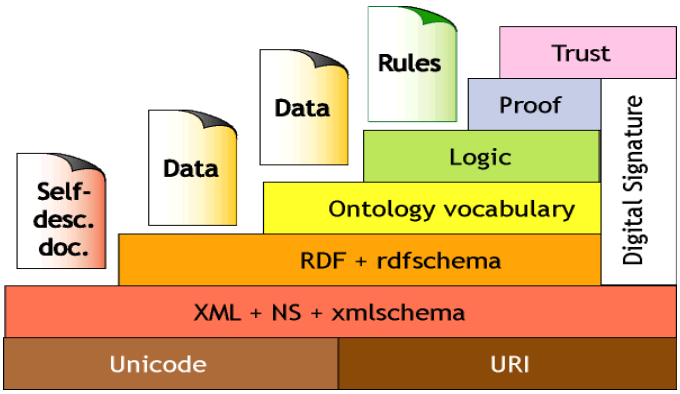

Fig. 1. Semantic Web architecture in layers

The role of the layers in this architecture is as follows:

1) Unicode and URI

Unicode is a computing industry standard for the consistent encoding, representation, and handling of text expressed in most of the world's writing systems.

URI (Uniform Resource Identifier) is a string of characters used to identify a name of a resource. Specifically, it is a short string allows identifying Web resources such as the string starts with "http:" or "ftp:" that we often see on the World Wide Web. Any one person can make a URI and own them and they are a technology base to build a global web system. World Wide Web system built on them and anything that has a URI is considered "on the Web".

URL (Uniform Resource Locator) is a special type of URI, specifically it is a network address.

URIref (URI reference) is an URI with an optional identification section at the end. Example: we have an URIref: http://www.example.org/Books#Ontology, which includes a URI: "http://www.example.org/Books" and part identification "Ontology" is separated by the symbol #.

According to convention, the namespace is the resources that generate the majority of resources, usually those which finishes URI symbol #. For example, "http://www.example.org/Books#" is a namespace. The resource no URIref is called white buttons. A white button indicates the existence of natural resources without a clear mention of resource URIref reference.

2) XML and XML Schema

XML (eXtensible Markup Language) is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. The design goals of XML emphasize simplicity, generality and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, it is widely used for the representation of arbitrary data structures such as those used in web services.

An XML Schema describes the structure of an XML document. The purpose of an XML Schema is to define the legal building blocks of an XML document, just like a DTD.

An XML Schema defines elements that can appear in a document; attributes that can appear in a document; which elements are child elements; the order of child elements; the number of child elements; whether an element is empty or can include text; data types for elements and attributes; default and fixed values for elements and attributes.

However, XML does not provide a complete solution to the requirements of the semantic Web. XML can only represent some semantic properties through its syntactic structure.

3) RDF and RDF Schemma

RDF (Resource Description Framework) is introduced by W3C to provide a standard syntax to create, change and use of annotations in Semantic Web. An RDF statement is a triad includes: topic is the resources that are described by attribute and object; properties represent the relationship between subject and object; also object here can be a resource or a value. The three components are all on the RDF URI.

RDFS (RDF Schema) is a simple ontology language of the Semantic Web, is considered a base language of the semantic Web. RDFS description language vocabulary on the RDF triple. It provides the following:

- Define the resource class;
- Define the relationship between the classes;
- Define the type of property that the above classes;
- Define the relationship between the properties.

4) Ontology Vocabulary

Ontology vocabulary is built upon RDF and RDFS layers. It provides flexible semantic representation for the semantic web resources and capable of inference. To build this ontology vocabulary, we can use the language to express them such as RDFS, OIL, DAML, DAML + OIL, OWL, ... These languages provide the ability to present and support different inference and they are based on description logical languages.

5) Logic

The presentation of the resources in terms of ontology vocabulary whose purpose is to do the inference. But the facts primarily based on logic. Therefore, the ontology is mapped to the logical description, namely description logic to be able to support the inference. Because description logic can present formal semantic (characterized by theoretical models) and can provide inference services. It is the basis for supporting to inference and understand the resources.

6) Proof

This layer is made inference laws. Specifically, from the available information we can deduce new information. Example: A is father of B, B is father of C then we have new information that A will be grandfather of C. The proof layer involves the actual deductive process as well as the representation of proofs in Web languages and proof validation. Currently the researchers are building for its law language such as SWRL, RuleML.

7) Trust

Ensure the reliability of the applications on the semantic web. Example: An application said *"x is green"*, another one said *"x is not green"*, so the semantic web is unreliable? The answer here is considered in the context. Each application on the semantic web will have a specific context, therefore the clause above may be located in different contexts while corresponding semantic differences; it remains true propositions, reliable in its context. To get the proof of reliability, the argument is applied is not monotonous and inspection mechanisms have proven technology combined with electronic signature to confirm reliability [11].

C. Description logic

Description Logics (DL) is a family of formal knowledge representation languages. It is more expressive than propositional logic. Additionally, it has more efficient decision problems than first-order predicate logic. DL is used in artificial intelligence for formal reasoning on the concepts

of an application domain (known as terminological knowledge). It is of particular importance in providing a logical formalism for ontologies and the Semantic Web [4]. Today, the description logic has become a foundation of the semantic Web by using it in the design of nature (ontology).

The idea for the development of Ontological Inference Layer (OIL) was derived from description logic.

III. MATHEMATICS SEMANTIC WEB

We propose to develop a system for mathematical documents based on Semantic Web. The aim of system is to create mathematical concepts, hierarchical ontologies that authors can easily contribute by knowledge models, model visually enhanced by the DL and the underlying technologies of the Semantic Web. Logical inference system can be operated on the basis of this huge knowledge and provide valuable services for everyone [11].

The reasons for the development of mathematics semantic web:

- Create a new service based on inference system to search on the ontologies and support mathematicians;
- Reuse efficiently ontologies of pre-defined knowledge from many different sources;
- Issue general knowledge of the mathematical concepts and access easily to information on the website.

A. *The model proposed*

We propose a general model for the operation of the system as follows:

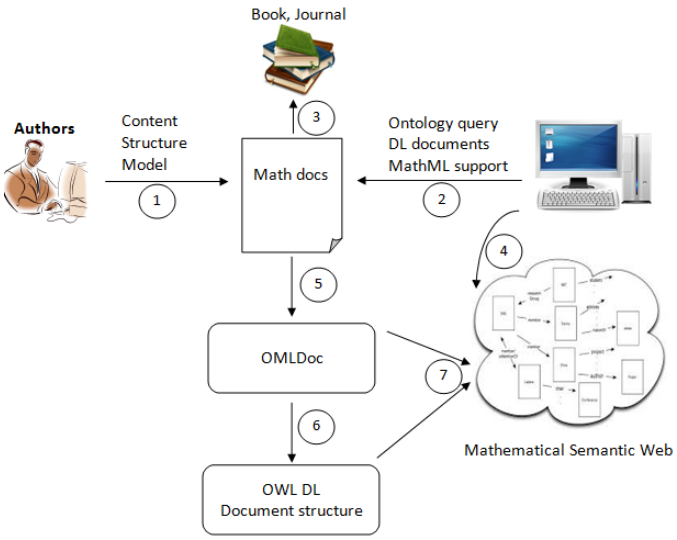

Fig. 2. Mathematical semantic web model

1) The author wrote a mathematical document, it includes:

- Mathematical document in LATEX or DOC formats and formulas in using MathML standard;
- Making the structure of the document in the format MathML;
- Modelization of mathematical knowledge based on Math Ontology Language.

2) Tools support ontology and different editing functions.

3) The document can be published under the format of books, newspapers and pushed the network to bring to users in other channels (such as web, electronic documents, ...).

4) Editor software can find on mathematical ontology.

5) Document MathML will be automatically transferred into OMDoc.

6) The ontology elements are extracted from the document automatically moved to an ontology OMDoc and OWL DL using standard forms defined in XSLT. This step is necessary to mathematics semantic web compatible with any one that is built on it. A OpenMath be built on mathematical ontology language will be embedded into OMDoc and not supported by the semantic web technology and existing services. The OWL DL ontology deduct allows us to inherit all the existing semantic web services.

7) In the both cases, documents are generated and OMDoc OWL DL ontology has been criticized as will be published on the Internet and from here it becomes part of the wemantic web system mathematically. OWL ontology document OMDoc and DL is clearly linked with the construction and closely related to each other.

B. *Ontology format*

Semantic Web ontologies described in OWL DL. Which syntax and semantics of the semantic web ontology is built in XML files. Thus, OWL DL is independent files, links through ontologies from outside but no reference to the content that has been used to define the ontology. The concept, the relationship and the circumstances specified in any ontology does not exist in a vacuum (vacuum is a space that does not contain theme). There exist a number of documents, some form of information or the ones not defined [6].

A format for mathematical ontology need full support for the original document and mathematics tight relationship between the original documents with defined ontologies. Open Mathematical Document (OMDoc) has been specially designed for this requirement and creates an optimal form for this purpose [7].

OMDoc is a semantic markup format for mathematical documents. While MathML only covers mathematical formulae and the related OpenMath standard only supports formulae and “content dictionaries” containing definitions of the symbols used in formulae, OMDoc covers the whole range of written mathematics [8]. OMDoc allows for mathematical expressions on three levels:

- Object level: content, formula written in MathML (the non-presentational subset of MathML), OpenMath or languages for mathematical logic.
- Statement level: definitions, theorems, proofs, examples and the relations between them.
- Theory level: A theory is a set of contextually related statements. Theories may import each other, thereby forming a graph. Seen as collections of symbol definitions, OMDoc theories are compatible to OpenMath content dictionaries.

On each level, formal syntax and informal natural language can be used, depending on the application.

OMDoc is built on XML and include OpenMath to present the mathematical formula. The format OpenMath defined by XML encoding for a model of abstract mathematics object. OMDoc is intended to be used for communication between mathematical web services.

For example:

- Declare a concept:

```
<type system="OntLang">
  <om:OMOBJ>
    <om:OMS cd="OntLang" name="concept"/>
  </om:OMOBJ>
</type>
```

- Declare a relation:

```
<type system="OntLang">
  <om:OMOBJ>
    <om:OMS cd="OntLang" name="relation" />
  </om:OMOBJ>
</type>
```

- Declare a specific case (instance):

```
<type system="OntLang">
  <om:OMOBJ>
    <om:OMS cd="OntLang" name="instance" />
  </om:OMOBJ>
</type>
```

In it, OntLang for "Ontology Language" and is the name that we include in the content of content dictionary. It contains the instructions of the description logic to build ontologies in OMDoc as symbols necessary for the expansion of ontology.

The using of the element symbol to declare lexical ontology is a natural choice to create this element. The example below declare concept Group, relations Equality and a KochSnowflake particular case.

```
<symbol name="Group">
  <type system="OntLang">
    <om:OMOBJ>
      <om:OMS cd="OntLang" name="concept"/>
    </om:OMOBJ>
  </type>
</symbol>
<symbol name="equality">
  <type system="OntLang">
    <om:OMOBJ>
      <om:OMS cd="OntLang" name="relation" />
    </om:OMOBJ>
  </type>
</symbol>
<symbol name="KochSnowflake">
  <type system="OntLang">
    <om:OMOBJ>
      <om:OMS cd="OntLang" name="instance" />
    </om:OMOBJ>
  </type>
</symbol>
```

OMDoc is a format for structural mathematical documents by providing elements to mark the definitions, theorems, proofs,... These elements often have a CMP and an arbitrary number FMP children. CMP for "Commented Mathematical Property" and the original binding of mathematical document, meaning a mix of text and formulas. FMP on another aspect for "Formal Mathematical Properties' and the constraints of an equivalent accuracy of content formats CMP corresponding element. CMP element containing mathematical content matches user requirements where the FMP element contents presented jointly represented in various formats.

The following example illustrates the integration in OMDoc:

```
<definition xml:id="some.def" for="#SomeConcept" type="simple">
<CMP>
 This is a definition ...
</CMP>
<FMP logic="dl">
 <!-- This is the Description Logic equivalent -->
</FMP>
</definition>
```

c. MathML

MathML (Mathematical Markup Language) is a markup language and it can specify mathematics in Web pages as text. The structure of MathML is not as TeX or LaTeX but can be easily used by the browser and can display clearly the mathematical formulas [9].

IV. EXPERIMENT

Based on the theory presented above, we have carried out construction of a mathematical semantic web system for Vietnamese (VM-SEMWEB) as follows:

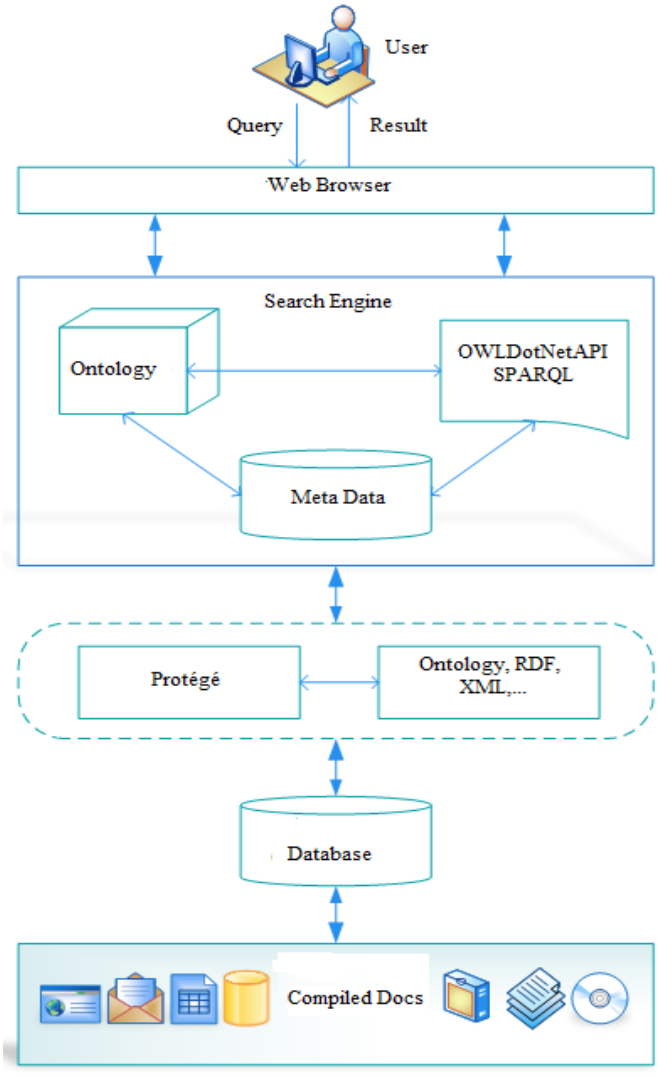

Fig. 3. Development process of mathematical semantic web

Web Browser: Play the role of communication between users with the system. The user can send a request and receive results through web interface. Especially, in this application, the user can find a mathematical document through formula.

Search Engine: This is the main function of program and it can do some tasks:

- Organize and store ontologies of mathematical documents.
- Execute the query and return results to users through a Web Browser. Specifically in this application, the system uses tools `OwldotnetApi` to access to ontology and return results through a web interface.

Searching based on semantics: search via hierarchical tree of information to access to information on a paper or a study area.

Algorithmic code fills all nodes from ontology files in the database:

```
public static IList<NodeModels> FillTree(IOwlGraph graph)
 {
   IList<NodeModels> result =
(IList<NodeModels>)HttpContext.Current.Session["FillTree"];          if
(result == null)
    {
      IList<NodeModels> list = new List<NodeModels>();
      IDictionaryEnumerator nEnumerator =
    (IDictionaryEnumerator)graph.Nodes.GetEnumerator();
        while (nEnumerator.MoveNext())
          {
             OwlNode _node =
          (OwlNode)graph.Nodes[(nEnumerator.Key).ToString()];
             OwlIndividual node = _node as OwlIndividual;
             if (node != null)
              {
                OwlEdgeCollection edges =
                   (OwlEdgeCollection)node.ChildEdges;
                foreach (OwlEdge e in edges)
                {
                   NodeModels n = new NodeModels();
                   n.ID = e.ID;
                   n.ParentID = node.ID;
                   n.Name = e.ChildNode.ToString();
                   list.Add(n);
                }
            }
      }
    HttpContext.Current.Session["FillTree"] = result =  list.ToList();
    }
 return result;
}
```

The code of the algorithm to fill in all the information from a string into the database:

```
public static IList<ArticleModels> SearchList(IOwlGraph graph, string
inputdata)
     {
        IList<ArticleModels> result = new List<ArticleModels>();
        IDictionaryEnumerator nEnumerator =
(IDictionaryEnumerator)graph.Nodes.GetEnumerator();
        while (nEnumerator.MoveNext())
        {
           OwlNode node =
(OwlNode)graph.Nodes[(nEnumerator.Key).ToString()];
           if (!node.IsAnonymous())
           {
              OwlEdgeCollection edges =
(OwlEdgeCollection)node.ChildEdges;
              foreach (OwlEdge i in edges)
              {
                 if
(i.ChildNode.ToString().ToLower().Contains(inputdata.ToLower()) &&
i.ID.ToLower().Contains("tieude"))
                 {
                    //select id
                    ArticleModels a = new ArticleModels();
                    a.ID = node.ID.Split('#')[1];
                    var _result = (from model in FillTree(graph)
                                   where (model.ParentID == node.ID &&
model.ID != "http://www.w3.org/1999/02/22-rdf-syntax-ns#type")
                                   select model).ToList();
                    foreach (var e in _result)
                    {
                       a.ParentID = e.ID;
                       if (e.ID.ToLower().Contains("tieude")) { a.Tieude =
e.Name.ToString(); };
                       if (e.ID.ToLower().Contains("hinhanh")) { a.HinhAnh =
e.Name.ToString(); };
                       if (e.ID.ToLower().Contains("noidung")) { a.Noidung =
htmlHelper.GetDescription(e.Name.ToString()); };
                       if (e.ID.ToLower().Contains("tacgia")) { a.Tacgia =
e.Name.ToString(); };
                       if (e.ID.ToLower().Contains("ngonngu")) { a.Ngonngu =
e.Name.ToString().Split('#')[1]; };
                       if (e.ID.ToLower().Contains("diachi")) { a.Diachi =
e.Name.ToString(); };
                       if (e.ID.ToLower().Contains("xuatban")) {
a.namXuatBan = e.Name.ToString(); };
                       if (e.ID.ToLower().Contains("tinhthanh")) { a.Ten =
e.Name.ToString().Split('#')[1]; };
                    }
                    result.Add(a);
                 }
              }
           }
        }
        return result;
     }
```

V. CONCLUSION

Semantic Web technology has enabled human beings to be able to add semantics to documents in a language that computer can understand. This causes the computer to understand the information on the Web, which makes searching fast and accurate. Semantic Web technology with data on the Web defined and linked in a way that computers can understand not just for display purposes, but for automation purposes, integration and reuse of data through different applications.

In this study, we have focused on the development of Semantic Web system to serve mathematics searches, look up information about documents by Vietnamese mathematicians. We studied and outlined the characteristics of the theoretical foundations of the Semantic Web, description logic, and how to build `OWL` ontology language. That is the most important component of web language. We studied also how to use the necessary tools to develop a semantic web application efficiently.

On experimental results, the development of information search system has proven mathematical theory research platform combining development model and support tools developed with `.NET` technology, totally can build a successful Web 3.0 applications. This application demonstrates the technological superiority of Web 3.0 with the Web technology has built before.

Our proposal searching system include the functionality allowing users to enter new data in Web pages, view, and search for information. The system also allows accessing data from files and resources available on the Internet to provide richer data.

However, the research and experimentation here stands at trial, document volume put less and can continue research to develop a number of other functions such as automatically find and detect online accounting documents (crawler), automatically creating ontologies from documents, enhance the inquiry function, automatic data conversion formula to `MathML` [10] [12].

REFERENCES

[1] T. Berners – Lee, J. Hendler, O. Lassila. The Semantic Web. Scientific American, vol.248, 2001, 28 – 37.

[2] Grigoris Antoniou and Frank Van Harmelen. A Semantic Web Primer. MIT Press, 2004.

[3] Toby Segaran, Colin Evans, Jamie Taylor. Programming The Semantic Web. O'Reilly – Media, July 2009.

[4] Gert-Martin Greue, Zentralblatt MATH, EMS Newsletter, EMS publíhed, March 2012

[5] D.E. Knuth, The TeXbook, Computers and Typesetting, Volume A, Published by Addison-Wesley, ISBN 0-201-13448-9, 1984

[6] Tom Gruber. Ontology. Entry in the Encyclopedia of Database Systems, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag, 2008.

[7] M. Kohlhase. OMDoc. An Open Markup Format for Mathematical Documents [version 1.2]. Springer-Verlag GmbH, 2006.

[8] Michael Kohlhase and Florian Rabe. Theory morphisms as first-class objects in OMDoc. To be published - just borrowed some formulations.

[9] D. Carlisle, P. Ion, R. Miner, Mathematical Markup Language (MathML) Version 2.0 (Second Edition), 2010

[10] David Carlisle, Patrick Ion, Robert Miner, Mathematical Markup Language (MathML) Version 2.0 (Second Edition), 2010.

[11] T. Berners-Lee, D. Connolly, HyperText Markup Language Specification, Published by IETF, 2010.

[12] Le Thanh Nhan, Vo Trung Hung, Cao Xuan Tuan, MATHIS – (MATHematic Information web Services) – semantic annotation and search tools for scientific resources on the web, Journal of Science and Technology, University of Danang, Volumn 4(39), 2010