Antology : Development, Deployment and Merging Aspects in Semantic Web : *An Overview*

Vishal Jain*
Sanjay Kumar Malik**
Pankaj Lathar***

Abstract

Semantic Web is a active and growing in many directions where Ontologies plays a significat role. Ontologies aims to capture knowledge of a given domain in a generic and formal way, to be reused and shared across applications and by groups of people and information which can be understood by both humans and machines. Therefore, Ontology development, deployment and merging of more than one ontologies is an important issue of concern to realize the goal of semantic web. In this paper, we have discussed some aspects of an Ontology development, deployment and merging. Ontology development has been highlighted by taking an example of Educational Institute Departments, deployment issues in concern to Jena and pellet, merging using prompt plug-in of Protégé.

Keywords : Semantic Web, Ontology, OWL, Protégé, Jena, Pellet, Prompt.

1. Introduction

Semantic Web identifies a set of technologies, tools and standards which form the basic building blocks of an infrastructure to support the vision of the Web associated with meaning [2]. The goal of semantic web may be realized by ontology significance to share common understanding of the structure of information among people or software agents and to analyze and reuse domain knowledge.

Ontology development includes various stages like determine scope, reuse, terms, define properties, taxonomy, facts and instances etc. Protégé is one of the most widely used tools to develop an Ontology. After developing an Ontology the next stage is Ontology deployment, which will enable the interaction between the Ontology and the application system. Various frameworks may be used for deployment like Jena, Pellet, and Sesame etc. But, defining Ontologies for a single web entity is not sufficient and there is need of higher level of abstraction so that information spread over multiple sites can be merged or united.

Ontology merging is of the most crucial and challenging task which needs to be discussed.

*M.Tech(CSE) III Sem, University School of Information Technology, GGS Indraprastha University, New Delhi, India
**University School of Information Technology, GGS Indraprastha University, New Delhi, India
***Government Engineering College, Jaffarpur, Delhi, India

2. Semantic Web, Ontology and OWL

In current Web, there are different types of data available in the form of structured data, semi- structured data and unstructured data. Unstructured data includes text files, videos and Power Point Slides, Semi Structured data are data that have some structure but are not rigidly structured and includes Curriculum Vitae whereas Structured data are very rigid and describe objects using strongly typed attributes in form of tuples or records [2]. Ontology may be used to define structured data on the Web. The popularity of Semantic Web grows with the emergence of something known as Ontologies [3]. Ontology plays an important role in increasing magnitudes with the performance of information processing system such as information integration, taxonomies - based document classification and information retrieval system [1]. Ontologies are expressed in terms of Ontology language, which build upon the layers of successful technologies such as OWL (Web Ontology language), XML (Extensible Markup Language), URI (Uniform Resource Identifier) and RDF (Resource Description Framework) [4]. Ontologies are used to represent data in more expressive and more meaningful way on web and OWL language may be used for its achievement. OWL is a vocabulary extension of the RDF, which is general purpose language for representing the web metadata and its main purpose is to represent the semantics (meaning).

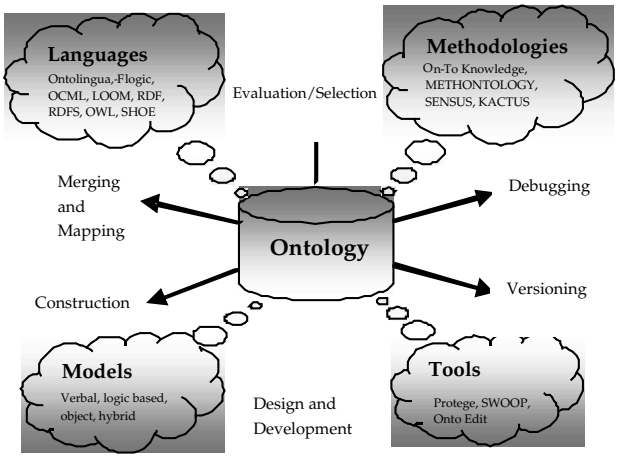

Figure 1 : "Ontology and its Constituents" [5]

3. Building Ontologies

One of the major functions of the specific Ontology is to define set of classes that together cover a domain of interest. For example, a says Ontology would need to cover customers, purchase orders, sales receipts, catalogues, inventory items, and so on. Many Ontologies are already in existence which cover variety of areas and many more will be added in future. Specific Ontologies are to be constructed with known vocabularies and rules of construction [10]. There are different tools to develop Ontologies Viz. Ontolingua, WebOnto, ProtegeWin, Adaptiva, Semantic Works 2008, COE, Conzilla2, HOZO, Onto Track, SWOOP, OntoSaurus, ODE and KADS22. Protégé is one of the most used to develop Ontology. Here, we are using Protégé version 3.4.1 to develop an Ontology on "Institute Departments"

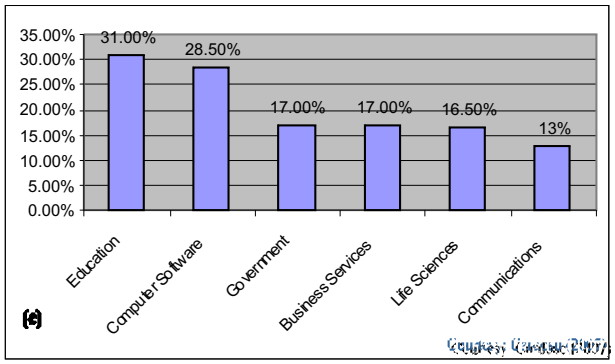

Figure2 "Development of ontologies in different domains (Source: Jorge Cardoso, "The Semantic Web Vision: Where are We?" IEEE Intelligent Systems, September/October 2007, pp.22-26, 2007.)"

Figure 2 shows that maximum numbers of Ontologies are being developed in Education Domain. Therefore, we choose to develop ontology in Education domain as follows:

3.1 "Institute Departments" Ontology

Following is the sample snapshot of an Ontology on different departments of an institute :

3.1.1 Screenshot

Figure 3 depicts the relationship between super classes like Institute Departments and their sub classes like Chemistry, Computer Science and Electronics etc. Here, Institute Departments inherits to Chemistry, Computer Science, Electronics, Mechanical departments. Computer Science inherits to Courses, Laboratory, persons. Courses inherit to Graduate and Post Graduate. Person inherits to Staff and Student.

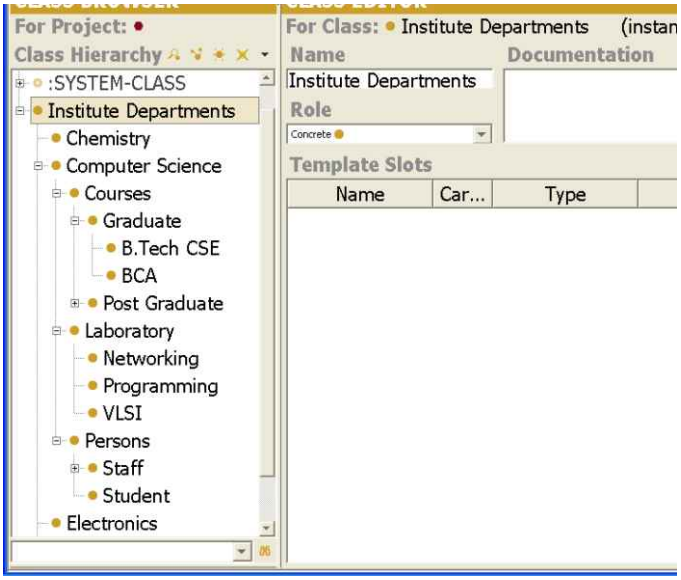

Figure 3 "Institute Departments Ontology" "

Various code snippets are generated as:

3.1.2 Code Snippets

3.1.2.1 XML Code Snippet

XML, or Extensible Markup Language, is a platform-independent way to represent data.

<?xml version="1.0" ?><knowledge_basexmlns="http://

```
protege.stanford.edu/xml"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://protege.stanford.edu/xml
http://protege.stanford.edu/xml/schema/protege.xsd">
<class><name>:THING</name>
<type>:STANDARD-CLASS</type><own_slot_value>
<slot_reference>:ROLE</slot_reference>  <value value_
type="string">Abstract</value></own_slot_value>
</class>
<class><name>Institute Departments</name>
<type>:STANDARD-CLASS</type><own_slot_value>
<slot_reference>:ROLE</slot_reference>
<value value_type="string">Concrete</value>
</own_slot_value><superclass>:THING</superclass>
</class>
```

3.1.2.2 OWL Code Snippet

Ontology Language is a markup language for publishing and sharing data using Ontologies on the internet.

```
<?xml version="1.0"?><rdf:RDF xmlns:xsp="http://www.owl-
ontologies.com/2005/08/07/xsp.owl#"
xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
xmlns="http://www.owl-ontologies.com/Ontology
1272349221.owl#"
xmlns:swrl="http://www.w3.org/2003/11/swrl#"
xmlns:protege="http://protege.stanford.edu/plugins/owl/pr
otege#"xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"xmlns:xsd="http://www.w3.org/2001/XML
Schema#"
mlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl="http://www.w3.org/2002/07/owl#"xml:base="ht
tp://www.owl-ontologies.com/Ontology1272349221.owl">
<owl:Ontology rdf:about=""/>
<owl:Class rdf:ID="Administrative">
<rdfs:labelrdf:datatype="http://www.w3.org/
2001/XMLSchema#string"
>Administrative</rdfs:label> <rdfs:subClassOf>
<owl:Class rdf:ID="Staff"/></rdfs:subClassOf>
</owl:Class>
```

3.1.2.3 RDF Code Snippet

RDF is used to describe the resources which are available on the web and also identify the relationship between them. The resources described by the RDF Model is Known as a RDF Instances [6]. It enables easy integration of triples contained in any number of documents distributed across the Web.

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
<!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-
ns#'> <!ENTITY rdf_ 'http://protege.stanford.edu/rdf'>
 <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
]><rdf:RDF xmlns:rdf="&rdf;" xmlns:rdf_="&rdf_;"
 xmlns:rdfs="&rdfs;">
<rdfs:Class rdf:about="&rdf_;Administrative"
rdfs:label="Administrative">
<rdfs:subClassOf rdf:resource="&rdf_;Staff"/>
</rdfs:Class>
```

4. Ontology Deployment

Next stage is translation with the code after Ontology building. Translation process is used to enable a type of high level Meta programming.

Figure 4 depicts steps of how the required web output is obtained from the Ontology engine through various translation processes which involves taxonomy, model, and visualization translations. For translating the Ontology, we deploy Ontology with the Online System.

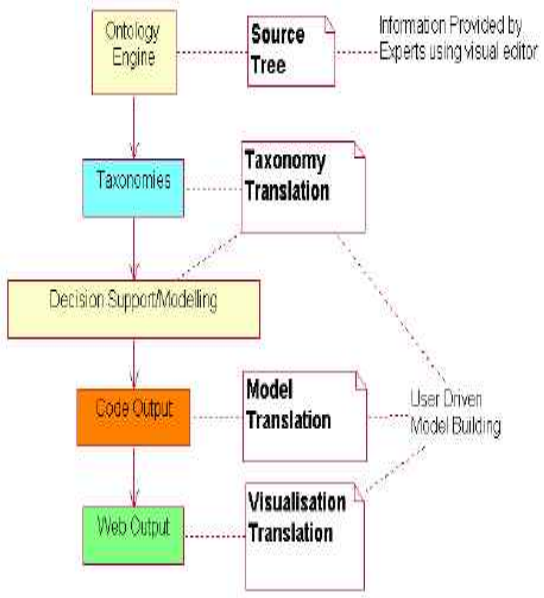

Figure 4 "Translation process" [9]

4.1 Deploying Ontology using Jena and pellet

Steps in using Jena framework involve: creating the object of ModelMaker class and creating the basic model by using ModelMaker object. The model may be populated from a file, from a URL and by adding statements directly [12]. A few concerned codes are :

```
Code to Populate the model by URL  [12]
Public void addfromurl () {
Model.read(http://xmlns.com/electronics/index.rdf)
```

Code to populate the model from specific location [12]

```
Public void addfromfile() throws IOException{
InputStream instance =  FileManager. get(). open("C:\Program
Files\Protege_ 3.4.1\electronics.rdf");
Model.read(instance,namespace);Instance.close();}
```

Code to Integrate the Pellet Reasoner- [12]

```
Public InfModel pelletreasoner() {
Reasoner reasoner = PelletReasoner Factory.theInstance().
create();
reasoner = reasoner.bindSchema(model)
```

Figure 5, shows the use of Protégé, Jena and Pellet tools in Semantic Web to deploy Ontology. Protégé is used to develop Ontology and Jena is a Semantic Web framework, used to manipulate ontology with the project system. It also provides interaction with the Semantic Web information like Ontology and provides query retrieval through SPARQL and Pellet is a reasoner, which provides full expressivity of OWL [12].

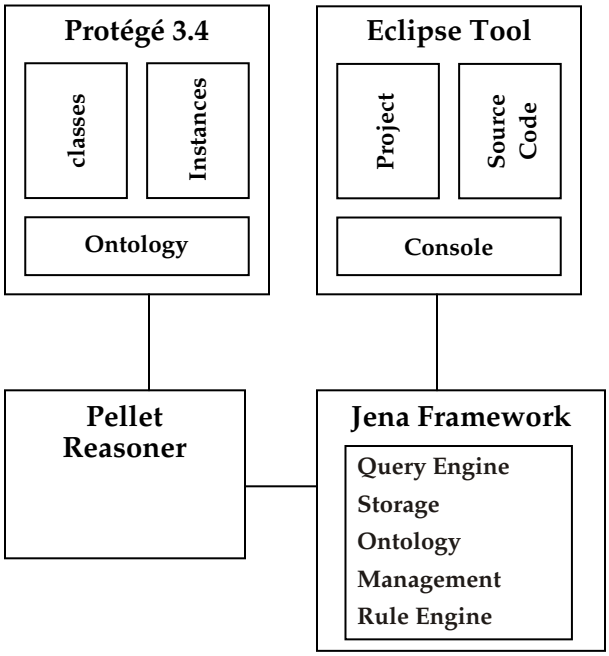

Figure 5 : "Semantic Web Development Environment" [12]

For deploying Ontology by adding statement, we can use OWL light browser, Figure 5, which is available at : pellet.owldl.com/ ontology-browser.

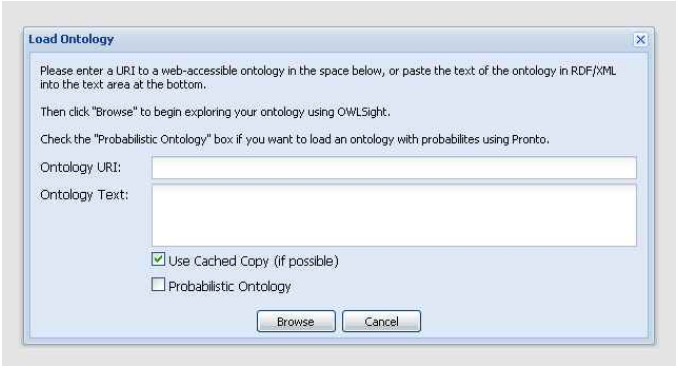

Figure 6 : "OWL Light Browser" [11]

5. Merging Ontology

There are generally two types of automatic ontology manipulation or management: ontology alignment and ontology merging [13, 14, 15]. Merging of Ontologies forms a new Ontology from two previously defined Ontologies and hence represents the combined knowledge domain of the two Ontologies. This makes it possible to define common concepts of two Ontologies represented by a single Ontology rather than using them separately. It is required to form larger and better modeled Ontology from smaller and not well defined Ontologies in order to present an effective representation of the knowledge domain and in a more structured presentation. The merging of two related Ontologies is obtained by taking the union of the terms and axioms defining them. Ontology merging defines the act of bringing together two conceptually divergent Ontologies or the instance data associated to two Ontologies.

5.1. Using Prompt Tab in Protégé for merging

In Protégé, Prompt plug-in may be used to merge two or more Ontologies. Prompt tab provides a semi-automatic approach to ontology merging and alignment. Prompt performs some tasks automatically and guides the user in performing other tasks for which his intervention is required. Prompt also determines possible inconsistencies in the state of the ontology, which result from the user's actions, and suggests ways to remedy these inconsistencies.

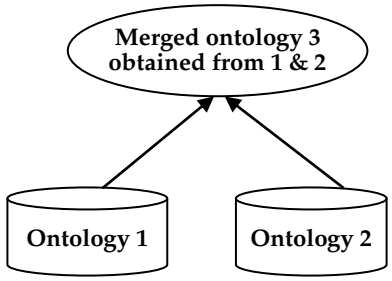

Figure 7 : "Merging of two Ontologies"

The Figure 7 represents two ontologies i.e. ontology 1 and ontology 2 that have someting in common to their knowledge domain. We perform the merging operation on the two onologies to obtain a third new ontology i.e. ontology 3.

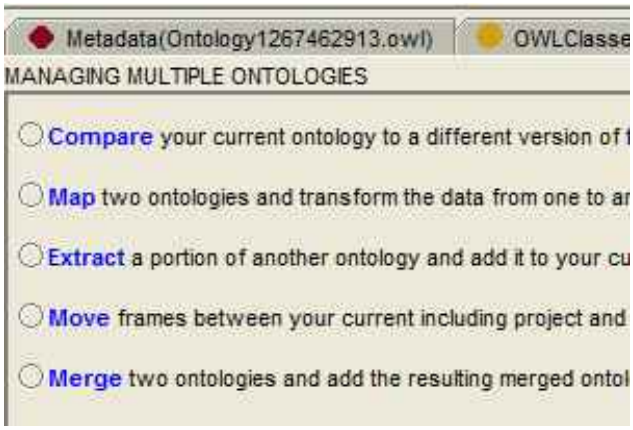

Figure 8 : "Prompt Menu – protégé Plugin for merging"

Figure8 show the prompt tool, plug-in of protégé. Prompt takes two ontologies as input and guides the user in the creation of a single merged ontology as output. Firstly Prompt creates an initial list of matches based on class names. Then the following cycle happens: the user triggers an operation by either selecting one of Prompt's suggestions from the list or by using an ontology-editing environment to specify the desired operation directly; and Prompt performs the operation, automatically executes additional changes based on the type of the operation, generates a list of suggestions for the user based on the structure of the ontology around the arguments to the last operation, and determines conflicts that the last operation introduced in the ontology and finds possible solutions for those conflicts.

6. Conclusion

This paper highlights and overviews some aspects of Ontology development, deployment, and merging. An Ontology in education domain has been created in protégé version 3.4 and their corresponding XML, OWL and RDF snippet codes created

have been presented. Some methods in Ontology deployment have been highlighted and Ontology merging has been discussed using Prompt Plug-in of Protégé tool. It may be helpful for researchers in some way working in Ontology development, deployment and merging.

7. References

1. *Panita Yongyuth, Dussadee Thamvijit, Thanapat Suksangsri, Asanee Kawtrakul, Sachit Rajbhandar, Margherita Sini, and Johannes Keizer, "The AGROVOC Concept Server Workbench: A Collaborative Tool for Managing Multilingual Knowledge", Iaald, AFITA WCCA 2008 World Conference on Agricultural Information and IT*
2. *Jorge Cardoso, "The Syntactic and the Semantic Web", IGI Global, Semantic Web Sevices*
3. *Kamidi Suresh, Sanjay Kumar Malik, Nupur Prakash, S.A.M. Rizva, "Role of Ontology Editors : Ontology Design" in 2008 "International Conference on Semantic Web and Web Services, SWWS'08", Monte Carlo Resort, Las Vegas, Nevada, USA : July 14-17, 2008*
4. *F.Manola and E.Miller, RDF Primer, Http://www.w3.org /TR/rdf-rimer/,Feb,2004*
5. *Ontology Revisited: Design and Development, Neeraj Seth, B.Vinoth Raj, Sanjay Kumar Malik, Nupur Prakash, SAM Rizvi*
6. *YounHee Kim, ByungGon Kim , AND HaeChull Lim ," The Index Organization for RDF and RDF schema", Basic Research Program of Korea Science & Engineering Foundation, ICACT 2006*
7. *http://www.owlpages.com/the-owls/Web+Ontology+ Language.html*
8. *www.ibm.com/developersworks/xml/library/j-jena*
9. *Peter Hale, "Automated Generation of Modelling Programs Using Jena and Protégé" (http://www.cems. uwe.ac.uk/~phale/)*
10. *Thomas B. passin, "Explorer's Guide to the Semantic Web", Manning Publications, pp 150-151*
11. *http://pellet.owldl.com/ontology-browser/*
12. *John Hebeler, Matthew Fisher, Ryan Blace, Andrew Perez-Lopez, "Semantic Web Programming", Wiley Publications*
13. *F. Noy and N. Musen, "An algorithm for merging and aligning ontologies: Automation and tool support", 1999.*
14. *Andrew August Carbonetto, "Ontology Alignment in the Presence of a Domain Ontology- Finding Protein Homology" Thesis, The University Of British Columbia, May, 2008*
15. *N. Noy and M. Musen,"The prompt suite: Interactive tools for ontology", merging and mapping, 2002.*