

Ontology Design and Development: Some aspects : An overview

Sanjay Kumar Malik¹ · Nupur Prakash² , SAM Rizvi³

¹ University School of Information Technology, GGS Indraprastha University, New Delhi

² University School of Information Technology, GGS Indraprastha University, New Delhi

³ Deptt. of Computer Science, Jamia Millia Islamia, New Delhi

sdmalik@hotmail.com¹, nupurprakash@rediffmail.com², samsam_rizvi@yahoo.com³

Abstract

Ontology play a key role in information exchange, use and re-using knowledge, shared and common understanding of a domain that can be communicated between people and across application systems which is the goal of semantic web. Ontology may be described as sharable conceptualization of a specific domain of interest in a machine-understandable format. Therefore, Ontology design and development is an important issue of concern for knowledge sharing in a structured way for which various tools are available. Protégé is one of the most commonly used editor. This paper discusses and highlights some aspects about the design and development of an Ontology using Protégé 3.4 using an example. It presents the query retrieval using query tab of protégé and also illustrates the TGviz tab which is used to provide the route of the ontology with a graph to reach to any classes or subclasses.

Keywords: Semantic Web, Ontology, Protégé, Query, TGviz .

1. Introduction

The aim of Semantic web is to tackle the growing problems of surfing the expanding web space, where presently most web resources can be found by syntactical matches i.e; keyword search. The semantic web relies heavily on Ontologies that structure underlying data for the purpose of comprehensive and transportable machine understanding. They properly define the meaning of data and metadata. [13]. Ontology building plays a significant role in the extraction of knowledge from different web sources.

Section 2 of the paper discusses about iterative steps of ontology construction, states of ontology life cycle, and reasons, approaches and problems of ontology development. Section 3 illustrates the starting of the new project and choosing the sub language using protégé tool. In Section 4, an example of “University School of Information Technology” Ontology design and development has been illustrated using Protégé. Super class and Sub class hierarchy and instances of USIT Ontology have been presented with code snippets of OWL, RDF and XML generated. Section 5 illustrates the Query retrieval process. Section 6 presents the use of TGviz

tab which illustrates the route of the above ontology with the graph focusing on “Dean” subclass.

1.1 Semantic Web and Ontology

Semantic Web is the new-generation Web that tries to represent information such that it can be used by machines not just for display purposes, but for automation, integration, and reuse across applications [2]. The popularity of Semantic Web grows with the emergence of something known as Ontologies. Ontologies are the most important tool in knowledge representation, as they allow to logically relate large amounts of data [2].

An ontology describes basic concepts in a domain and defines relations among them and consists of a finite list of terms and the relationships between these terms. The terms denote important concepts (classes of objects) of the domain. An ontology is a model of a particular domain, built for a particular purpose and is the statement of a logical theory [4].

Researchers have created the vision of semantic web [15], where data has structure and Ontologies describe the semantic of the data. Ontologies allow users to organize information into taxonomies of concepts, each with their attributes, and describe relationships between concepts [15].

The ontology design and development process is the set of activities for building ontologies and may be classified into following categories [1]:

- Ontology Management activities: Scheduling, control and quality assurance.
- Ontology development oriented activities: predevelopment, development, and post development activities.
- Ontology support activities: a series of activities including knowledge acquisition, evaluation, integration, merging, alignment, documentation and configuration management.

2. Ontology Design and Development

When an Ontology becomes large, it needs to be treated like a project. The design of an ontology is an iterative maturing process. This means the ontology will become to full development, become mature, by evolving through intermediate states to reach a desired or final condition[12].

While making design and development decisions, we need to choose what and how to represent something

in an ontology whose purpose is knowledge sharing and interoperation among programs based on a shared conceptualization.[8].

Iterative steps of ontology construction, states of ontology life cycle, and reasons, approaches and problems of ontology development are given as below:

2.1 Ontology Construction: Basic Steps

Ontology permits sharing common understanding of the structure of information among people and software agents and since there is no unique model for a particular domain, ontology development is best achieved through an iterative process. An iterative approach for building Ontologies starts with a rough first pass through the main processes as follows: [14] [5].

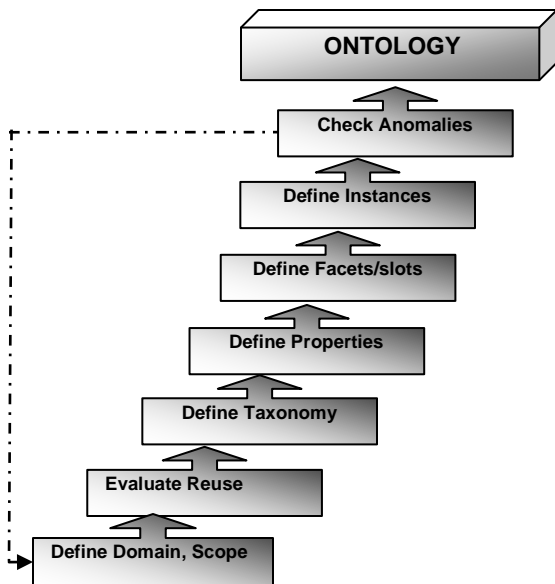


Figure 2.1: Ontology Construction Steps[14] [5]

Developing ontology includes following steps [14][5]:

- Start with defining domain and scope of the ontology which will include: What will ontology cover, how it will be used and who will use it.
- Check to see if it can be extended or reused.
- To list all terms for overlapping concepts.
- Define top-down(starts by defining general concepts) and bottom-up process(starts with definition of most specific classes)
- Define properties for internal structure of concepts.
- Define facets where properties add cardinality, values and characteristics.
- Slots cardinality: the number of values a slot has.

- Define individual instances of a class which requires choosing a class, creating an individual instance of that class and filling in the slot values.
- Check for anomalies such as incompatible domain and range definitions.

2.2 Steps of Ontology Life Cycle

Ontology life cycle moves through following states with support activities of knowledge acquisition, documentation and evaluation: [12]

- Specification
- Conceptualization
- Formalization
- Integration
- Implementation
- Maintenance

2.3 Ontology Development: Reasons

Why someone will create an ontology? Some of the reasons are: [5]

- To share common understanding of the structure of information among people or software agents.
- To reuse of domain knowledge
- To make domain assumptions explicit.
- To analyze domain knowledge.

2.4 Ontology Building: methodologies/approaches

Different approaches are used for building Ontologies like manual extraction of common sense knowledge from different sources. It includes various stages like: identifying the purpose of the ontology (why to build it, how will it be used, the range of the users), building the ontology, evaluation and documentation [10]. Building Ontologies is further divided into three steps: ontology capture, ontology coding and possible integration with existing Ontologies as shown in figure 2.2: [10]

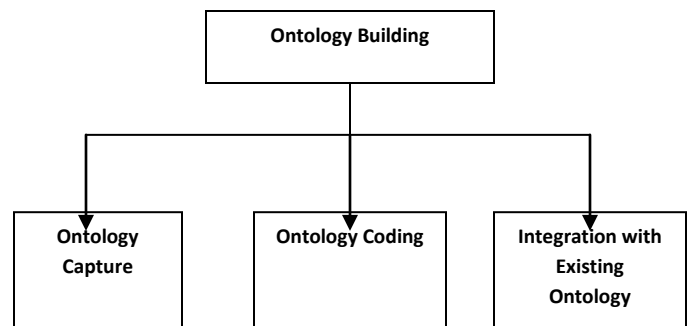


Figure 2.2: Ontology Building Stages

- **Ontology-Capture:** Key concepts/ relationships identified, involved actors agree on the definitions and terms.

- **Ontology Coding:** to represent defined conceptualization in some formal language.
- **Integration** with existing Ontologies.

2.5 Ontology Construction: Problems/tasks [10]

Some problems or tasks involved in Ontology construction are: [10]

- **Extending the existing ontology:** using some text.
- **Learning Relations** between concepts for an existing ontology.
- **Ontology construction based on clustering:** Split each text document into sentences, parse the text and apply clustering.
- **Ontology construction based on semantic graphs:** Parse the text documents, perform coreference resolution, anaphora resolution, extraction of subject-predicate-object triples, and construct semantic graphs.
- **Ontology construction from a collection of news:** based on named entities.

3. Starting with Protégé for new project

Design criteria for Ontologies may include clarity, coherence, Extendibility, minimal encoding bias and minimal ontological commitment [8].

The development of ontologies demands the use of various software tools which may be applied to several stages of the ontology life cycle including the creation, implementation, and maintenance of ontologies.

These tools are used for building a new ontology either from scratch or by reusing existing Ontologies. Protégé is one of the most widely used ontology development tools which is a free, open-source platform that provides a growing user community with a suite of tools to construct domain models and knowledge-based applications with Ontologies [3].

It is an ontology editor which we can use to define classes and hierarchy, slots, and slot-value restrictions, relationships between classes and properties of these relationships [11].

Developing an ontology includes: [3]

- Defining classes
- Arranging the classes in a subclass, superclass hierarchy
- Defining slots and describing allowed values for these slots.
- Filling in the values for slots for instances

Figure 3.1 depicts that how we may open the existing project or may create the new protégé project in pprj format.

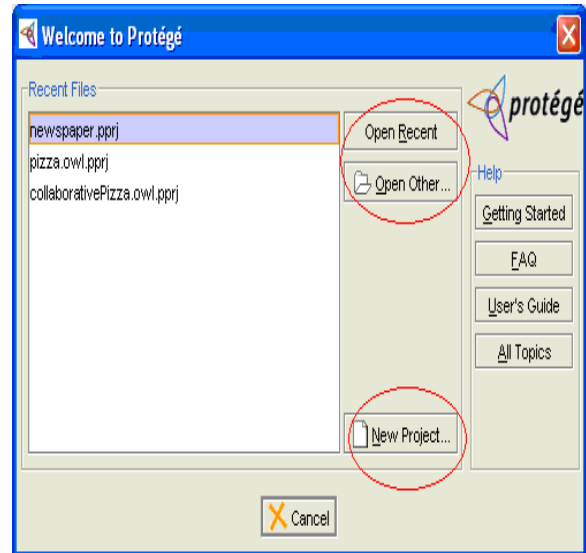


Figure 3.1: Open the existing project or create the new protégé project

The above figure illustrates about selecting the option of either opening the existing project or creating the new protégé project by clicking on "Open Recent" or "New Project" options respectively.

In case, we select the option of "New Project", then either of the options viz, OWL DL, OWL Lite, or OWL Full etc, any one is selected based on the application.

Figure 3.2 refers about which OWL/RDF dialect to be used i.e.; which format of saving the new project.

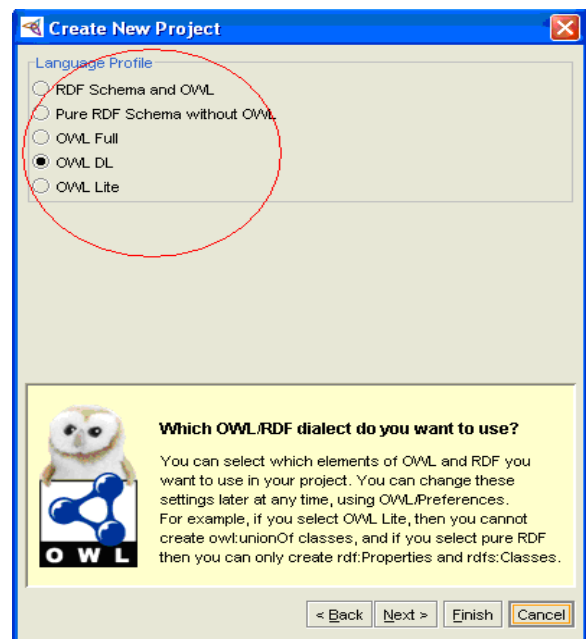


Figure 3.2: Selecting OWL/RDF dialect to be used

3.1 OWL Ontologies [7]

OWL is the most recent development language which is divided into three sub-languages: OWL-Lite, OWL-DL and OWL-Full. OWL-Lite is the syntactically simplest sub-language and is intended to be used in situations where only a simple class hierarchy and simple constraints are needed. OWL-Full is the most expressive OWL sub-language and is intended to be used in situations where very high expressiveness is more important. OWL-DL is much more expressive than OWL-Lite and based on Description Logics (hence the suffix DL) and are therefore amenable to automated reasoning. [7]

3.2 Choosing Sub-Language to Use

A few simple rules for choosing the sub-language to be used are: [7]

- The choice between OWL-Lite and OWL-DL may be based upon whether the simple constructs of OWL-Lite are sufficient or not.
- The choice between OWL-DL and OWL-Full may be based upon whether it is important to be able to carry out automated reasoning on the ontology or whether it is important to be able to use highly expressive and powerful modeling facilities such as meta-classes (classes of classes) [7].

4. Example: “University School of Information Technology” Ontology

An example of “University School of Information Technology” Ontology development has been illustrated using Protégé. The ontology is composed by a set of terms semantically related with its attributes, properties and relationships. The first step is to create the classes that describe the major concepts and then adding the properties (slots) and features (facets) to the classes to describe the internal structure of these concepts. Below is the snapshot showing a few classes and subclasses of the above mentioned ontology. Figure 4.1 shows the Sub Class and Super Class hierarchy of USIT ontology. The super class is *USIT* and the sub classes of *USIT* super class are *Persons* and *Programs* which also have further subclasses like Staff, Graduate etc respectively as shown in above figure 4.1.

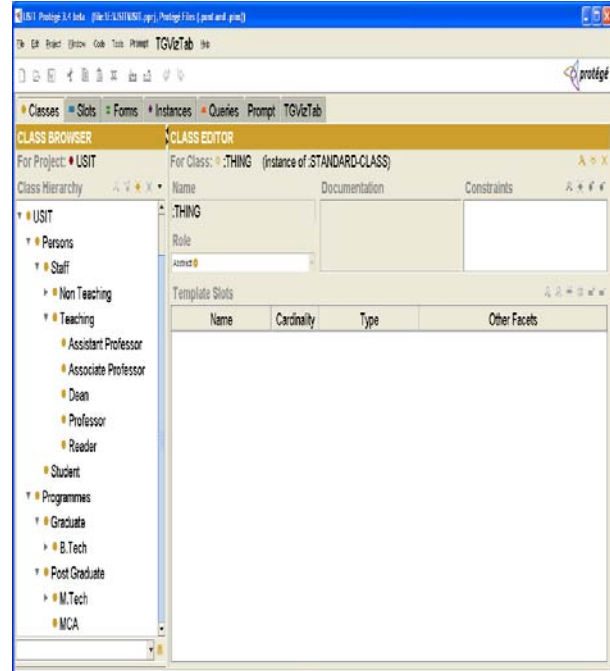


Figure 4.1: Super class and Sub class hierarchy of USIT Ontology

Below is the snapshot of “Instances of USIT Ontology” which give details of the corresponding classes that may be useful in finding some information about the class like figure 4.2 shows the details of Prof. B.V.R. Reddy of “Dean” subclass with the slots value like: Add, Name, DOB, Qualification, DOJ, Ph.No., Sal, and ID.etc.

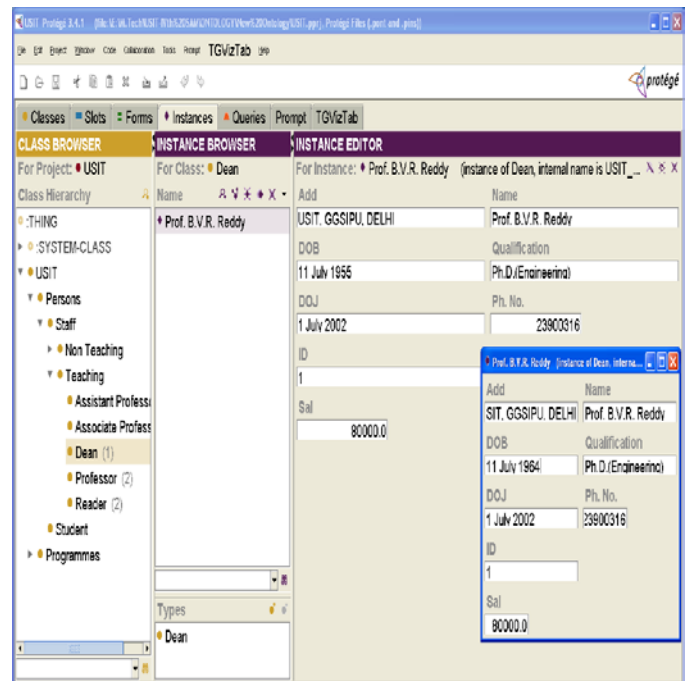


Figure 4.2: Instances of USIT Ontology

Following snippet codes of OWL , RDF and XML are generated.

OWL Code Snippet

```
<?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;"
<rdf:Property rdf:about="&rdf_;ASIP_ADD" rdfs:label="ASIP_ADD">
<rdfs:domain rdf:resource="&rdf_;Assistant_Professor"/>
<rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
```

RDF Code Snippet

```
<?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;"
<rdf_:Assistant_Professorrdf:about="&rdf_;USIT_Class0" rdf_:ASIP_ADD="USIT, GGSIPU, DELHI"
rdf_:ASIP_DOB="5 June 1972"
rdf_:ASIP_DOJ="July 1999" rdf_:ASIP_ID="1"
rdf_:ASIP_Name="Mr. sanjay Kumar Malik"
rdf_:ASIP_Quali="PHD" rdf_:ASIP_Sal="40000.0"
rdf_:ASIT_PH="23900345" rdfs:label="1Mr. sanjay Kumar Malik"/>
```

XML Code Snippet

```
<?xml version="1.0" ?>
<knowledge_base
xmlns="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>
]
```

5. Query Retrieval Process

Figure 5.1 below shows that how we run the query using the query tab and find the particular information about any particular instances or classes. When the query is run giving the value of ID as "1", an instance of "Dean" subclass is created with its slot values like, Name, Add, DOB, DOJ, Ph.No., etc .

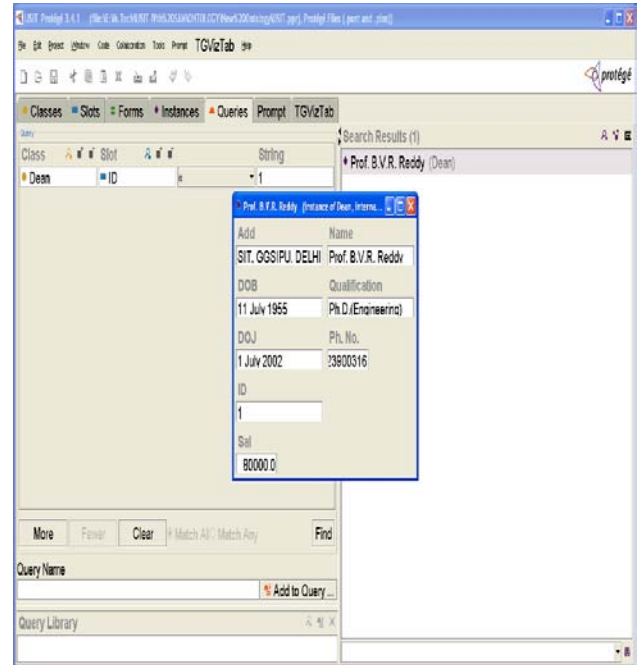


Figure 5.1: Running the query and getting the result

6. Using TGviz tab for route graph

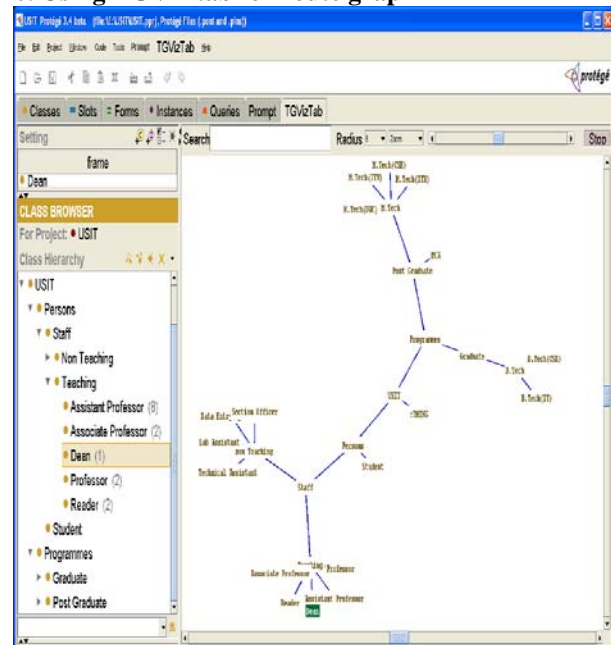


Figure 6.1: Graph corresponding to Dean sub- class using TGviz tab

Figure 6.1 above shows a graph corresponding to the class “*Dean*” using the TGViz tab which provides the route of the ontology and shows the possible options of how one can reach to any class from any other class or subclass. Using the radius option, one can vary the path distances between the nodes/classes.

7. Conclusions and Future Work

The paper discusses and highlights about some issues and methodology of design and development of an ontology and illustrates the usage of protégé for starting of the new project with an example. Super class and Sub class hierarchy and instances of Ontology have been presented with code snippets of OWL, RDF and XML generated. Query retrieval process and use of TGViz tab illustrating the with graphical route of ontology with some classes and subclasses. It may be useful for researchers who would like to start with design and developing an Ontology and it’s related aspects. In future, it may be extended to deployment, evaluation and other ontology issues.

8. References

- [1] Jorge Cardoso, “Semantic Web Services: Theory, Tools and Applications”, Information Science Reference, pp 47-49.
- [2] Vladan Devedzic. “Semantic Web Education” Springer, springer.com ,2006, Pages 33 – 50.
- [3] Eliza Sachs, “Getting Started with Protégé-Frames”, The Protégé Project-<http://protege.stanford.edu>, .
- [4] Grigoris Antoniou, Frank Von Hormelen. “A Semantic Web primer”, The MIT Press ambridge, Massachusetts London, England.
- [5] Natalya F. Noy, Deborah L. McGuinness. “Ontology Development 101: A Guide to creating Your First Ontology”. Stanford University, Stanford, CA, 94305.
- [6] Riichiro Mizoguchi, “Tutorial on Ontological Engineering: Part 2: Ontology Development, Tools and Languages”. New Generation Comput., 22(1), 2003.
- [7] Matthew Horridge, Holger Knublauch, Alan Rector, Robert Stevens, Chris Wroe. “A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools Edition 1.0”. The University Of Manchester, August 27, 2004.
- [8] Thomas R Gruber, “Toward principles for the design of Ontologies used for knowledge sharing.”, revised, Aug23,1993 , presented at International workshop on Formal Ontology, March 1993, Padova, Italy.
- [9] Neeraj Seth , B.Vinoth Raj, Sanjay Kumar Malik, Nupur Prakash, SAM Rizvi. “Ontology Revisited: Design and Development”.
- [10] John Davies, Rudi Studer, Paul Warren, “Semantic Web Technologies”, trends and research in ontology-based systems, Wiley, pp 22-23.

- [11] Dieter Fensel, Ian Horrocks, Frank van Harmelen, Deborah, Peter, “OIL:Ontology Infrastructure to enable the semantic web”.
- [12] Xiaomeng Su, “Semantic enrichment for ontology mapping”, pp 29.
- [13] S. Staab, J. Angeles, S. Decker, M. Erdmann, A. Maedche, R. Studer, and Y. Sure, “Semantic Community web portals”. In proceedings of the 9th World Wide Web Conference, Amsterdam, Netherlends, 2000.
- [14] Berners Lee, Godel, and Turing, “Thinking on the Web”, Wiley, pp 127-128.
- [15] T. Berners-Lee, J. Handler, and O. Lassila, “The Semantic Web”, Scientific American, 279, 2001.