Description Logic based Quantifier Restriction and Query of an OWL Ontology

Rajiv Pandey
Asst. Professor
Amity University
Uttar Pradesh,
Lucknow-226010

Sanjay Dwivedi
Associate Professor
Dr. BBA University
Lucknow
226025

S. K. Singh
Professor
AmityUniversity
Uttar Pradesh,
Lucknow-226010

ABSTRACT
It is important to represent the knowledge residing on WWW in a uniform manner understandable by both man and machine to make it semantic. Thus some taxonomy is needed to make representations of the web contents which can be machine readable and usable .This taxonomy can be thought of as ontology. This paper proposes to relate Description logic (DL) and Web ontology language (OWL) based Ontology and demonstrate how a OWL ontology quantifier description can enable DL query to generate inferences there by demonstrating the machine interpretability aspect of the ontology and thus render support to the Semantic web

Keywords
Description Logic (DL), Ontology, OWL (web ontology language).

1. INTRODUCTION
Ontology as we have seen primarily consists of the hierarchical structure (terminology) of a domain and its relationships. The description logic languages are the heart of Knowledge Representation systems. The basic elements of knowledge representation are Concepts which symbolizes the set of individuals and their role one being the unary and the other being the binary predicate respectively. The DL (Description Logic) languages enable to formally describe the terminology with different expressive capabilities to address the needs of various domains like NLP (Natural Language Processing and databases). DL assigns reasoning complexity and decidability to the structure consisting of Concepts (the unary predicates), and role (the binary predicates).The DL can thus enhance the Reasoning aspect of the ontology for the agents that use it. The aspect of reasoning is beyond the scope of this thesis therefore only an introduction to basic DL concept is undertaken in the subsequent paragraphs.

"Description Logics(DL)[1][2] are a family of class-based knowledge representation formalisms, equipped with well defined model-theoritic semantics

Therefore Ontology can be considered as Terminology whereby the hierarchical structure is formulated to show relationship between those concepts, whereas in DL the knowledge representation is achieved using the Set of constructors and, in DL we move from disjoint atomic concepts to complex concept and Role description which enables more of reasoning related knowledge description.

This paper is organized in the following seven sections: 1) introduction, 2) Introduction to Logic 3) Description logic : Is it for the Semantic web 4) DL Syntax rules and inductive statements 5) Protégé Ontology Building tool and validator 6) Querying the OWL ontology On DL perspective 7)Conclusion

2. INTRODUCTION TO LOGIC
This section shall discuss Description logic and its significance to Semantic web but it is important that logic as a whole be introduced, this introduction shall also serve as the base to understand the OWL constructors and how they are used. It is also important to introduce how sentences can be written in logic or formal language.

Logic can be considered as:

i) As a description to support reasoners.

ii) Look at a language and its sentences

iii) May be used to formally represent language.

The sentence that is intended to be represented consists of a subject and a predicate.

Predicate is the part of the sentence that comments on the subject. The symbol used for predicate representation is in Capital letter.

Example: in the sentence

"Rajiv teaches mobile-computing".

Teaches is the predicate which talks about the subject Rajiv and mobile-computing and may be represented by capital "T".

Subject is that part of the sentence that a predicate says some thing about,in the above sentence "Rajiv" and "mobile-computing" are both subjects and may be represented by lower case "r " and "m" respectively

Therefore the logic representation of the sentence

"Rajiv teaches mobile-computing"

Becomes "Trm"

Predicate in uppercase is prefixed to the subject, in this case there are multiple individuals.

Logic can also enable us to represent variable and constants:

Example: “y is a professor”

“y” can take any value from the domain specified, thus is a variable.

and in the statement

“Rajiv is a professor”

“Rajiv” is a fixed value thus a constant.

Quantifiers and its symbolic representation

Quantifies are used to attach quantity to the sentence, the basic quantifiers in logic are

Some(at least one)

All

None

In the sentence

“y is a professor”

Logically represented as Py

If we add quantifier it can be said as

“for any y ,y is a professor”

The same is represented as : (y)Py

If we add universal ($\forall$)(all)or existential quantifier($\exists$)(at least one)

Then the sentence

“for all y ,y is a professor” will be prefixed with the Universal quantifier and represented as ($\forall$ y)Py.

And with prefixing existential quantifier, the sentence shall read as

“for some y, y is a professor”

represented as ($\exists$ y)Py. And infer as, only some are professors, rest may be lectures or assistant professors or anything but not professor.

The final basic quantifier None(negation) represented by the symbols (~)and ($\neg$)

when prefixed to the sentence

“for some y, y is a professor”

When negated shall read as

“not for some y, y is a professor”

Represented in logics as (~ $\exists$ y)Py

Meaning nobody is professor.

It is thus understood that when quantity is ‘some’ existential quantifier ($\exists$) is used, and when ‘every’ is the quantifier then universal quantifier ($\forall$) is used.

Logic operators or connectives: Logic connectives are used to build our sentences. the connectives may be used to either conjunct variables like x,y or conjunct sentences denoted by p and q

The basic connectives are

AND

OR

AND is used to join with the help of symbols (.) and ($\wedge$). This symbol is used if we have two sentences like

“Rajiv is Professor”(Pr) and “Rajiv is coordinator”(Cr)

By building on we can say

Rajiv is professor AND Rajiv is coordinator.

Pr $\wedge$ Cr

Logical AND signifies the conjunction of both things that we are joining.

The same can apply to variables also.

($\exists$ y)Py $\wedge$ Cy

To be inferred as some y are Professor AND Coordinator.

Or some professors are coordinators also.

OR: OR connective is symbolized by ($\vee$)

Everybody is either Staff OR Academician or both Staff and Academician.

($\forall$ y)Sy $\vee$ Ay

‘Read as every y ,y is Staff OR y is Academician’

Inferring that y is either Staff or Academician or both.

If we do not wish to include both then we can use exclusive-OR option symbolized by $\oplus$

And represented as

($\forall$ y)Sy $\oplus$ Ay

Now inferring that y is either Staff or Academician but not both.

Conditional constructs to build basic sentences and logic:

If ---- then symbolized as($\rightarrow$) or ($\supset$)

Example :

‘Academician is a teacher’

Can also be said

‘for all y , if y is academician then y is a teacher’

And represented as

($\forall$ y)Ay $\supset$ Ty

Bi-Conditional constructs

If and only If : symbolized by ($\leftrightarrow$) or($\equiv$)

Example:

‘One is Academician if and only if he teaches’

$(\forall y)Ay \equiv Ty$

Therefore ($\therefore$)symbol can be used to draw conclusions

Like

'Every programme consists of students'

MCA is a programme therefore MCA consists of students.

The logic and grammar has been briefly explained and the same is related to description logic in the following section.

3. DESCRIPTION LOGIC : IS IT FOR THE SEMANTIC WEB

This section introduces fundamental description logic relates it to sematicweb and list the constructors that it uses to build complex concept structure from the basic Atomic structures. It is also important to introduce DL because the flavor of Description Logic has further been extended in the OWL framework also the relationship will be discussed in the section when the various Owl categories shall be discussed.

It is however important to begin with the definition and fundamentals of DL and highlight the features of basic services the DL permits from the semantic web perspective.

"The name description logics is motivated by the fact that,

The important notions of the domain are described by concept descriptions,

i.e., expressions that are built from atomic concepts (unary predicates) and atomic roles

(binary predicates) using the concept and role constructors provided by the particular DL"[3]

"Description Logics (DLs) is the most recent name for a family of knowledge representation (KR) formalisms that represent the knowledge of an application domain (the "world") by first defining the relevant concepts of the domain (its terminology), and then using these concepts to specifyproperties of objects and individuals occurring in the domain (the world description)"[4].

"A description logic is mainly characterized by a set of constructors that allow to build Complex concepts and roles from atomic ones" [5] DL is a family of Formal conceptualization languages which are used in knowledge representation systems. DL is used for reasoning within the Domain of interest.

DL terminology and its synonym with OWL

- Concepts in DL are Classes in OWL which can be considered as set of objects
- Role in DL is Property in OWL
- Individual of DL is object in OWL

DL provides reasoning support to the semantic web. The software agents on the Web can perform reasoning and subsequent inferences based on the Axioms described in the DL. Axioms are those logical statements so called truth described by the terminology developer, which relates the roles of the concepts.

The DL axioms [7] are statements that describe:

i) relations between class (property)descriptions
ii) characteristics of properties such as asserting a property is transitive
iii) instance –of relations between (pairs of) individuals and classes(properties).

Therefore DL logic can ensure satisfiablity of both class hierarchy/relation and also whether the constraint specified by the relating axiom is satisfied.

The DL consists of Classes that are unary predicates that may be considered as set of individuals and their role which are binary predicates as they exist in pairs.

The DL formulates complex Class and their role definition by generating from the atomic concepts

The DL in brief is discussed which explains how ontology can be represented using the basic constructors provided in the DL in the subsequent paragraphs:

The Atomic Concepts are represented by upper case alphabets as A ,B etc. These atomic concepts can now help us to build basic DL representations using the constructors Atomic concepts A and B

A intersection B is represented in DL as

'A Π B'

In the above representation we have seen that both concepts A and B do not use variables unlike in a first order logic representation the same may be represented as

'A (y) $\wedge$ B(y)' where y is the variable that talks about all individuals that relate to the concept A and B respectively and also should be true for domain of interest .

A sample extension of these insertions of concept has been elaborated in the section of ABox description in the subsequent paragraphs.

Description logic is fundamentally used for Knowledge representation. The knowledge representation system should be able to generate answers to the queries based on the reasoning logic. Thus the DL should have express- ability feature so that it is capable to define the Axioms for appropriate and authentic reasoning support but with the increase in Expressiveness the complexity of the DL tends to increase.

The KR(knowledge Representation) system diagram shows that for a reasoning to be carried out the KR system should suitably be able to perform the knowledge formalization.A KB system fundamentally consists of two components

i)TBox

The TBox caters to the terminology definition, in which the vocabulary with respect to the concepts and their roles is incorporated.

"The basic form of declaration in a TBox is a concept definition,i.e. defining new concepts based on the previously defined concepts"[6]

"woman defined [6] as

"*Woman* $\equiv$ *Person* Π Female"

The above may be interpreted as person is a Woman 'if and only if' she is both Person and Female.

Thus a concept Woman is built by other defined concepts namely Person and Female.

ii)ABox

In the ABox the assertions with respect to the domain knowledge is specified

"The ABox contains extensional knowledge about the domain of interestwhich is assertions about the individuals,called as membership assertions"[6]

The ABox performs the basic checking if a given individual is the instance of the specified Concept called as 'instance checking' and also if every concept in the knowledge base has at least one individual known as 'realization'

Thirdly the ABox fetches individuals from the knowledge base as per the relation to the given concept called as 'retreival'

Thus Abox performs the under mentioned reasoning tasks

i) Instance –checking
ii) Realization
iii) Retrieval

A brief example [6] about the ABox assertions is as follows

'Professor Π HOD (ANIL)'

Stating that ANIL is professor and HOD(head of the Department)

As ANIL is only an instance of two intersected concepts thus this type of assertion is called as concept assertions

And the assertion below

'headsDepartment(ANIL,COMPUTERSCIENCE)'

Asserts the role of ANIL thus, these assertions are called as role assertions.

The next section describes the various syntax rules and how induction on statements can be applied to build upon concept definition and make inferences.

The knowledge representation and the Reasoning support to a KR system is shown in the Figure 1 below:

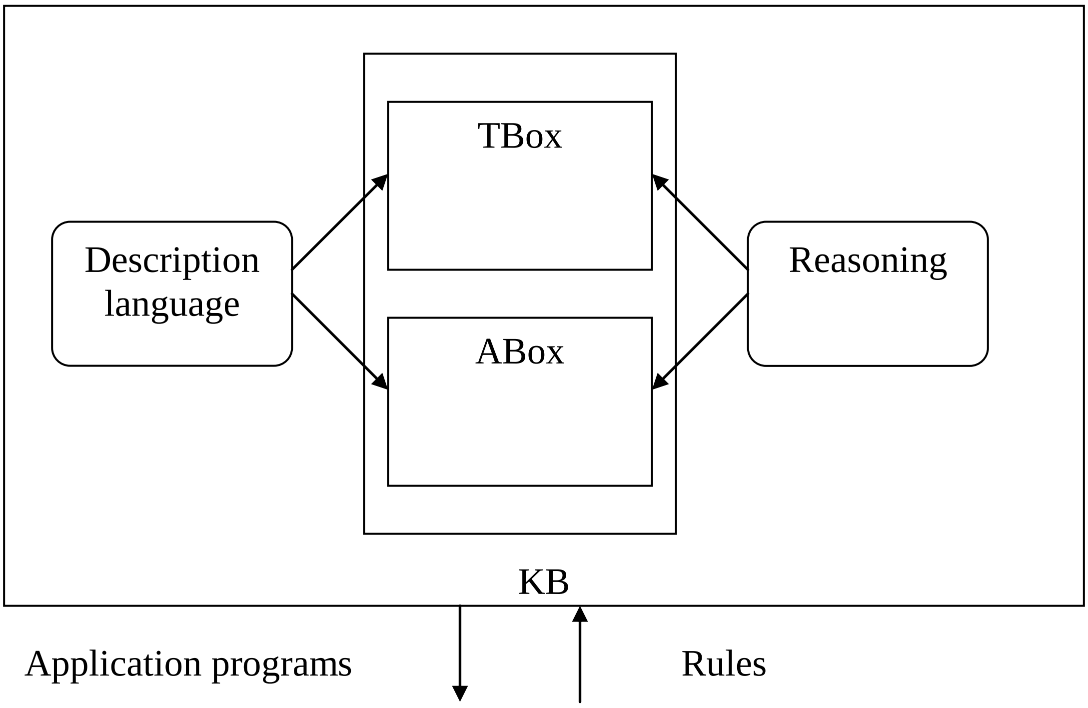

Figure 1.Architecture of a knowledge representation system based on Description Logic [4].

3. DL SYNTAX RULES AND INDUCTIVE STATEMENTS

Description language is formalizes complex concept build up from the basic Atomic concepts and atomic roles. This build up is inductively performed by the help of the constructors provided By the DL. The syntax below is listed using the basic description language AL(attributive Language) because AL is minimal language and most of the DL are a evolved version of the same

The atomic concepts are represented by A and B, R represents the atomic role and the concept description for the under mentioned syntax rule [4] is by C and D:

C,D	$\rightarrow$	A |	(atomic concept)
		T |	(Top/universal Concept)
		$\perp$ |	(Bottom concept)
		$\neg$A |	(atomic negation)
		C Π D |	(intersection)
		$\forall$ R.C |	(value restriction)
		$\exists$ R. T	(limited existential quantification)

Some important aspects before we can proceed to complex concept building with respect to the above syntax rule are:

- Negation is permitted with atomic concepts only. ex. $\neg A$
- Existential quantifier can only be associated with role description of Top concept. ex. $\exists R.\top$

In order to understand some of the above $\mathcal{AL}$ fundamentals we can consider two concepts 'Professor' and 'HOD' picked up from our earlier examples and the Atomic role *headsDepartment*.

Therefore the under mentioned representation shall intuitively describe that Professor is HOD

$$\text{Professor} \sqcap \text{HOD}$$

And the negation added to the above shall describe that Professor is not HOD

$$\text{Professor} \sqcap \neg \text{HOD}$$

The above concepts can further be build up by adding atomic roles as

Professor $\sqcap \exists$ *headsDepartment* $.\top$

The above statement also exhibits the 'limited existential quantification' rule where existential quantifier($\exists$) is used with the role *headsDepartment* and also with the Top concept ($\top$) the statement further describes that Some professor, *headsDepartment* or may be intuitively said that some professor *headsDepartment* statement is unconditionally true as the Top concept ($\top$) is used. Vice versa we can also describe those professors who are not heading the department can be represented by the ($\bot$) bottom concept. This statement exhibits the 'value restriction rule' ,using the universal quantifier($\forall$) with role(R) and bottom concept ($\bot$)

$$\text{Professor} \sqcap \forall \text{ headsDepartment} .\bot$$

We shall now briefly list the constructors that will help us to build roles and complex concepts from the basic role names and concept names.

"Description logic is defined in terms of interpretations. An interpretation ($\mathcal{I}$) consists of a domain ($\Delta^{\mathcal{I}}$) and an interpretation function ($.^{\mathcal{I}}$)"[7] .

The domain is the non empty set of objects.

The interpretation function maps each individual concept A (a set) $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$

(every element of *A* is also an element of interpretation domain) to every atomic role R(a binary relation) $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

Table 1. Constructors with the syntax and the semantics [7]

Constructor	Syntax	Semantics
Top	$\top^{\mathcal{I}}$	$\Delta^{\mathcal{I}}$
Bottom	$\bot^{\mathcal{I}}$	$\emptyset$
Concept name	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
Negation	$(\neg A)^{\mathcal{I}}$	$\Delta^{\mathcal{I}} \setminus A^{\mathcal{I}}$
Conjunction	$(C \sqcap D)^{\mathcal{I}}$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
Value restriction	$(R.C)^{\mathcal{I}}$	$\{a \in \Delta^{\mathcal{I}} \mid \forall b.(a,b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$.
Existential restriction	$(\exists R.\top)^{\mathcal{I}}$	$\{a \in \Delta^{\mathcal{I}} \mid \exists b.(a,b) \in R^{\mathcal{I}}\}$.

Two concepts are considered to be equivalent if and only if they are equal for all interpretations.

$$\text{i.e. } A \equiv B \text{ , if } A^{I} = B^{I}$$

the above can be inferred by equivalence of the following two syntax (i) and (ii)

i) $\forall$ headsDepartment .HOD $\sqcap$ $\forall$ headsDepartment .Professor

Then

ii) $\forall$ headsDepartment .(HOD $\sqcap$ Professor) are equivalent.

The above description was a brief on description logic as the objective was to highlight the role of DL in semantic web and ontology framework. The DL is also incorporated in OWL-DL a species of OWL.

5. PROTÉGÉ ONTOLOGY BUILDING TOOL AND VALIDATOR

This paper uses Protege_4.1_beta [8] for the purpose of Ontology development .Protégé is a free, open-source platform [8] that provides a growing user community with a suite of tools to construct domain models and knowledge-based applications with ontologies. Protégé incorporates a wide range of features including the GUI and extending to the customization and scalability using plug-ins. Protégé implements a rich set of knowledge-modeling structures and actions that support the creation, visualization, and manipulation of ontologies in various representation formats. Protégé supports customization to the extend of knowledge creation in a specific domain and associating external ontologies for a seamless integration to form a complex ontology Further, Protégé incorporates java API(application Programming Interface) to support other tools for knowledge based applications.

Features [9] of protégé may be as:

An extensible and customizable toolset for constructing ontologies and for developing applications that use these ontologies

• Outstanding features

– Automatic generation of graphical-user interfaces, based on user-defined models, for acquiring domain instances.

– Extensible knowledge model and architecture.

– Possible embedding of standalone applications in Protégé knowledge engineering environment and vice versa.

– Scalability to very large knowledge bases.

6. QUERYING THE OWL ONTOLOGY ON DL PERSPECTIVE

The above sections of this paper discussed the various aspects of description logic and how the same can be useful in asserting quantifier restrictions on the instances or properties of any ontology. This section also draws inference based on the description logic based quantifier restriction imposed on the UnivPeopleProgram Ontology.

The under mentioned portion of the code is a part of the UnivPeopleProgram ontology developed by the author and is represented below, as the whole ontology description is beyond the scope of this paper.

The Ontology is developed using Protégé OWL editor tool.The same queried and the inference drawn is displayed using the screen shot of the Protégé editor in Figure 2.

The ontology when subjected to the query

‘hasStrength min 1’

Returns classes of

Biotechnology, engineering and management as we have placed the cardinality restriction on programs and the instances of all the programs are retrieved by the above said query.

Snippet to assert minCardinality which is the basis for the results of the above query.

```
<SubClassOf>

    <Class IRI="#programs"/>

    <ObjectMinCardinality cardinality="1">

      <ObjectProperty IRI="#hasStrength"/>

      <Class IRI="#students"/>

    </ObjectMinCardinality>

  </SubClassOf>
```

The above code has asserted a cardinality restriction to the class of programs and students and its object property of hasStrength. Thereby asserting a rule on the ontology that every program hasStrength of minimum one student. This cardinality restriction can be customized for various universities as the need may be.

The query aspect is restricted to above queries only as it is treated only at an introduction level and the DL query applications can be it self be a topic of discussion at large which is beyond the scope of this thesis.

Executing query however signifies that the OWL[10][11] ontology can be viewed as a dataset which stores data values and returns the result when subjected to DL query.

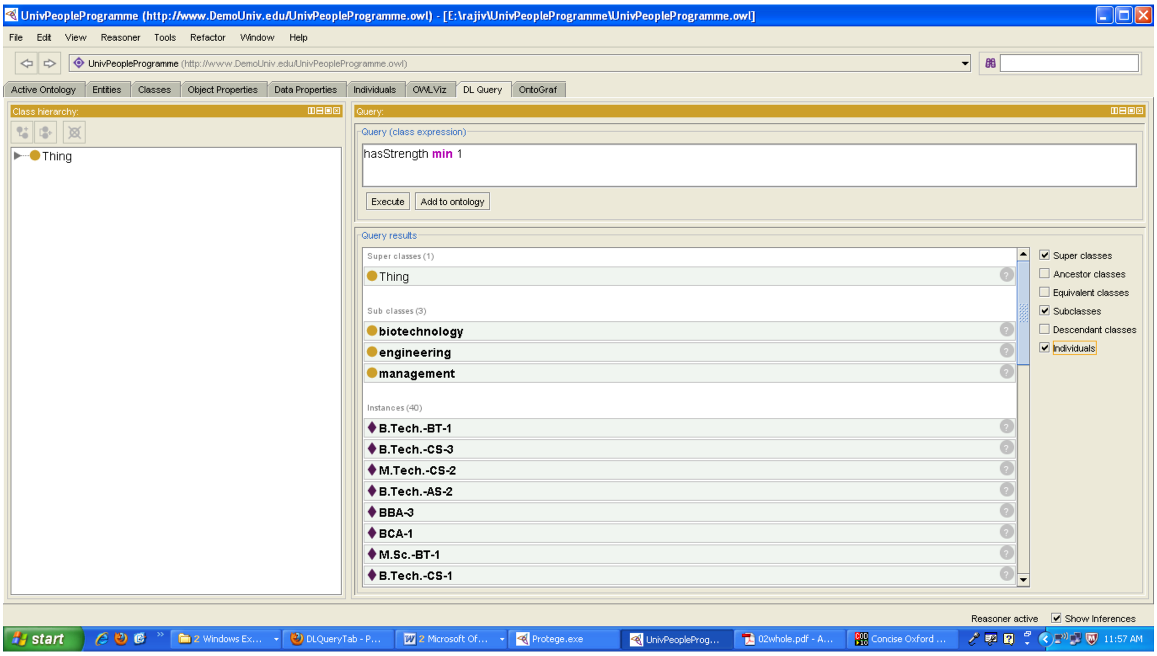

Figure 2. Screen shot displaying the DL query and the inferred results.

7. CONCLUSION

Despite the need being felt to make today's web more machine interpretable and knowledge extractable there is a lot desired to make the semantic web a reality. This paper prompts the importance of the Ontology and incorporating restriction on the ontologies developed. Ontological definition of the Web contents which will not only keep the contents human usable but also add the feature of machine interpretability and after a meaningful interpretation the same is also demonstrated in the section above by exposing the ontology to DL query. It is also observed that OWL will play a dominant role in the future for defining ontologies for supporting context reasoning and knowledge sharing. OWL Ontology using protégé has returned results to the query and also had enabled DL restriction as follows:

a)A Restriction Framework axiom based on the following to enhance inferencing as the need may be.

i) Quantifiers
ii) Cardinality

b) Incorporating large scale DL based query where Data sets may be the concerns and the Ontology aspect is not incorporated i.e. Developing OWL based databases and injecting queries on them

REFERENCES

[1] F.Baader,D.L.Mcguiness, D.Nardi and P.Patel-schneider,editors.The description Logic Handbook : theory implentation, and Applications.CambridgeUniversity press,2002

[2] F.Baader and W.Nutt.Basic description logics.In franz baader,Diego Calvanese,Deborah mcguinness,Daniele nardi,and petert F.patel- Schneider,editors,The description logic handbook: theory implentation, and Applications,pages 43-95. Cambridge University press,2003

[3] Description Logics Franz Baader, Ian Horrocks and Ulrike Sattlerc 2007 Elsevier.

[4] Basic description Logics,Franz Baader and Werner Nutt

[5] DESCRIPTION LOGICS-basics, Applications, and more, Ian Horrocks University of Manchester,UK

[6] introduction to description logics,Daniele Nardi,Ronald J. Brachman.

[7] Description logics: Reasoning support for the Semantic web,Jeff Z.Pan,2004.

[8] http://protege.stanford.edu/

[9] Creating Semantic Web (OWL) Ontologies with Protégé ,Holger Knublauch, Mark A. Musen, Natasha F. Noy, ISWC2003

[10] http://www.w3.org/TR/2009/PR-owl2-primer-20090922/OWL 2 Web Ontology Language Primer W3C Proposed Recommendation 22 September 2009

[11] http://www.w3.org/TR/owl-features/, OWL Web Ontology Language Overview W3C Recommendation 10 February 2004