"Vasile Alecsandri" University of Bacău
Faculty of Sciences
Scientific Studies and Research
Series Mathematics and Informatics
Vol. 23 (2013), No. 1,

ABOUT DESCRIPTION LOGICS AND ONTOLOGY

SIMONA-ELENA VÂRLAN AND COSMIN-ION TOMOZEI

Abstract. Description Logic (DL) it is considered a tool that can make the Semantic Web vision a reality. The Ontology Web Language (OWL), the language to create ontologies, is the product of the combination between Resource Description Framework (RDF) language and DL. In this paper, we analyze the utility that DL brings in implementing ontologies which are the main way to create semantics in Web space.

Key words: Description Logics, Ontology, Semantic Web.

I. INTRODUCTION

Description Logics are often named as a tool that can support the Semantic Web. What Semantic Web (SW) represents and what academia and industry is trying to achieve become in the past decade one of the main subject for many conferences, research paper, projects and software application development.
Ontologies plays an important role for Semantic Web, because it represents real knowledge bases on which we can make inferences and most important, create de semantic context that present web documents needs, in order to be intelligible for computers.

The aim of this paper is to analyze the role that Description Logics plays in realizing the Semantic Web vision. We will make a brief presentation of the Ontology and Semantic Web vision, we will present an introduction to Description Logics and how it interacts with Ontology and we will end with a concrete example of applying Dl inferences on a proposed Ontology about IT&C components.

II. ONTOLOGY AND SEMANTIC WEB VISION

The vision of Tim Berners-Lee about the Web is simple but yet powerful:

The Semantic Web … information is given well-defined meaning … idea of having data on the Web defined and linked in a way that it can be used for more effective discovery, automation, integration and reuse across various applications - Tim, Berners-Lee, et. al. 2001-

The Web space has become indispensable for most people. Searching what we need and want through the Web space it is a simple, daily and common activity that everyone is doing. But, the quest for finding the right, in real time answer (because the information is "power") it is some times frustrating because today's search engines offer only keyword search and many of the answers are irrelevant from the semantic query point of view. This is happening because web resources (the web pages) are only human readable (understandable), the HTML tags with what we create web pages only provides information presentation intended only for human readability.

The Semantic Web vision comes before this lack of machine-understandable mark-ups that are missing from the web resources. It's aim is to create a semantic context that can be machine readable not only human readable in order for information to be shared, processed both by automated tools like search engines, or any other software application, and by human users.

The use of Ontologies permits to create the required well-defined knowledge bases that support reasoning tools. To create an ontology we use ontology languages, such as Web standards like RDF, RDFS, OWL.
If we want to answer at the question what is Ontology we will find many definitions in the literature.

The most known ones are the ones of:

1. Neches [8]: "defines the basic terms and relations including the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary";
2. Gruber [5] (which is the most quoted): "an explicit specification of a conceptualization";
3. From philosophy [12] it is defined to be a systematic explanation of being.

We conclude that Ontology defines concepts and relations between them from a specific knowledge domain.
The manners of expression of ontology specification languages are different, of which the most used are OIL (Ontology Inference Layer) [9], DAML-OIL(DAML - DARPA Agent Markup Language) [10] and OWL(Web Ontology Language) [11].
Web Ontology Language is a family of languages for knowledge representation. It was created to provide a way to process and understand the

content of a web page (not just to display it). It was built to help computers "understand" the data.

III. INTRODUCTION TO DESCRIPTION LOGIC (DL)

DLs (Description Logics) represent a family of KR languages used to represent knowledge in a structured and formally well-understood way.
Basic constructions of a DL are concepts (or classes), roles (or properties, relationships) and individuals (or instances of that class) [4]. A concept is a set of objects from a specific domain. For example, the concepts Person and Employee are the set of people and respective the set of employees.

A role is a binary relation between two objects in the domain. For example, the role *hasSubordonate* represents the relationship between parents and their children. Concepts and roles correspond to unary predicates (arity one), of first order logic, and respective binary predicates (arity two).

The concepts and complex roles of higher arity (we consider the predicate P (x, z, y) of the arity 3) may be developed from known predicates using the operators: intersection⊓, negation¬, and universal ∀ and existential ∃ quantifiers. For example, if we want to define the concept Employee as a Stipendiary Person and a Manager a Person how has at least one subordinated Employee we have the following axioms:

$$\text{Employee} \equiv \text{Person} \sqcap \text{Stipendiary}$$

$$\text{Manager} \equiv \text{Person} \sqcap \exists(\geq 1\ \text{hasSubordinate}).\text{Employee}$$

Where the concept employee is defined as the intersection between Person and Stipendiary, and Manager as a person who has at least one subordinate.

DL includes semantic logical formalisms, therefore knowledge represented in these languages expressed as sets of DL formulas cannot be interpreted differently by different people. DL is used both for knowledge representation and for reasoning based on them.
This means that we can infer (create) new knowledge, based on existing ones, using inference rules provided by DL. One of these rules is subsuming. This is based on concepts classification that determines subconcepts (subclasses) or superconcepte (superclasses). The rule applied to the above example provides the following deductions: Employee is derived from Person and Manager from Employee. This rule is used to make inferences and find out if an object is an instance of a concept.

Description Logic is applied to knowledge bases. A knowledge base – KB, consists of TBox and ABox.

TBox include terminological knowledge, knowledge about concepts and roles. Axioms in the example above fall into this category.
ABox contains assertions about objects and roles between them: example hasSubordinate(A,B).
Objects can be considered individuals (constants), the concepts sets of individuals, and roles, binary relations between individuals. The purpose of reasoning over terminology (TBox) is to determine whether a description is *sufficient* or *satisfy* [1] a KB or, in other words, if a description is more general than another is, or if the first one deduce the other.
ABox basic problem is to check whether the set of assertions is *consistent*, that means any assertion must be an instance of a concept.
Therefore a significant KB = consistency + sufficiency.
We use the notation (adapted from Franz Baader):
Let KB be a knowledge base with KB = T+A, where T - set of terminology TBox (terminological axioms) and A - set of assertions or ABox (individual axioms);

C and D concepts description C, D $\in$T,

C$\subseteq$D – inclusion axiom,

C$\equiv$D – axiom of equivalence,

R – role,

a,b – assertions, a,b$\in$ A of form C(a) and D(b) with (a,b): R or R(a,b).

Let I be the interpretation consisting of non-empty set ΔI (interpretation domain) and an interpretation function that assigns to each atomic concept A a set of interpretations: AI cu AI $\subseteq$ ΔI and each role a binary relation of the form: RI $\subseteq$ ΔI x ΔI. We say that two concepts are equivalent if CI≡DI for $\forall$I.

Example:

$$\forall \text{hasSubordinate.Employee} \sqcap \forall \text{hasSubordinate.Manager} \equiv \forall \text{hasSubordinate}(\text{Employee} \sqcap \text{Manager}).$$

The KB is significant if it meets the sufficiency and consistency.

Checking sufficiency:

Terminology is of the form C$\subseteq$D or C≡D. We say that an interpretation I satisfy:

- inclusion C$\subseteq$D if CI$\subseteq$DI ,
- equivalence C≡D if CI≡DI.

Generally we say: T set of terminological axioms, then I satisfies T if I satisfies each element of T: $\forall$ t $\in$T , tI $\in$ ΔI.
Checking consistency:
ABox individuals have names and properties. Let a,b,c be individuals and the concept C described in TBox and role R. We can make assertions: C(a) and R(b,c) $\Rightarrow$ a assertion for C and c has role R for b.

Let A be a finite set of assertions. I satisfy assertions about C, C (a) if aI $\in$ CI and satisfy assertions about the role between a and b: R(a,b) if (aI, bI) $\in$RI. An interpretation satisfies A if it satisfies each assertion in A.
Ability to express taxonomies as hierarchical classifications and the ability to make reasons offered by DL, led to a number of proposals that use DL to represent and reason with information from the Web or, to model web structures and to acquire new information from the Web [4].

IV. DESCRIPTION LOGICS AND ONTOLOGY

To create Ontology we use OWL, which is based on DL, offering the necessary support to make inferences upon the knowledge base (the ontology). Ontologies have been created over the structure of RDF, which provides only the way to describe web resources by creating metadata. Ontologies provide further description of the structure of concepts and relationships between concepts, statements of classes and subclasses, types and cardinality for concepts properties. Therefore provides adequate support for the development of rule-based reasoning and inferences. Ontology is based on specific Description Logic (DL) concepts or, we can say, the OWL language is based on description logic.

Ontology is represented by TBox and ABox – which incorporates elements of terminology, vocabulary domain modeling and respective conceptual assertions or individual axioms.

Description of the instances and records in the ABox are made by inserting RDF taggs which offer the basic semantic level for the semantic descriptions. RDF-Schema adds the declaration of classes and subclasses, of predicates and data types, enriching elements necessary for creating vocabularies. OWL, language dedicated to the description of ontologies, based on RDF and RDF Schema, provides possibility to described concepts and relations, the terminology needed to define the TBox from the DL.
Operations for TBox and ABox are different: the ones forTBox are used mostly to make inferences that determine and verify hierarchy of classes and those for ABox are based on rules, which checks the instances and their attributes.
ABox context can be defined using RDF such: [7].

```
<rdf:Property rdf:about =http://xmlns.com/foaf/0.1/homepage rdfs:label=”homepage”>
    <rdfs:subPropertyOf rdf:resource=”http://xmlns.com/foaf/0.1/page”>
</rdf:Property>
<foaf:Person rdf:about=”#me” xmlns:foaf=”http://xmlns.com/foaf/0.1”>
    <foaf:name>Nume</foaf:name>
```

```
        <foaf:homepage rdf:resourse="http://danbri.org" />
    </foaf:Person>
```

According to the example above it is noted that is defined the property *homepage* (identified by URI) which is sub-property of the property *page* (identified also by URI) and instance *person* (identified by URI) with the attributes *name* and *homepage* (identified by URI)

The rule applied here is:

```
If      Property(homepage) , Property(page)
        subProperty(homepage, page)
        hasProperty(Person, homepage)
Then    hasProperty(Person, page)
```

Or

http://xmlsn.com/foaf/0.1/#me (person - subject) foaf:page (has property page - predicat) http:// danbri.org (URI - object)

TBox context is defined using RDFS and OWL as:

```
<rdfs:Class rdf:about="http://xmlns.com/foaf/0.1/Document" rdfs:label="Doc">
        <rdfs:subClassOf rdf:resourse="http://xmlns.com/wordnet/1.6/Document" />
</rdfs:Class>

<rdfs:Class rdf:about="http://xmlns.com/foaf/0.1/PersonelDocument">
        <rdfs:subClassOf rdf:resourse="http://xmlns.com/foaf/0.1/Document" />
</rdfs:Class>
```

According to the example it declares two classes (documents), the first one is subclass of documents wordnet, and the second class (personal documents) is subclass of the first declared class. We can deduce the following fact:
http://xmlns.com/foaf/0.1/PersonelDocument (the second class)
rdfs:subClassOf (is subclass of)
http://xmlns.com/wordnet/1.6/Document (class worldnet documents)

V. APPLYING INFERENCE REASONING OVER AN ONTOLOGY

In section III and IV we presented a short introduction bout Description Logic and how it offers the necessary support to create inferences on ontology concepts and roles. In this section, we will present a concrete example of applying inferences over an Ontology. In order to realize that we create an Ontology – the knowledge base, about IT&C components. We choused to describe the IT&C components in Romanian because the ontology is under development and it is used for tests in our research department.

The main **concepts** from the IT&C Ontology are: *Produse* (Products), *Producator* (Produser), *Categorie* (Categories), *Promotii* (Promotions) and *Lichidari_Stoc* (Stock Liquidations). In Figure 1 is presented the graphical representation of the ontology.

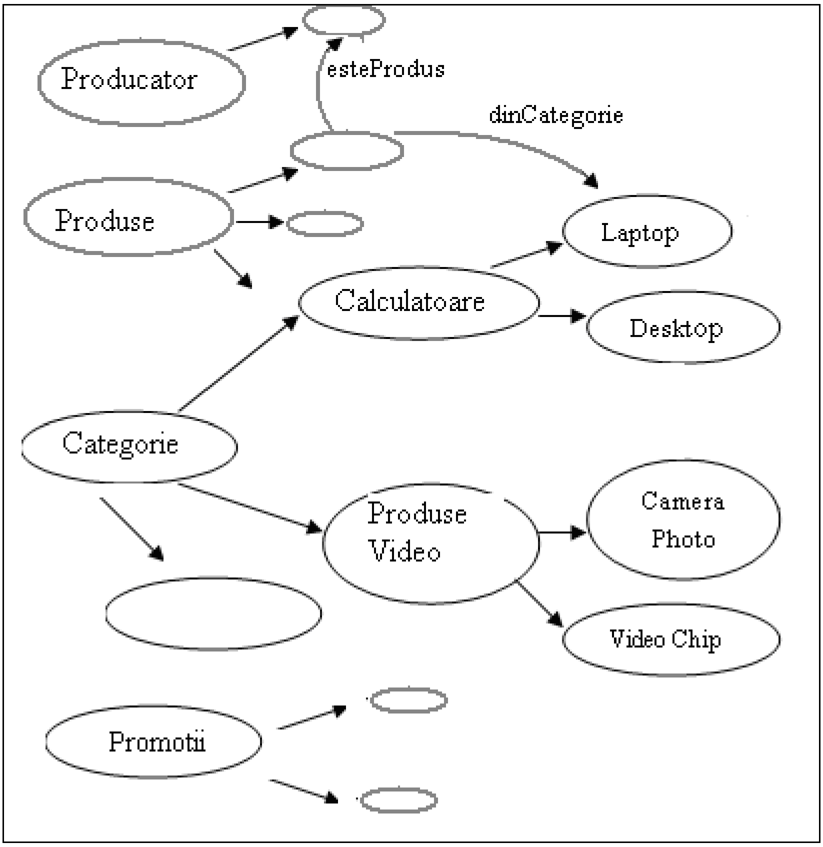

Figure 1: Graphical representation of Ontology

The Ontology was created using the Protégé version 4.3 [13] ontology editor and OWL language. It is a free; java open-source editor developed at Stanford University and is one of most used ontology editors. The make inferences we used the Hermit OWL reasoner [14] that it comes with Protégé 4.3.

Some properties that are subject of interest for this case study are:

Object properties:

- *produce(A, B)* which means that A produce the product B and *produce(A,B*) is inverse of *esteProdus(B,A)* property which means that product B is produced by producer A;

- *laPromotie(B)* which means that product B is at promotions.

Data type properties:

- *arePretValoare* meaning has price value;
- *areReview* for expressing (with marks from 1 to 10) the review that a specific product has received from other users.

In Figure 2 are captured the use of these properties. The property that specifies that the product Acer_Aspire_AOD_255 is at promotion is marked with the blue circle and the review that this product has received is marked with green circle. We also captured the inference that the inference engine made about the property *produce* (red circle). Because we declared previously that Acer Company produces Acer_Aspire_AOD_255 and because the property *produce* is inverse of *esteProdus*, the reasoning engine inferred also the following fact: *esteProdus*(Acer_Aspire_AOD_255, Acer_Company).

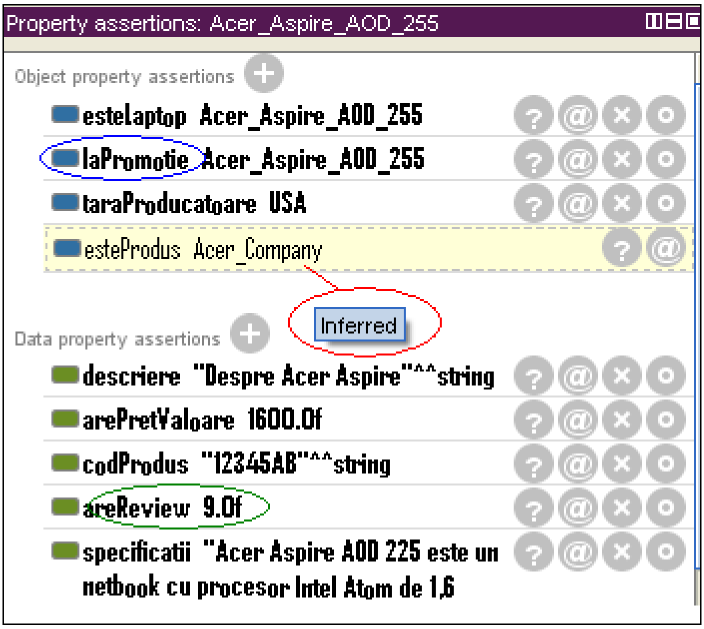

Figure 2: Properties for Individual Acer_Aspire_AOD_255

To test how inferences work over our IT&C ontology, we created some concrete Description Logic queries. A test that we conducted was to find out, for an interested buyer, what would be the recommended computers. The criteria that a computer has to accomplish are a very good review mark (over 9.5 points) or to be at stock liquidations or at a promotion. The DL query I showed in Figure 3 together with the expected results.

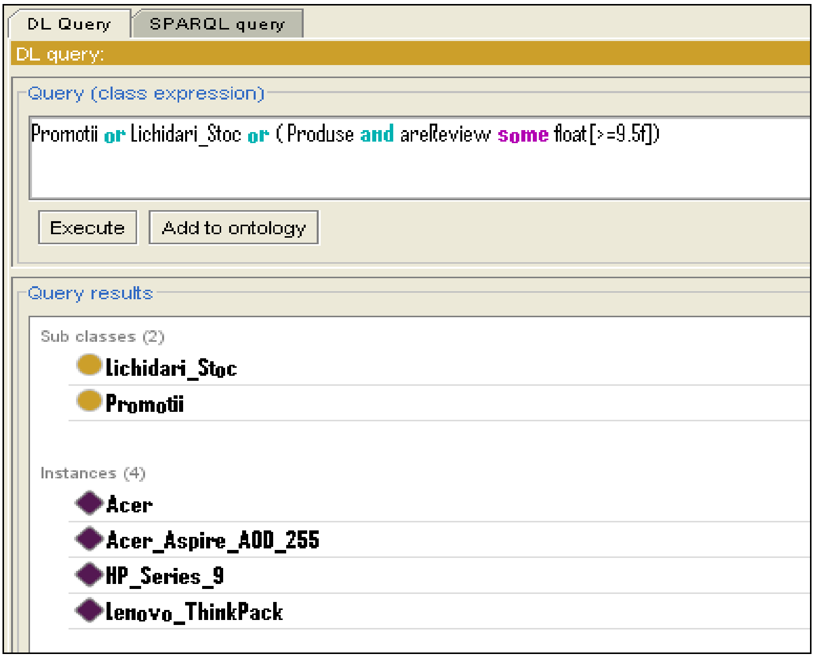

Figure 3: DL Inference

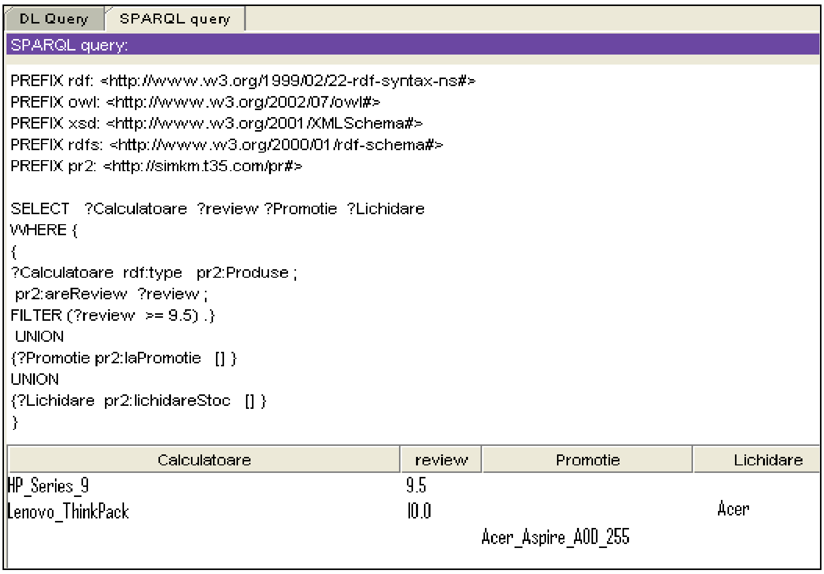

Figure 4: SPARQL query

The Protégé application offers also the possibility to create SPARQL [sparql??] query over the ontology. The SPARQL query is showed in Figure 4 and, as it can see, we listed the property values for the products. The corresponding name space for our ontology is *pr2*. [mutat de deasupra figurilor]

VI. CONCLUSIONS

Ontology represents a fruitful research area in the last two decades contributing also to continuous research and interest in knowledge representation area, respectively to Description Logics.

Ontology targets the Semantic Web vision through semantics and DL-based statements that allow inferences and semantic queries upon knowledge base.

Many Ontologies, more or less complicated, are already used to describe, in a computer readable way, the web content. Most of them are represented using OWL language, language created in accordance with DL statements.

Acknowledgment

This work was supported by CNCSIS-UEFISCSU, project number PN II-RU code 188/2010.

References

[1] Baader, F., Nutt, W., **Basic Description Logic** , 2002, http://www.inf.unibz.it/~franconi/dl/course/dlhb/dlhb-02.pdf

[2] Berners-Lee T., Hendler J., Lassila O. (2001) **The Semantic Web.** Scientific American, May 2001.

[3] Baader F., Calvanese D, McGuinness D.L., Nardi D., Patel-Schneider P.F., eds.(2003), **The Description Logic Handbook: Theory, Implementation and Applications,**. Cambridge University Press, 2003

[4] Eklöf, M., Hörling, P., Svan, P., Suzic, R., **Choong-ho Yi, Information & Knowledge Management for NBI, User report, Systems Technology**, Stockholm, 2004

[5] Gruber, T. R. (2001). **What is an Ontology?** Stanford: Knowledge Systems Laboratory, Stanford University. http://www-ksl.stanford.edu/kst/what-is-an-ontology.html

[6] Horrocks, I., **Ontologies and the Semantic Web**, Communications of the ACM, 51 12, 2008, pp. 58–67

[7] Myungjin, L., **Inference on the Semantic Web,** 2008, http://www.slideshare.net/onlyjiny/inference-on-the-semantic-web, p.15

[8] Neches, R., Fikes, R.E., Finin, T., Gruber, T.R, Senator, T., Swartout, W.R., **Enabling technology for knowledge sharing**, Al magazine, 12(3): 36-56, 1991

[9] www.w3.org/TR/daml+oil-walkthr

[10] www.daml.org
[11] www.w3.org/TR/owl-ref
[12] http://en.wikipedia.org/wiki/Ontology
[13] http://protege.stanford.edu/
[14] http://hermit-reasoner.com/

Simona Elena Vârlan
„Vasile Alecsandri" University of Bacău, Faculty of Sciences, Department of Mathematics, Informatics and Education Sciences, 157 Calea Mărăşeşti, Bacău, 600115, Romania, e-mail: varlan_simona@yahoo.com

Cosmin Ion Tomozei
„Vasile Alecsandri" University of Bacău, Faculty of Sciences, Department of Mathematics, Informatics and Education Sciences, 157 Calea Mărăşeşti, Bacău, 600115, Romania, e-mail: cosmin.tomozei@ub.ro