Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 6 Issue 04, April-2017

The Foundation of the Web Semantic Architecture and Development Concept

Ramahefy T.R. , Rakotomiraho S.[2] , Rabeherimanana L. I. [3]
Embedded Systems - Instrumentation - Modelling of the Systems and Electronic Devices (SE-I-MSDE)
ED STII - University of Antananarivo
BP on 1500, Ankatso - Antananarivo 101 – Madagascar

***Abstract*— The Semantic Web, more technically called “Linked Data” allows machines to understand the semantics or meaning of information on the Web. It extends the network of hyperlinks between conventional web pages by a network of structured data links, thus enabling automated agents to access more intelligently the different data sources contained on the Web.**

The aim of this article is to represent and organize the knowledge in the data and the information in the Web 2.0 so that the computer systems are able to reason itself. This organization and modeling of knowledge which is presented in the form of the semantic Web is none other than the “Internet of Things” and allows us to have the interoperability of systems .

Keywords*— Semantic web, ontology, RDF, description logic, SPARQL, Tim Berners-Lee

I. INTRODUCTION

The evolution of the technologies of the computer systems currently used by the Web 2.0, allows us to have an exponential increase in the volumes of information exchanged, network traffic and equipment performance.

But the activities of the current Web in general are not particularly adapted to software tools. And it would not have been as successful without search engines. Now with current search engines: we have low precision, resulting in very sensitive vocabularies, leading to the form of a web page which only humans can gather and exploit. Moreover, the interpretation of information by computers is currently very difficult because it is stored in an unstructured way.

The challenge of the semantic Web is therefore to provide a language that expresses both data and rules of reasoning on the data and that allows exporting from the Web, the rules of any system and the representation of knowledge.

This article examines all the essential aspects that the semantic Web is based on to set up an autonomous computer system capable of extracting, presenting to acquire and sharing knowledge between computers.

II. THE SEMANTIC WEB

A. W3C to the semantic web

Tim Berners-Lee's work on the Web began in the 1990s, 20 years after ARPANET [1]. The Web and Internet association is the result of a fruitful collaboration between an American and a European. For the Web in particular, three versions have been listed since its creation: the static Web, the dynamic Web or web participative and the semantic Web.

1) Static Web

The static Web or Web 1.0 constitutes the “sites first version”. Images, texts, videos and sounds, in short, the contents, are designed and hosted by a company administrator of the site. They were the first information systems of the Internet age. They were static and the content of the pages at the time was updated rarely. The Web 1.0 of the 90s was functionally very linear and very restrictive. It was a passive Web, since the Internet user consumed just information, as if they were reading a book. [2]

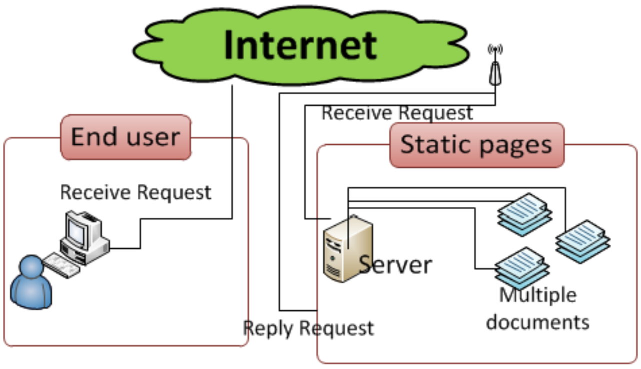


Fig. 1. Web 1.0 architecture

2) Web 2.0

The replacement of Web 1.0 was Web 2.0. Web 2.0 first appeared in August 2004, during a conference on the trend of web modification. With the invention of new technologies such as ASP or PHP language, associated with databases, a number of sites became more dynamic. Images, text, video or sound content could be manipulated by a Content Management System (CMS). Web 2.0 is therefore the acquisition by Internet users of new applications, belonging to the open source group, to spread digital data through wikis and blogs, share pictures, videos, movies, make online purchases and share collective intelligence.

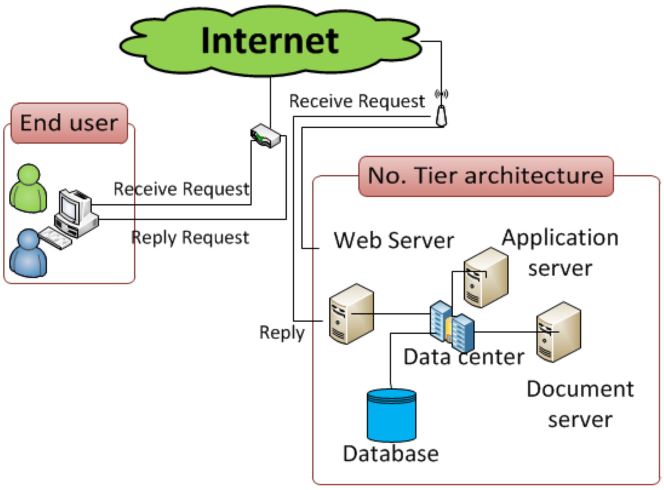


Fig. 2. Web 2.0 No.Tier architecture

B. W3C to the semantic Web

Semantic Web Stack is a term created in 2008 by Tim Berners-Lee. The Semantic Web Stack, which can be translated as a pyramid of the Semantic Web or Semantic Web Stacks, is a way of schematizing the flowchart of computer languages: each layer uses and exploits the abilities of the layers that lie below it. This demonstrates their planning at the heart of the Semantic Web, which is an extension, and not a replacement, of the "standard hypertext" technology used on the Web. The Semantic Web pyramid is constantly increasing as each layer and associated language is optimized [3]

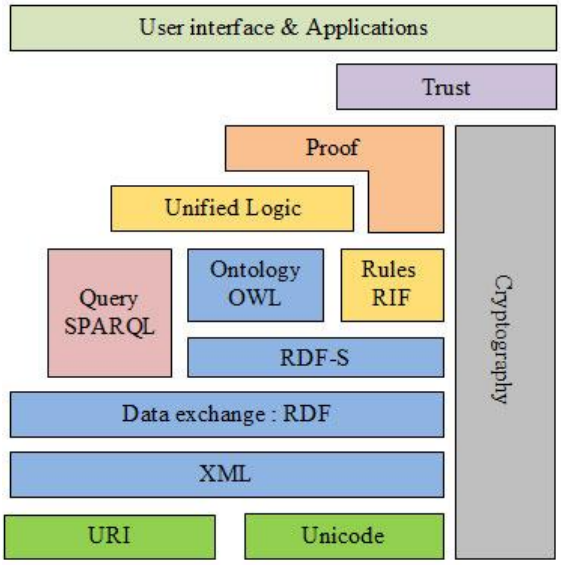


Fig. 3. Semantic Web Stack

1) URI and unicode

The Uniform Resource Identifier (URI) is a simple and extensible protocol to identify, in a unique and uniform way, any resource on the web. This element forms the basis of the layer architecture of the Semantic Web.

2) XML

XML is a Markup Language, a subset of SGML. In reality, it refers to a metalanguage that gives us the means to characterize our own tags for documents. XML uses Document Type Definition (DTD) for its data models.

3) RDF

The Resources Description Framework (RDF) is a graphical model developed to expose proposals. It is an assertion language for the access and use of Web resources. The RDF uses metadata to characterize Web resources. It makes use of the semantics of a formal language and its ultimate goal is to edit an expression of the propositions concerning a subject. By using non-formalized and non-hierarchical data on the Web, RDF is a key language for interoperability between applications.

4) SparQL

SPARQL refers to a query language and a protocol to access the RDF set up by the W3C RDF Data Access workgroup. As a query language, SPARQL is "data-oriented", thus it only questions the information acquired in templates.

There is no inference in the query language itself. The assignment of SPARQL is to take note of the application request characteristics which presents itself as a request, and re-transmits this information in the form of a links set or an RDF graph

5) Logic of description

Descriptive logics are a family of knowledge representation languages that can be used to display application domain knowledge in an orderly and formal manner. An essential feature of these languages is that they have a formal semantics.

The logic of description makes concepts notions use, roles and individuals. Concepts coinciding with classes of individuals, roles are defined as relationships between these individuals through Abox and Tbox.

6) Ontology

Ontology can be defined as a formal vocabulary meaning description used in an RDF document (or a triplet graph or a triplet warehouse). The Ontology word comes from philosophy. Ontology for the Semantic Web refers to the properties, classes of subjects and objects of triplets and entities that can be displayed. It is possible to consider Ontology as a contract that binds the data producer and the data consumer. For a good Ontology to be used, it must be neither too easy nor too complicated. Ontology is displayed in RDF, generally in the same graph of triplets.

7) Logic and proof

Logic and proof are very useful for making inferences as well as its explanations. Logic is a language that allows us to express the rules of reasoning. These rules make it easier for us to deduce new facts from existing facts. A proof is a series of rules applications which allows us to deduce a new fact.

8) Trust

If agents are designed to make decisions in the humans' place, their actions must be reliable, that is, there is *confidence* in the results. Thus, the agent in question must be able to:

- Clarify how reached its conclusions with proof
- Ensure the reliability and the source of the information used for the digital signature.

III. LANGUAGES USED

A. Assertion and annotation language

Assertions attest to the presence of relationships between objects. They are thus granted with the annotation expression that one wishes to combine with the web resources. To illustrate this, RDF is used as it displays superior privileges for computer processing. It will be used to annotate documents written in unorganized languages, or as an interface for documents written in languages with equivalent semantics, for example the databases.

B. Ontology Definition Language: OWL

OWL language, is intended for classes determination and properties types, and thus for Ontologies determination. Inspired by descriptions logic, it produces innumerable constructors which give the possibility of exposing in a very specific way the classes determined properties. The cost of this expressivity is the undecidability of the language received by attempting to account for all of these constructors. This is why OWL has been divided into three distinct languages:

- OWL LITE has only a limited subset of available constructors, but its use ensures that the comparison of types can be calculated.
- OWL DL contains all the manufacturers, but with specific constraints on their use that guarantee the decidability of the type comparison. The drawback however is the enormous difficulty of this language (one of its fragments is P-SPACE-complete), is that it requires a heuristic approach;
- OWL FULL, without any constraint, for which the concern of type's comparison is undecidable.

C. Description languages and composition of services

The purpose of this section is to describe different languages, architectures and standards for Web services. New languages dedicated to Web services are regularly offered by industrial and academic research organizations. It should not be forgotten that most of the languages presented are complementary and do not respond to the same needs. We will therefore present the objectives and functionalities of the main languages devoted to services on the web like UDDI, WSDL, DAML-S, XL, ebXML, RosettaNet

D. Query languages

1) SQL

SQL is a "declarative" language. We specify what we want to get or to do and it is the computer decides how it should be executed. SQL contains 5 main parts, which allow it to define the elements of a database (tables, columns, keys, indexes and constraints), manipulation of the data (insertion, deletion, update and extraction), the management of data access rights (acquisition and revocation of rights), transaction management and finally integrated SQL.

2) SPARQL

As we saw in the previous section, SPARQL can be used to express queries through various data sources, whether the data is stored natively as RDF or viewed as RDF via middleware. SPARQL is able to search for compulsory and optional graph patterns and their conjunctions and disjunctions. SPARQL also manages the extensible test of values and the query constraints by a source RDF graph. The results of SPARQL queries can be result sets or RDF graphs.

IV. SEMANTIC WEB CONCEPT

Semantic Web operates under the "open world hypothesis". Unlike the first systems of artificial intelligence such as expert systems, text analysis, pattern recognition. Semantic Web systems do not use a fixed vocabulary. Semantic Web encompasses a variety of concepts and terminologies that evolve regularly and are maintained by several speakers on the Web. The following figure shows the synoptic process diagram to develop a Semantic web site.

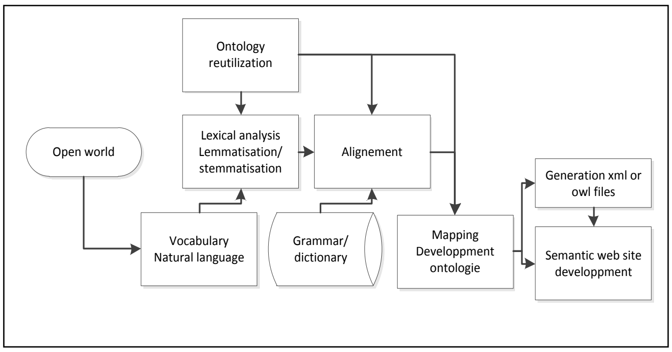

Ontology reutilization
Open world
Lexical analysis Lemmatisation/ stemmatisation
Alignement
Generation xml or owl files
Vocabulary Natural language
Grammar/ dictionary
Mapping Developpment ontologie
Semantic web site development

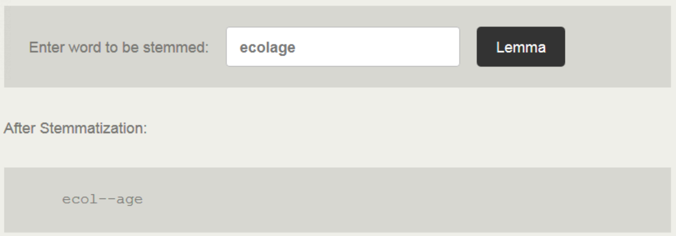
Fig. 4. Web semantic development synoptic schema

A. Lexical analysis and lemmatization

Regular expressions are a very powerful and fast way to search strings for example sentences. It is a method to search or replace text in strings which becomes a near necessity. Regular expressions will allow us to perform extensive searches and replacements in texts.

Lemmatization is the content lexical analysis of a text gathering the words of the same family. Each of the word in the content is reduced to an entity called "lemma", also known as canonical form. The lemmatization brings together the different forms that a word can have: verb, noun, adverb, etc.

Several algorithms allow the normalization of words. It allows removing the affixes of the words to obtain its canonical forms. We use the regular expression of lexical analysis to achieve our goal.



Enter word to be stemmed: ecolage

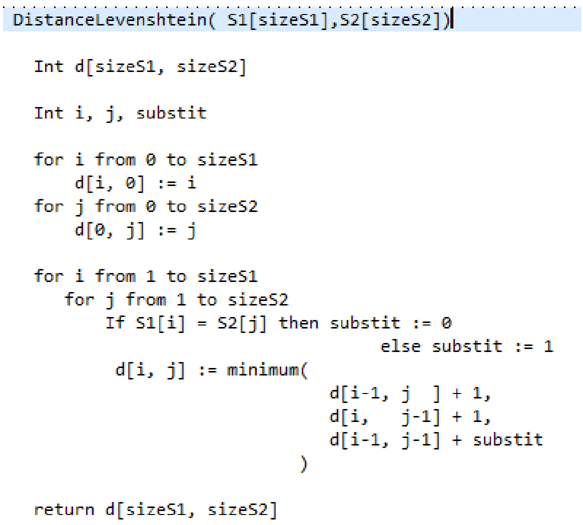
After Stemmatization:

ecol--age

Fig. 5. French lemmatization result to remove word affixes

B. Alignment

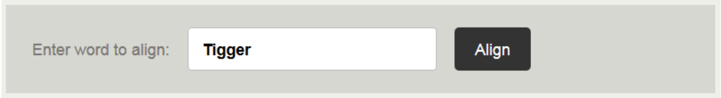
Several alignment methods exist such as Levenshtein's distance. Levenshtein's distance between words or strings of character is the editing distance similarity; or more simply, the indications calculation on the resemblance degree of these chains. Its definition is as follows:
If S_1 and S_2 are two words, Levenshtein's distance (dist) is the minimum number of substitutes, additions and deletions of letters from S_1 to S_2
dist satisfies the distances definition [4]:
S_1, S_2 and S_3 being any three words (possibly empty),
1. dist (S_1, S_2) is a real positive or zero
2. dist (S_1, S_2) = 0 if and only if $S_1 = S_2$
3. dist (S_1, S_2) = dist (S_2, S_1) (symmetry)
4. dist (S_1, S_3) is less than or equal to dist (S_1, S_2) + dist (S_2, S_3) (triangular inequality)
It may also be noted that dist (S_1, S_2) is an integer. The Levenshtein's distance algorithm is represented as below



```
DistanceLevenshtein( S1[sizeS1],S2[sizeS2])

   Int d[sizeS1, sizeS2]

   Int i, j, substit

   for i from 0 to sizeS1
       d[i, 0] := i
   for j from 0 to sizeS2
       d[0, j] := j

   for i from 1 to sizeS1
      for j from 1 to sizeS2
          If S1[i] = S2[j] then substit := 0
                              else substit := 1
           d[i, j] := minimum(
                            d[i-1, j  ] + 1,
                            d[i,   j-1] + 1,
                            d[i-1, j-1] + substit
                           )

   return d[sizeS1, sizeS2]
```

Fig. 6. Levenshtein's algorithm

In our application we designed an input entry to receive the term S_1 to be aligned and a French dictionary for S_2

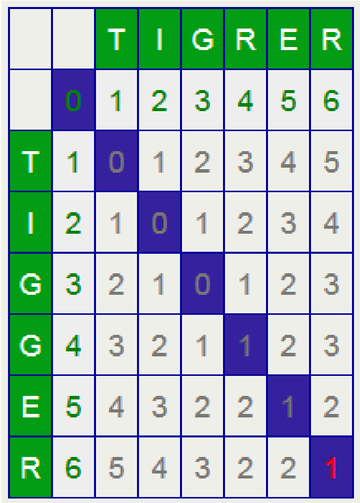


Enter word to align: Tigger

Fig. 7. Input text for the concept similarity

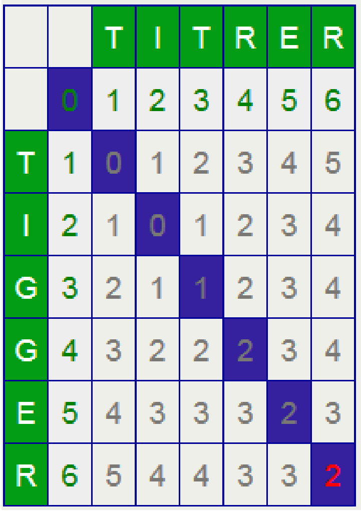
After alignment we have: TIGRER OR TITRER

To simplify the experiment we took the distance less than or equal to 2 as colored in red below



		T	I	G	R	E	R
	0	1	2	3	4	5	6
T	1	0	1	2	3	4	5
I	2	1	0	1	2	3	4
G	3	2	1	0	1	2	3
G	4	3	2	1	1	2	3
E	5	4	3	2	2	1	2
R	6	5	4	3	2	2	1

Fig. 8. a-Highly accurate result



		T	I	T	R	E	R
	0	1	2	3	4	5	6
T	1	0	1	2	3	4	5
I	2	1	0	1	2	3	4
G	3	2	1	1	2	3	4
G	4	3	2	2	2	3	4
E	5	4	3	3	3	2	3
R	6	5	4	4	3	3	2

Fig. 8. b - Accurate result

C. Construction of Ontology

1) Design of the term

The following Fig.9 comes from a mind map and is our basis for ontology construction where we find the subsumption or hierarchization of concepts.

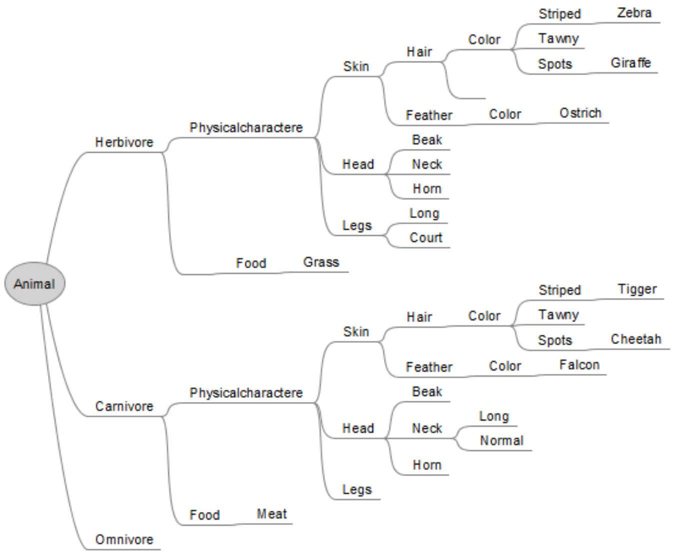


Fig. 9. Concept hierarchization map

2) Semantic distance of the term

To find the semantic distance between the concepts based on the mind map, the distance of the arcs 'depths is used and the basic formula is [5]

$$\frac{2 * depth(s)}{(depth(s1) + depth(s2))} \qquad (1)$$

S indicates the last common term of the concepts S1 and S2

TABLE I SEMANTIC DISTANCES BETWEEN CONCEPTS

*	Zebra	Giraffe	Tigger	Cheetah	Ostrich	Falcon
Zebra	1	0.75	0.13	0.13	0.50	0.13
Giraffe	0.75	1	0.13	0.13	0.50	0.13
Tigger	0.13	0.13	1	0.75	0.13	0.50
Cheetah	0.13	0.13	0.75	1	0.13	0.50
Ostrich	0.50	0.50	0.13	0.13	1	0.13
Falcon	0.13	0.13	0.50	0.50	0.13	1

We saw in this table that the distance between a zebra and a giraffe is 0.75 while a Falcon and a zebra is 0.13. We can also thus see that an ostrich is closer to a zebra than a falcon because the distance between zebra and ostrich is 0.5.

To further explain this correlation, the difference between a Zebra and a Giraffe, based on the map, is limited to only the color of the hair. In this case, it is once removed (one parent difference).

In the case of the comparison between Zebra and Ostrich, the difference between them is twice removed (2 parent differences), color and hair or feather.

The closer the difference is to 1, the less the differences are between the concepts.

3) Subsumption

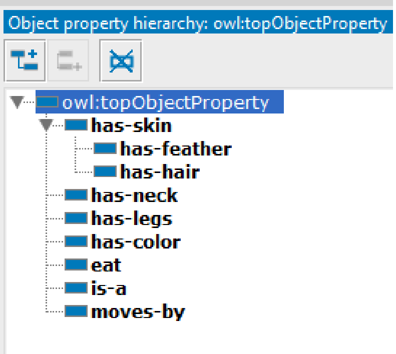
The elements representation of the real world is modeled and represented by concepts, roles and individuals. The subsumption relation arranges concepts and roles in the form of hierarchization.

The manipulations on the concepts and the roles are made according to the semantics. The two knowledge types taken into account are:

. The concepts associated with their components and

. Facts or assertions where the concepts and concepts instances are manipulated

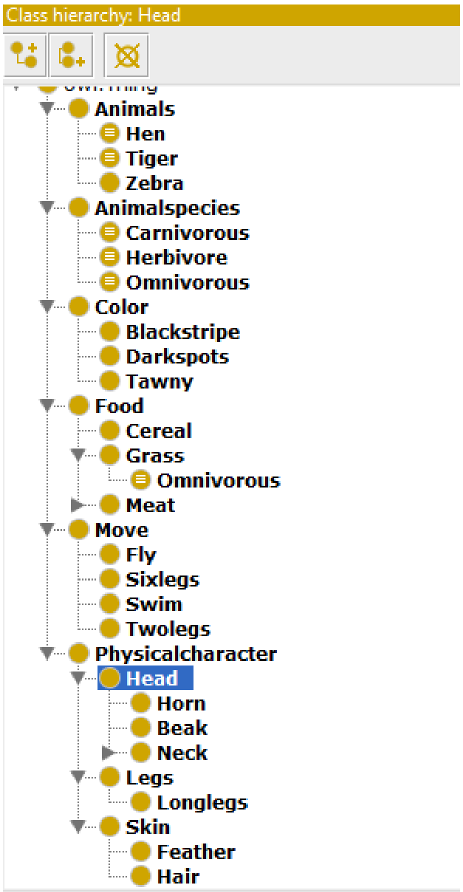
Fig.10 and Fig.11 represent respectively object property subsomption and concept subsomption under "protege tools".



Object property hierarchy: owl:topObjectProperty

- owl:topObjectProperty
  - has-skin
    - has-feather
    - has-hair
  - has-neck
  - has-legs
  - has-color
  - eat
  - is-a
  - moves-by

Fig. 10. Object property subsomption



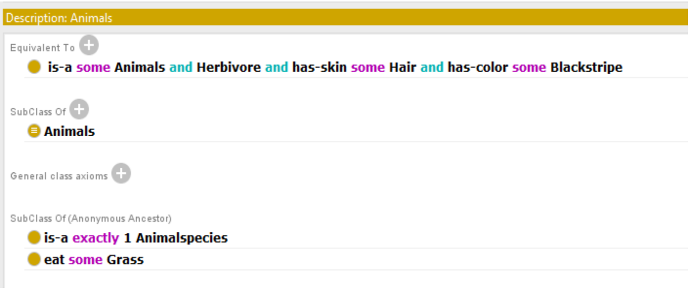
Class hierarchy: Head

- owl:Thing
  - Animals
    - Hen
    - Tiger
    - Zebra
  - Animalspecies
    - Carnivorous
    - Herbivore
    - Omnivorous
  - Color
    - Blackstripe
    - Darkspots
    - Tawny
  - Food
    - Cereal
    - Grass
      - Omnivorous
    - Meat
  - Move
    - Fly
    - Sixlegs
    - Swim
    - Twolegs
  - Physicalcharacter
    - Head
      - Horn
      - Beak
      - Neck
    - Legs
      - Longlegs
    - Skin
      - Feather
      - Hair

Fig. 11. Concept subsomption

4) Logical and rules

We use "protege tools" [6] to set up inference rules and put syntax transformation to infer a conclusion from concept and data property. The Fig 12 below describes a Zebra.



Description: Animals

Equivalent To

is-a some Animals and Herbivore and has-skin some Hair and has-color some Blackstripe

SubClass Of

Animals

General class axioms

SubClass Of (Anonymous Ancestor)

is-a exactly 1 Animalspecies

eat some Grass

Fig. 12. Inference rules

5) *Vizualization*

We use “Ontograph tool” [7] to visualize result in Fig. 13

D. Web semantic development

After having structured concepts, adding relations and applying logical rules in our ontology, we can have SPARQL queries launch to refine our search. The ontologies are exported in java code form and associated with “jena framework” to be able to develop them in a web site.

Another product coming from the ontology is the export of the data in xml file. The exported data can be used to be reused in other ontologies or to format them using the language XSLT or XSL in order to have a Semantic Web site.

Semantic Web is crucial concept in internet to apply the hypothesis of the open world. The open world presents itself as the inverse of the closed world. In the open world hypothesis, we cannot be sure and confirm that something does not exist until it has been explicitly proved and ruled that it did not actually exist. We can see this especially in the inference engine.

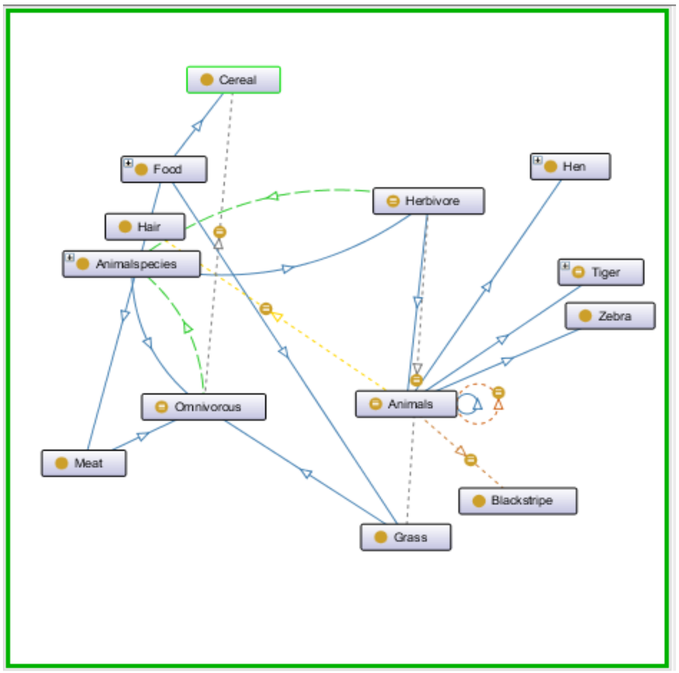


Fig. 13. Ontology vizualization on Ontograph

V. CONCLUSION

The Semantic Web thus offers all the possibilities to promote the evolution of human knowledge as a whole, but also it gives the possibility to interconnect computers allowing the latter to understand the information shared on the Semantic Web by structuring the data, which might be unstructured, utilizing different languages.

The Semantic Web is a continuation of the standard Web facilitating the information processing automation. It does not re-examine the standard Web such HTML and HTTP, etc. and the information is not defined in a natural language, but modeled using languages that can be interpreted by machines such as description language. Ontology forms the details of this modeling of knowledge and this will be the subject of our next article.

REFERENCES

[1] History of the Web, Oxford Brookes University 2002
http://www.w3c.it/education/2012/upra/documents/origins.pdf

[2] Identité numérique,Blaise Pascal Clermont-Fernand University, http://www.univ-bpclermont.fr/Ressources_Num/Les_reseaux_sociaux_web_web/co/module_Les_reseaux_sociaux.html

[3] Representing Knowledge in the Semantic Web,W3C, slide 7
http://www.w3c.it/talks/2005/openCulture/slide7-0.html

[4] Distance de Levenshtein, Wikipedia
https://fr.wikipedia.org/wiki/Distance_de_Levenshtein

[5] Z. Wu and M. Palmer, Verbs semantics and lexical selection, In Proceedings of the 32nd annual meeting on Association for Computational Linguistics, pp. 133-138. Association for Computational Linguistics, 1994, June.

[6] Stanford Center for Biomedical Informatics Research, Stanford University
http://protege.stanford.edu/

[7] Sean Falconer, Stanford University,
https://protegewiki.stanford.edu/wiki/OntoGraf