Multi-Viewpoints Semantic Annotation of XML Documents

Djama Ouahiba and Boufaida Zizette

***Abstract*—We are interested in the problem of semantic annotation of XML Web pages belonging to a heterogeneous domain by taking into consideration different viewpoints. This type of annotation requires the use of multi-viewpoints ontology of domain. For a given viewpoint, the page will not be fully annotated because only some elements of this page are relevant in this viewpoint. Our goal is to propose an approach to detect the elements and relationships between elements that are relevant in a given viewpoint. In the proposed approach, in order to manipulate the elements of the page, we used the tree representation of the page generated with DOMXML. We based on a semantic annotation of the leaves of the DOMXML tree in a given viewpoint to detect the elements and their relationships that are relevant in this viewpoint.**

***Index Terms*—Semantic Web, Semantic Annotation, Multi-Viewpoints Ontology, XML, DOMXML**

I. Introduction

THE multi-viewpoints semantic annotation is a technique based on the exploitation and the instantiation of a multi-viewpoints ontology. The latter is used to group different possible conceptualizations of the domain modeled according to different perspectives in a single ontology [18], [20]. Annotating a resource using multi-viewpoints ontology can give different interpretations to the content of this resource according to the viewpoints referred to.

For a given viewpoint, the resource would not be fully annotated because some elements of this resource are relevant in this viewpoint and others are not. To solve this problem, we propose to filter the content of the page; we only leave the elements that concern this viewpoint. The problem is how to detect, in the resource to annotate, the elements which are relevant in a given viewpoint.

We are interested in this work, in the multi-viewpoints semantic annotation of an XML page (semi-structured resource [5]). We use the tree representation of the page generated with DOMXML to manipulate the elements of this page. Our goal is to propose an approach that allows detection of elements and their relationships which are relevant in a given viewpoint.

In the following section, we present some related work. In sections three and four, we give a description of DOMXML tool and multi-viewpoints ontology. So we present our approach in section five and apply it on an example. We give some guidelines for future work in section eight. Finally, we conclude our work.

II. Related work

Current systems of semantic annotation [6], [7], [8], [9] and [10] cannot give more than one description to the resource content. As an example, in a classical ontology, the Apartment concept is defined with three attributes: room_nbr, rent and address.

"*The Apartment 10 of 4 rooms in Constantine city whose rental price is 250000 DA*" sentence can be semantically annotated (by using one of current systems) as an instance of Apartment concept.

But, it is more interesting to describe this sentence in three viewpoints: size, finance and localization. Thus, we need to use a multi-viewpoints ontology in order to give a multi-viewpoints description of each concept of this domain.

The Apartment concept can be defined in a multi-viewpoints ontology with an attribute room_nbr according to the size viewpoint, an attribute rent according to the financial viewpoint and an attribute address according to the size, localization and financial viewpoints. From those descriptions, we can define new concepts in each viewpoint. For example, in the size viewpoint: Small_Apartment and Large_Apartment are new concepts. Large_Apartment can be defined as an apartment which has, for instance, more than 2 rooms. From the financial viewpoint, an Expensive_Apartment is as an apartment which has more than 12000DA for rental price.

The multi-viewpoints semantic annotation of the sentence: *" Apartment 10 of 4 rooms in Constantine city whose rental price is 250000DA "* becomes an instance of Large_Apartment concept according to the size viewpoint and an instance of the Expensive_Apartment concept according to the financial one.

In a previous work [1], [2] we proposed an approach of semantic annotation from multi-viewpoints ontology. We have exploited the tree representation of the original page to be annotated. DOMXML was used to get a page for each viewpoint. DOM has several methods for manipulating XML files, as removeChild method which removes a target node. The node is not deleted in the original page, but only in the XML tree located in memory. As a result, a subtree is generated for each viewpoint. Then we get a DOM structure for each viewpoint instead of a page. This work was done manually by experts of the domain.

Manuscript received March 10, 2013; revised April 10, 2013.

Zizette Boufaida is with Lire Laboratory, University of Constantine 2, Algeria (corresponding author : 213-3181-8888; fax: 213-3181-8888; e-mail: zboufaida@ gmail.com).

Ouahiba Djama is with University of Constantine 1, Algeria (e-mail: djama.ouahiba@gmail.com).

In the current work, we propose a method to automatically extract these tree nodes without any intervention of human beings.

III. DOMXML Representation

DOM (Document Object Model) is a specification of the W3C (World Wide Web Consortium) defining the structure of a document as an object hierarchy [4].

DOM defines each XML tag as a node of the tree and the relationships between tags as edges. We distinguish two types of nodes: strings are represented by text nodes, and XML tags by element nodes. These nodes are typed elements by a tag name.

DOM has several methods for manipulating XML files. Here are the most used [4]:

RemoveChild: to delete a target node.

InsertBefore (newnode, oldnode): adds a new node newnode before the node existing oldnode.

AppendChild (newnode): adds a new node to the target node.

ReplaceChild (newnode, oldnode): This can replace a newnode by an oldnode.

So, the DOM is of particular interest for reading XML data and loading them into memory in order to modify the structure, to add or delete nodes, or to change the data in a node.

IV. Multi-viewpoints ontology

Multi-viewpoints ontology is a multiple description of the same universe of discourse according to different viewpoints. It is defined by a 4-tuple ([3] and [15]) of the form O = <C^G, R^G, Vp, M>, where:

- C^G is the set of global concepts,
- R^G is the set of global roles,
- Vp is the set of viewpoints and
- M is the set of bridge rules.

A viewpoint: is a partial description of a universe of discourse within a particular perception. A viewpoint is defined by a triple [15] VP_K= <C^L, R^L, A^L>, where:

- C^L is the set of local concepts,
- R^L is the set of local roles and
- A^L is the set of local individuals.

In the following, we give some fundamental definitions:

Global concept is a concept, which is seen by all viewpoints with some common properties. The latter are visible to all views and constitute the key of the global concept.

Local Concept is a concept that is seen and described locally according to a particular viewpoint.

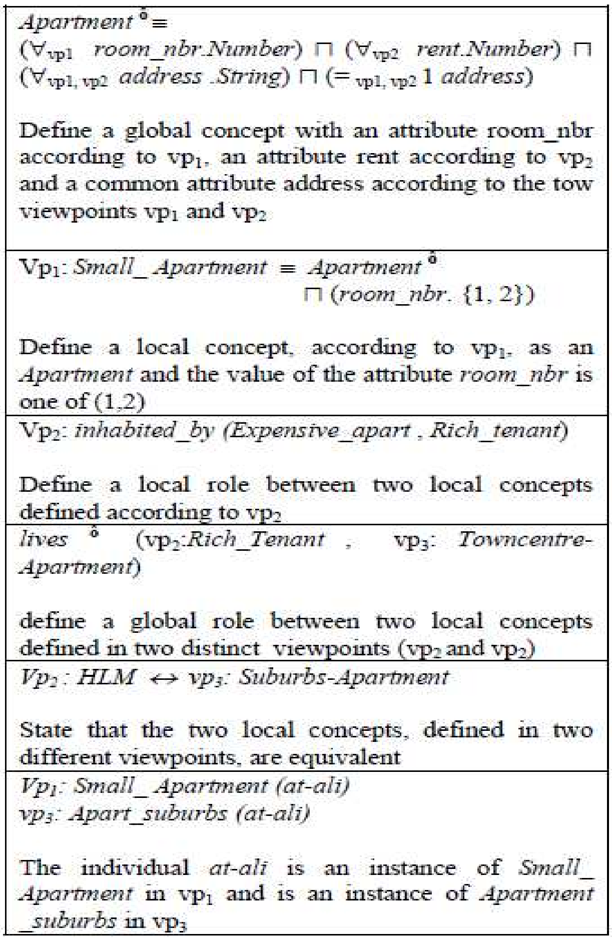
Global Role is a relationship between two local concepts defined in two different perspectives.

Local Role is a relationship between two local concepts defined in the same viewpoints.

Stamp (i.e. label) identifies for each ontological element (i.e. concept, role, individual) to be known by the viewpoint that it belongs to.

Bridge rule represents consensual relationships between two local concepts or two local roles represented in two different viewpoints.

Multi-Instantiation allows associating the same individual to several concepts according to different viewpoints.



Apartment $\overset{\circ}{}\equiv$ $(\forall_{vp1}\ room\_nbr.Number) \sqcap (\forall_{vp2}\ rent.Number) \sqcap (\forall_{vp1,vp2}\ address\ .String) \sqcap (=_{vp1,vp2} 1\ address)$ Define a global concept with an attribute room_nbr according to vp_1, an attribute rent according to vp_2 and a common attribute address according to the tow viewpoints vp_1 and vp_2
Vp_1: *Small_ Apartment* ≡ *Apartment* $\overset{\circ}{}$ $\sqcap (room\_nbr.\ \{1, 2\})$ Define a local concept, according to vp_1, as an *Apartment* and the value of the attribute *room_nbr* is one of (1,2)
Vp_2: *inhabited_by (Expensive_apart , Rich_tenant)* Define a local role between two local concepts defined according to vp_2
lives $\overset{\circ}{}$ (vp_2:*Rich_Tenant* , vp_3: *Towncentre-Apartment*) define a global role between two local concepts defined in two distinct viewpoints (vp_2 and vp_2)
Vp_2 : HLM ↔ *vp_3: Suburbs-Apartment* State that the two local concepts, defined in two different viewpoints, are equivalent
Vp_1: Small_ Apartment (at-ali) *vp_3: Apart_suburbs (at-ali)* The individual *at-ali* is an instance of *Small_ Apartment* in vp_1 and is an instance of *Apartment _suburbs* in vp_3

Fig. 1. Example of a multi-viewpoints ontology.

Fig. 1. Illustrates a multi-viewpoint ontology. In this example, three ViewPoints (VP) are considered: Size (vp1), Finance (vp2), and Localization (vp3).

V. The proposed approach

In this section, we propose an approach to detect in the page to annotate the elements and relationships relevant in a given viewpoint without the intervention of experts. Our approach is made of the following steps:

A. *Step1*

The first step is to build the DOMXML tree of the page to annotate using the existing tools.

B. *Step2*

Second step is to identify from the ontology, the viewpoint that we are interested in, for example a *Vp* viewpoint.

C. *Step3*

The third step is to detect text nodes that are relevant in the *Vp* perspective.

Usually, the text contained in a text node is a string that corresponds to a primitive entity (primitive concept or attribute of a concept) in the ontology. So for each text node *NTi*, we seek in the domain ontology, a local concept or attribute of a local concept that is seen and described locally according to a *Vp* viewpoint and can correspond to the string containing in the *NTi* text node (that is why to say primitive annotation of the text node *NTi*). If we cannot find an appropriate concept or attribute, then the text node *NTi* will be deleted from the RemoveChild because *NTi* is irrelevant in a viewpoint *Vp*.

D. *Step4*

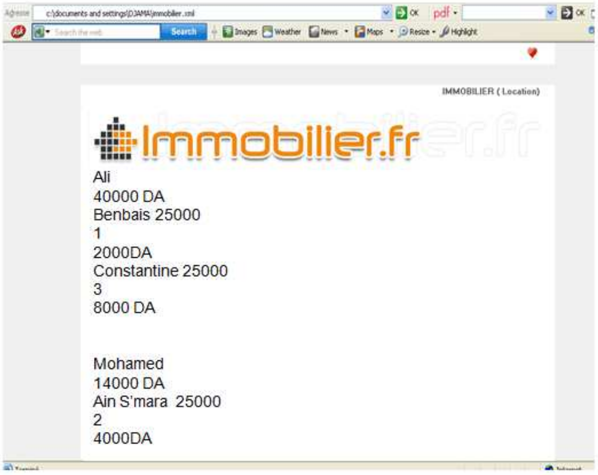
The fourth step is to detect node elements that are relevant in the viewpoint *Vp*. We start with the parent nodes of text nodes and we come up until the root. For each node element *NEi*, we seek its children with *hasChildNodes()* method which returns True if the node has children, False otherwise.

If *NEi* does not have children, it will be removed by using RemoveChild because *NEi* is irrelevant in a *Vp* viewpoint.

At the end of the process, we will obtain a subtree that represents the original page related to a *Vp* viewpoint.

VI. Case Study

We consider the XML web page of an estate agency used in [1] presented in Fig. 2.



IMMOBILIER (Location)

Immobilier.fr

Ali
40000 DA
Benbais 25000
1
2000DA
Constantine 25000
3
8000 DA

Mohamed
14000 DA
Ain S'mara 25000
2
4000DA

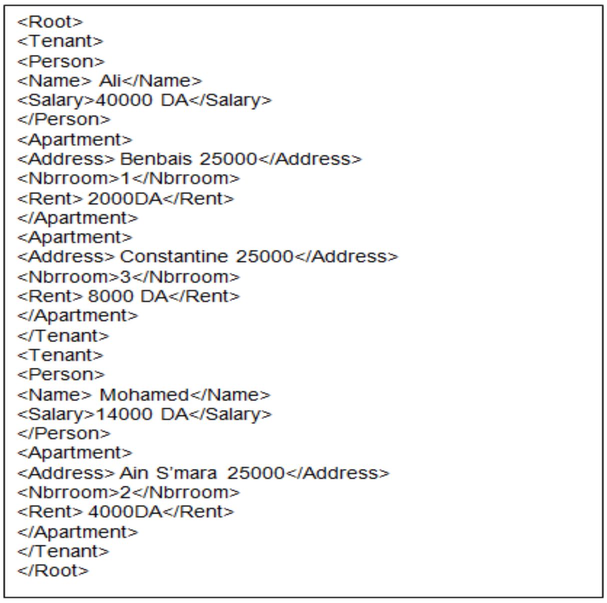
Fig. 2. Estate agency web page.

This example of an estate agency web page contains a list of tenants. Each one is identified by a name, can have a salary, and rents one or more apartments. Each apartment is identified by an address and is characterized by a room number and a rental price.

Three viewpoints are considered for this page: Size, Finance, and Localization. To apply our approach on the estate agency web page, we should follow the steps outlined in the previous section as follows.

A. *Step1*

The XML code of the estate agency page is shown in Fig. 3. and the corresponding DOMXML tree is shown in Fig. 4.



```
<Root>
<Tenant>
<Person>
<Name> Ali</Name>
<Salary>40000 DA</Salary>
</Person>
<Apartment>
<Address> Benbais 25000</Address>
<Nbrroom>1</Nbrroom>
<Rent> 2000DA</Rent>
</Apartment>
<Apartment>
<Address> Constantine 25000</Address>
<Nbrroom>3</Nbrroom>
<Rent> 8000 DA</Rent>
</Apartment>
</Tenant>
<Tenant>
<Person>
<Name> Mohamed</Name>
<Salary>14000 DA</Salary>
</Person>
<Apartment>
<Address> Ain S'mara 25000</Address>
<Nbrroom>2</Nbrroom>
<Rent> 4000DA</Rent>
</Apartment>
</Tenant>
</Root>
```

Fig. 3. XML code of an estate agency web page.

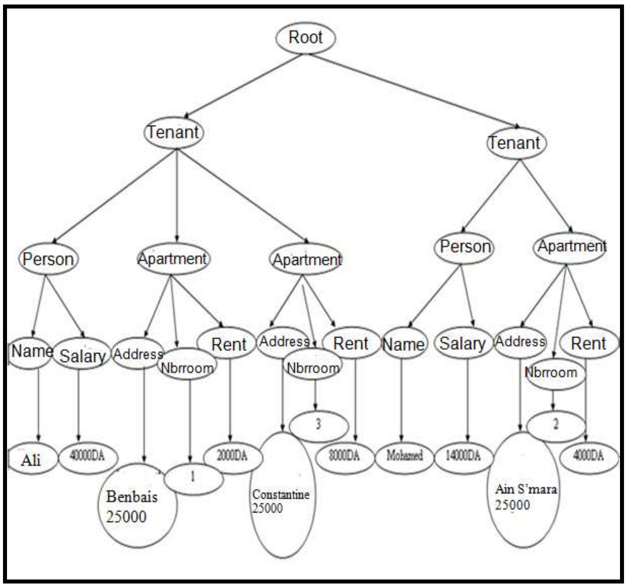


Fig. 4. DOMXML of an estate agency web page.

B. *Step2*

Among the concepts presented in the multi-viewpoints ontology of the estate agency [3] we have:

$Apartment^{\hat{o}} \equiv (vp1\ room\_nbr.Number) \sqcap (vp2\ rent.Monetary) \sqcap (vp1, vp2, vp3\ address.String) \sqcap (\geq vp1, vp2, vp3\ 1\ address) \sqcap (\leq vp1, vp2, vp3\ 1\ address)$

$Person^{\hat{o}} \equiv (vp2\ salary.Monetary) \sqcap (vp1, vp2, vp3\ name.String)$

Apartment is a concept defined with *room_nbr* attribute according to *vp1*, *rent* attribute according to *vp2* and *address* attribute according to the three viewpoints *vp1*, *vp2* and *vp3*.

vp1, *vp2* and *vp3* respectively represent size, finance and localization viewpoints.

We will apply our approach to the financial perspective. So in this view, the concepts *Apartment* and *Person* are defined as follows:

$vp2{:}Apartment \equiv (vp2\ rent.Monetary) \sqcap (vp2$

$address.String) \sqcap (\geq vp2\ 1\ address) \sqcap (\leq vp2\ 1\ address)$

$vp2{:}Person \equiv (vp2\ Salary.Monetary) \sqcap (vp2\ Name.String)$

C. Step3

We detect text nodes that are relevant in the financial perspective:

Current systems of semantic annotation [6], [7], [8], [9], [10], [17], [19], [21] use exclusively primitive concepts defined in an ontology to annotate a web page. They identify the strings that correspond to these concepts. Similarly, in this step we identify strings that correspond to entities (attribute of a concept or primitive concept) defined in the multi-viewpoints ontology of estate agency according to the financial perspective.

Ali and *Mohamed* strings correspond to *(vp1, vp2, vp3 name.String)* attribute.

Benbais 25000, Constantine 25000 and *Ain S'mara 25000* correspond to *(vp1, vp2, vp3 address.String)* attribute.

40000 DA and *14000 DA* correspond to *(vp2 salary.Monetary)* attribute.

2000 DA, 8000 DA and *4000 DA* correspond to *(vp2 rent.Monetary)* attribute.

1, 3 and *2* don't correspond to any terms in the ontology according to the financial perspective, so the corresponding text nodes will be deleted.

D. Step4

The fourth step is to detect element nodes that are relevant in the financial perspective. We seek the element nodes that have no children.

In our example, the element nodes that have no children are named *room_nbr*. So, these nodes will be deleted.

At the end of the process, the DOMXML subtree obtained according to the financial viewpoint is presented in Fig. 5.

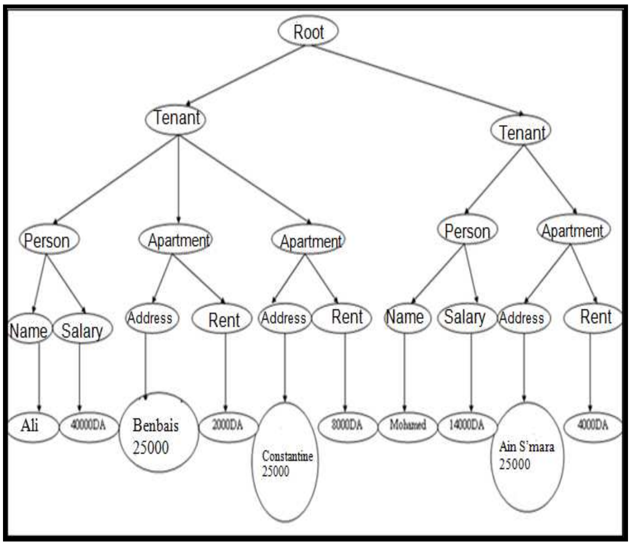


Fig. 5. DOMXML Subtree of estate agency page depending on Financial viewpoint

VII. Implementation

We programmed our approach in JAVA [11]. We also used JDOM [12] that can generate and manipulate the DOMXML tree. The ontology is implemented in OWL-MPV defined in [3] and [13]. We used JENA [14] to handle the domain ontology, and the OntoMat tool [9] for a primitive annotation.

VIII. Future work

We will use this approach as a step in an approach of multi viewpoints semantic annotation of heterogeneous resources to reduce the human intervention in order to achieve an automatic approach.

We will extend the XML language to multi-viewpoints XML in order to describe the content of a web page belonging to a heterogeneous domain which consists of a set of views.

IX. Conclusion

The multi-viewpoints semantic annotation consists of giving several interpretations of a web page belonging to a vast domain composed of several viewpoints. The principle is to associate a semantic annotation to a page in each viewpoint. To annotate the page in a given viewpoint, we must detect the page elements that are relevant in this viewpoint. For this, we were interested particularly in the semantic annotation of XML page belonging to the domain of interest. The latter is described by a multi-viewpoints ontology. We used DOMXML tree and primitive annotation of text nodes of the tree DOMXML (leaves of DOMXML tree) depending on a desired viewpoint in order to get a specific subtree to this viewpoint which represents a tree structure of the relevant elements in this viewpoint.

References

[1] O. Djama, and Z. Boufaida, "Une approche d'annotation sémantique à partir d'une ontologie multi-points de vue,". Thesis of magister, University Mentouri of Constantine, Algeria, 2010

[2] M. Hemam, O. Djema, and Z. Boufaida, "Vers une approche d'annotation sémantique à partir d'une ontologie multi-points de vue," presented at the Journées de l'école doctorale JED10, Annaba, Algéeia, 2010

[3] M. Hemam, and Z. Boufaida, "Développement des ontologies multi-points de vue : une approche basée sur la logique de description," PhD thesis, University Mentouri of Constantine, Algeria, 2012

[4] V. Apparo, and T. Pixley, " The NGLayout Document Object Model (DOM) ," Roadmap. Mozilla Organization, [Online]. Available: http://www.mozilla.org/newlayout/dom-toadmap.html

[5] M. Thiam, "Annotation sémantique de documents Semi-structurés pour la recherche d'information," PhD Thesis of University of Paris-Sud , France, 2010

[6] H. Davulcu, S. Vadrevu, and S. Nagarajan, "Ontominer: Automated metadata and instance mining from news websites," *IJWGS International Journal of Web and Grid Services,* vol. 1, no2, pp 196-221, 2005

[7] R. Baumgartner, S. Flesca, and G. Gottlob, "Visual web information extraction with lixto," *VLDB Very Large Data Base Endowment Inc,* vol. 10, no. 1, pp. 119-128, 2001

[8] M. Vargaz-vera, E. Motta, J. Domingue, M. Lanzoni, A.Stutt, and F. Ciravegna, "MnM: Ontology driven tool for semantic markup,". *In Proceedings of the Workshop on Semantic Authoring, Annotation et*

Knowledge Markup (SAAKM'02), EKAW, Lyon, France, 2002, pp.379–391

[9] C. Saathoff, N. Timmermann, S. Staab, K. Petridis, D. Anastasopoulos, and Y. Kompatsiaris, "M-OntoMatAnnotizer: Linking ontologies with multimedia lowlevel features for automatic image annotation," presented at AceMedia, 2006

[10] F. Ciravegna, and Y. Wilks, "Designing adaptive information extraction for the semantic web in Amilcare,". *In H. S. & S. S., Eds., Annotation for the Semantic Web, Frontiers in Artificial Intelligence and Applications, IOS Press, Springer-Verlag,* vol. 96, pp. 112–127, 2003

[11] K. Arnold, j. Gosling, and D. Holmes, "Le langage Java,". International Standard Book Number (ISBN), 978–2-7117-8671-8: 2001

[12] N. Cynober, "Manuipulation des données XML avec JAVA et JDOM. (developper.com) ," [Online]. Available: http://www. cynober.developpez.com/tutoriel/java/xml/jdom.html

[13] M. Hemam, and Z. Boufaida, "MVP-OWL: a multi-viewpoints ontology language for the Semantic Web," *IJRIS International Journal of Reasoning-based Intelligent Systems,* Vol. 3, no.3/4 pp. 147 - 15*5,* 2011

[14] I. Dickinson, "JENA ontology API," [Online]. Available: . http://jena.sourceforge.net/ontology/index.html

[15] M. Hemam, and Z. Boufaida, "Multi-Viewpoints Ontologies Representation: A Description Logics Based Approach,". *In 18th International Conference on Computer Theory and Application (ICCTA),* Egypt, 2008

[16] A. Abecker, and L. van Elst, "Ontologies for knowledge management,". S. Staab and R. Studer, (eds.) Handbook on Ontologies, Springer, ISBN 978-3-540-40834- 5, pp. 435–454, 2004

[17] S. Bechhofer, "The semantics of semantic annotation,". *In ODBASE. The conference on Ontologies, DataBases, and Applications of Semantics for Large Scale Information Systems provides*, Irvine, California, 2002

[18] L. van Elst, and A. Abecker, "Ontologies for information management: balancing formality, stability, and sharing scope," *IJEA Interntional Journal of Expert Systems with Applications,* vol. 22, no. 1, pp. 23:357–366, 2002

[19] J. Euzenat, "Eight Questions about Semantic Web Annotations," *IEEE Intelligent Systems*, vol. 17, no. 2 pp 55–62, 2002

[20] T. R.. Gruber, "Towards principles for the design of ontologies used for knowledge sharing,". *In N. Guarino and R. Poli, (eds.) Formal Ontology in Conceptual Analysis and Knowledge Representation. Kluwer Academic Publishers,* 1993

[21] V. Uren, "Semantic annotation for knowledge management: Requirements and a survey of the state of the art,". *JWR Journal of Web Semantics,* vol. 4, no. 1, pp14–28, 2006