Ontology Interoperability Techniques, The State of the Art

Siham Amrouch[1], Sihem Mostefai[2]
[1]Department of computer science
Med Cherif Messadia university
Souk Ahras, Algeria

[2]MISC Labo
Department of computer science
Mentoury University Constantine
Algeria
sihamamrouch@yahoo.fr, xmostefai@yahoo.com

ABSTRACT: *Since the emergence of semantic web, researchers are facing a great challenge to represent the semantics of data in order to bring the actual web into its full power and hence achieve its objective. Herein, ontologies have emerged as the best means to accomplish the task because of their ability to explicitly specify the modeled domains. Indeed, ontologies are also designed and developed to be shared between several organizations and information systems. Using ontologies as common or shared vocabularies requires a certain degree of interoperability between them. Therefore, the interoperability between different ontologies that model the same or similar domains is a challenging issue that can be handled by ontology mapping, alignment and merging processes. These later produce a Meta layer accessible by different applications and information systems, of course with preserving the semantics embedded in all the source ontologies. In this paper we aim to introduce the three main techniques of ontology interoperability, namely, ontology mapping, alignment and merging. These techniques constitute the basis for several other ontology interoperability techniques. This basis is built during the mapping discovery stage which is the meeting point for all ontology interoperability techniques. Several tools and algorithms for ontology Mapping, Alignment and Merging exist in the literature. In this paper, we will survey and compare the most outstanding ones after describing their common stage which is the mapping discovery process, then discuss the main forms of heterogeneities met and that have to be solved in order to bridge the semantic gap between the source ontologies.*

Keywords: Ontology interoperability, Ontology mapping, Ontolgy alignment, Ontology merging, Semantic integration, Linguistic mismatches

Received: 6 November 2011, Revised 12 December 2011, Accepted 17 December 2011

1. Introduction

The aim of the semantic web is to bring the actual web to its full potential. As it is reflected by its name, this objective can not be achieved without representing the semantics embedded in data within the web. Because of their potential of explicitly specifying the data conceptualizations, ontologies are considered as the best means to annotate data on the web. The most known and famous definition of ontologies is given by T. Gruber in 1993 [2], where the author defines an ontology as "*an explicit specification of a conceptualization*". But, as more and more people get involved, many ontologies are created.

Interoperability among different ontologies becomes essential to take advantage from the semantic web [4]. It is required for combining distributed and heterogeneous ontologies [5]. Hence, to promote the interoperability between these differently designed ontologies, ontology mapping, alignment and merging is a prominent solution. This, in turn, enables people and software agents to work in a more smooth and collaborative way [18]. Therefore, researchers of the semantic web community are facing a hard challenge to design and develop tools and algorithms that perform the mapping, alignment and merging of ontologies. We have chosen to explore these three notions because they are the basis for several other algorithms such as ontology translation, reconciliation, coordination, articulation, negotiation, etc. This basis is built during the mapping discovery stage, which is a common step for several ontology interoperability tools and algorithms. These variations of terms and tools make it difficult to identify problem areas and to understand the solution that they provide. These difficulties are due to the lack of comprehensive survey, standard terminology, hidden assumptions or undisclosed technical details and the dearth of evaluation metrics [8]. We may assume that the three notions that we will explore are the steps of a graduate process that begins from mapping stage, then the alignment of the mapped concepts and ends by merging the concepts that are mapped through relationships of similarity. Indeed, alignment process takes two input ontologies and produces a set of relationships between concepts that match semantically with each other. These matches are called "*Mappings*" [16, 2]. Ontology merging, and as its name implies, merges the concepts that match semantically with each other into a single concept, and then produces a unique ontology from two source ontologies. During their common mapping discovery stage, ontology mapping, alignment and merging have to solve the different types of mismatches to bridge the gaps between the source ontologies. Several classifications of mismatches exist in the literature [3]. In this paper, we have proposed a simple classification of the most frequent and more addressed mismatches, namely: the syntactic, semantic, and lexical mismatches. A lot of algorithms and tools of ontology mapping, alignment and merging have been designed and developed by different researchers. In this paper, we will survey the literature of the most known tools and mention their main drawback. At the end, we will compare them to each other according to a set of critical criteria.

The organization of this paper is as follows, in the next section, we will discuss and compare the three main techniques that handle the ontology interoperability processes. Then, we will describe their common module: the mapping discovery stage, and discuss the semantic integration process that serves the mapping discovery by promoting its performances using domain specific dictionaries to identify the different relationships between the source ontologies. In section 5, we classify the most common forms of mismatches during mapping discovery stage. Finally, we will describe, discuss and compare the most known algorithms and tools and we conclude by section 8.

2. Ontology interoperability techniques

Ontologies have emerged as the best means for explicitly describing semantics and contexts of data to be shared among different organizations and information systems. The ontology sharing character requires a certain degree of interoperability between them. This later is mainly handled by three basic processes:

Ontology Mapping which defines a formal expression that describes a semantic relationship between two (or more) concepts belonging to two (or more) different ontologies.

Ontology Alignment describes a semantic bridge between two (or more) ontologies through a set of matches between them. The source ontologies have to describe the same domain or related domains. These matches are called "*Mappings*". In summary, Mapping is the process of aligning two ontologies, a similar process used in data base schema is called "*Matching*".

Ontology Merging takes place once the mappings between the two source ontologies are identified, the mapped or aligned concepts are merged into a single one. That creates a new ontology O*m* from two input ontologies O' and O". See figure 1.

3. Semantic integration

This is a substantial research field that serves the semantic web by facilitating interoperability between different applications and/or knowledge sources such as ontologies. The semantic integration or even enrichment is performed through external resources such as domain specific dictionaries. An example of the most known and general designed computerized dictionary is WordNet[1]. We recall that WordNet is a computerized english dictionary where the basic unit is the concept. It uses two different

[1]http://www.wordNet.princeton.edu/wordNet

means to define the meaning of a word, the synsets and the lexical relations. A word is then defined by a set of synonyms (synset) and a definition.

Example: Board: synset = {board, blank} Definition: A piece of wood.

These external resources are used to avoid the limitations of the lexical aspect in the ontology merging process after their possible extensions according to their application domain. So, it is at this stage where the semantic aspect acts to support the mapping discovery process. Herein, the more the extension of the source ontologies is close to the same shared ontology, the easier will be the mapping identification process. In addition, reasoning and inference processes allowed by the ontology representation languages contribute in specifying the constraints of similar concepts merging.

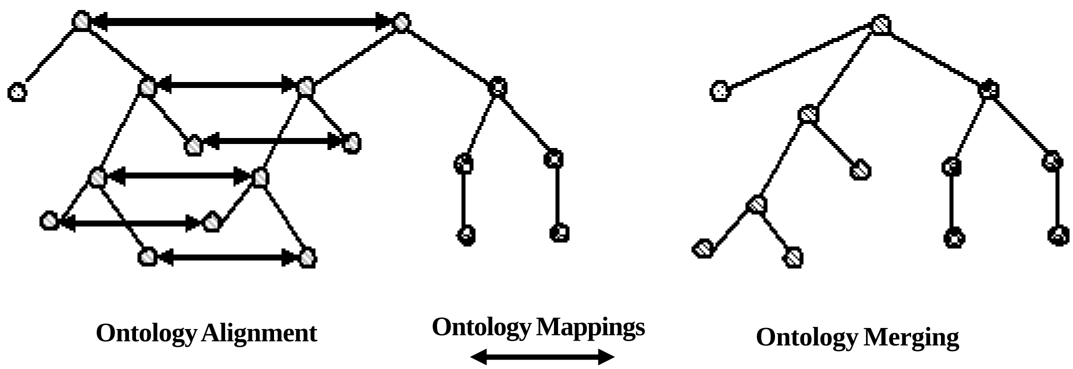

Figure 1. Ontology Mapping, Alignment and Merging

4. Mapping discovery

It's a common sub-process for the three processes of ontology Mapping, Aligning and Merging. Its objective is to identify similar concepts in source ontologies. The concepts judged similar will be matched by mapping relations (when Mapping or Aligning ontologies) or merged into a single concept (when Merging ontologies). We recall that an ontology is designed and developed to serve as a common vocabulary that is shared by multiple applications and communities of information systems developers, which is not possible with data bases. The reader can refer to [20] for more details on the limitations of data bases that may be avoided by ontologies. A common ontology is then accomplished by domain or application-specific concepts and properties by the knowledge engineer. The mapping discovery process can be effectively very easier if it is performed between two extensions that refer to the same common ontology. In addition, ontologies are developed to be manipulated by inference engines. And ontology representation languages are specified at the basis of reasoning. Then inference and reasoning have a prominent effect to discover mappings between the two ontologies under discussion. Based on these two aspects, Noy [16] scales two major architectures to find mappings between source ontologies:

Using a shared ontology, where the common ontology is accomplished by the application-specific concepts and properties. The More these extensions are consistent with the definitions provided in the common ontology, the easier will be the mapping discovery between the two extensions. Among the well known and common ontologies in the literature, there are SUMO, Suggested Upper Merged Ontology [12], which is designed with the objective of developing a common and standard ontology that may be used in data interoperability, information retrieval, automated inference and natural language processing. And DOLCE [7]. which is an upper ontology that is designed with the objective of providing a common framework as a reference to the semantic web ontologies that facilitates information sharing between them. DOLCE aims to capture the ontology categories highlighting the natural language and cognition.

Using heuristics and machine learning. Mappings discovery approaches, based on heuristics, for ontology mapping process are similar to matching discovery approaches for data base schemas and XML structures matching process, where lexical and structural components of definitions are used [9,17]. The ontology-based approaches are often more sophisticated because they exploit the semantics contained in ontologies (semantics of relations: is-a, part-of, attachment of properties to classes, property domain and co-domain definitions, etc.). In contrast with data base schemas, ontologies have much more specific constraints. These ones provide the main basis for the automatic mapping (matching) discovery methods.

5. Types of mismatches

Different types of mismatches may occur between different ontologies. Indeed different ontology designers opt for different representation languages and use different ontology editors to represent knowledge at different levels of granularity (detail). This explains the emergence of different forms of ontology mismatches. The identification of these types of mismatches is essential in order to solve them during the mapping, alignment or merging process. In what follows, we describe the three types of mismatches usually studied in the previous processes:

5.1 Syntactic mismatches

Two ontologies are syntactically heterogeneous if they are represented by different representation languages, such as OWL (ontology Web Language) for the first one and KIF (Knowledge Interchange Format) for the second one. To resolve this type of mismatches, simply transform the representation language of one ontology to the representation language of the other ontology. Herein, we state that sometimes the translation is difficult and even impossible or may lead to source information omission.

5.2 Lexical mismatches

Describes the heterogeneities between the names of entities, instances, properties or relations. In this type of mismatches we may find four forms of heterogeneities:

- **Synonyms.** The same entity is represented by two different names, eg. *Auto* and *car*.
- **Homonyms.** The same name represents two different entities, eg. the word "*hard*" has two different meanings in the following statements: *hard–disk-* and *hard-luck-* .
- **The same name in different languages (English, French) labels the same entity.** Eg. *computer* and *ordinateur*.
- **The same entities are named with the same words but with different syntactic variations**, such as abbreviations, optional prefixes or suffixes, eg. *example* and *exp*.

In a mapping process, this type of mismatches is treated by the expression of relations between lexical expressions (word or suite of words) used to name the entities to be mapped.

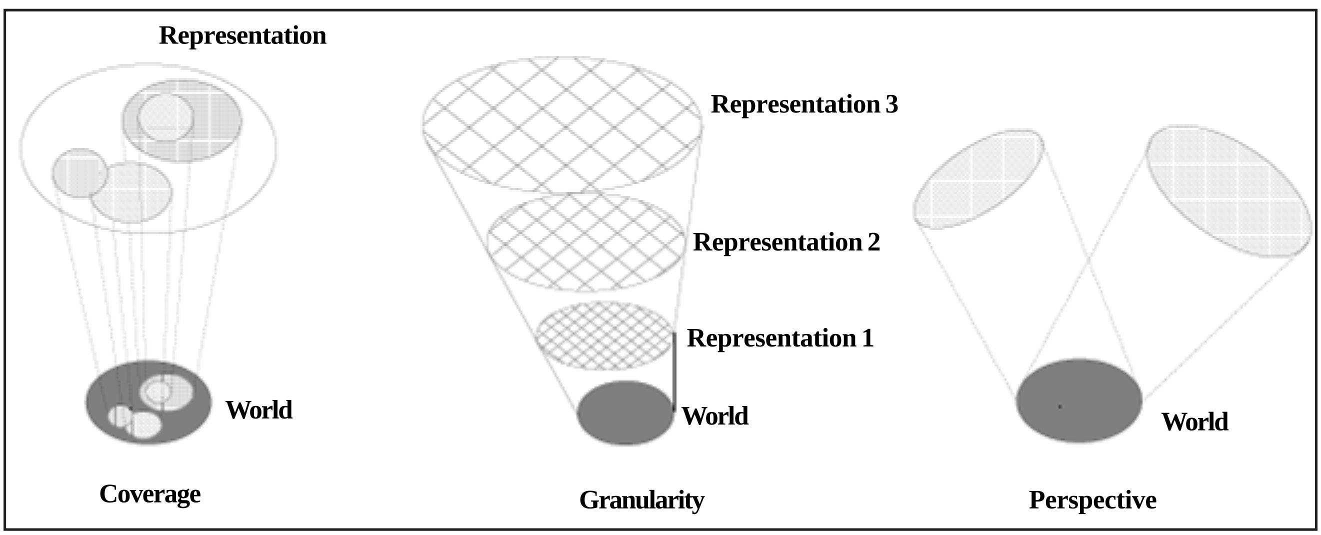

Figure 1. The three forms of mismatches according to [3]

5.3 Semantic mismatches

The mismatches identified at this level are related to the content of the input ontologies, in other words, the heterogeneities between all entities, properties and relationships represented in the input ontologies. According to [3], the conceptual or metaphysical mismatches (which matches the semantic mismatches in our paper) are classified into three abstract forms of mismatches:

Coverage. Two ontologies are different from each other in that they cover different (possibly overlapping) portions of the world (or even of a single domain). For example an ontology represents all the pieces of a gas turbine, while the other describes only the ones that have undergone a technical repairing intervention in a well determined interval of time.

Granularity. Two ontologies are different from each other in that one provides a more/less detailed description of the same entity. For example, an ontology describes the publications of an author X, while the other details these publications to distiguish between articles, books, biographies ...

Perspective. Two ontologies are different from each other in that one may provide a viewpoint on some domain which is different from the viewpoint adopted in another ontology. For example, one ontology describes a product of a production company from a commercial viewpoint, while the other describes it from a financial viewpoint. Figure 2 graphically illustrates these three forms of mismatches according to [3]:

6. Tools for ontology mapping, alignment and merging

Several tools for ontology Mapping, Alignment and Merging exist in the litterature. Most of these tools are semi-automatic and the design of fully automatic tools is usually a delicate issue. In this section, we outline the well known and recent ones:

6.1 FCA-Merge [19]

It's a method for automatic ontology merging. Its process is summarized as follows: First, from a set of input documents, popular ontologies (ontologies equipped by their instances) are extracted. Once the instances are extracted and the concept lattice is constructed, FCA-techniques are used to generate the formal context of each ontology. Using lexical analysis, FCA-techniques retreive specific information that combines a word or an expression to a concept if it has a similar concept in the other ontology. Then the two formal contexts are merged to generate the pruned concept lattice. Since the two contexts may contain the same concepts, this step may end with some conflicts and duplications. So, the knowledge engineer's role will be at the final stage of FCA-Merge, to resolve conflicts and eliminate duplications. In that, he uses a knowledge background about the domain and is automatically guided by FCA-Merge through a mechanism of "*question-answer*". This later aims to target his attention to a specific part of the ontology merging process. It should be mentioned that the major drawback of FCA-Merge is that it is based on instances to identify similar concepts, however, in most applications, there are no objects that are simultaneously instances in both source ontologies.

6.2 Smart, Prompt and Prompt -Diff [13, 14, 15]

Tools designed as plugins in protégé 2000, for ontology Mapping, Alignment And Merging. The Alignment or Merging process is initiated by mappings of linguistic similarities between the concepts of two source ontologies. Then they use the ontological structures provided by protégé 2000 (classes, slots, facets) to inform a set of heuristics to identify subsequent mappings:

• **SMART** goes beyond class name matches and looks for linguistically similar class names, studies the structure of relations in the vicinity of recently merged concepts, and matches slot names and slot value types to enhance the results of initial merging.

• **PROMPT** is an interactive ontology merging tool, it proposes a list of all possible merging actions (to-do list). After that, the knowledge engineer selects the appropriate proposals that go with his needs. Then, PROMPT automatically merges the selected pairs of concepts, provides the conflicts generated after merging (conflict-list) and proposes their appropriate solutions. Finally, the knowledge engineer selects the most suitable solutions.

• **PROMPT-Diff** uses various heuristics to align and compare two versions of the same ontology. The Alignment processes based on various heuristics are combined so that the results of the first will be used as input for the second and so on until the Alignment process does not give more changes. PROMPT-Diff is based on empirical assertions that a large portion of frames remains unchanged and that if two frames have the same type and the same names or very similar names so it is highly possible that one is a version (an image) of the other and therefore they are intended to be aligned or merged.

The major drawback of these algorithms is that they require a lot of human intervention, because every step of the algorithms is based on the user's decisions. Hence, when the ontologies are huge the mapping, alignment or merging will be very tedious and laborious task.

6.3 Chimaera [10]

An interactive ontology merging tool, where the knowledge engineer is charged to make decisions that will affect the merging process. Chimaera analyzes the source ontologies and if it finds linguistic matches the Merging is performed automatically, otherwise, the user is prompted for further action. Like PROMPT, Chimaera is an ontology editor plugin, namely Ontolingua, but they differ in the suggestions they make to their users with regard to the merging steps. The major drawback of this algorithm is that every thing requires human intervention. It is very similar to PROMPT.

6.4 Glue [6]

To find mappings between two source ontologies O' and O'', Glue uses machine learning techniques. So, for each concept of ontology O', Glue finds its most similar concept in ontology O" based on different practical similarity measures and several machine learning strategies. Each strategy uses a different information type such as word frequencies, instance values, instances names, formats of values, the characteristics of values distribution, etc. To cope with this different types of information, Glue opted for two types of machine learners: content learner and name learner. The former uses a text classification method and the later uses the same text classification method, but uses the full name of instance instead of its content. Then for each learning prediction it assigns a learner weight that indicates how much it trusts its predictions. Finally, a meta-learner that combines the predictions of the two learners is developped. The authors also used a technique called "*relaxation labeling*" to map the two hierarchies of the two ontologies. This technique assigns a label to each node of a graph and uses a set of domain independent constraints, such as, two nodes of concepts c' and c" match if the nodes of their neighbourhood[2] v(c') and v(c") also match, and a set of domain dependent constraints, such as, if X is an ascendent of Y and Y matches "direction" then X does not match "*sub-direction*". The major drawback of this algorithm is that it is based on instances to identify similar concepts. However, in most applications there are no objects that are instances in both ontologies.

6.5 ONION [11]

According to the authors, ontology Merging is inefficient because it is costly and not scalable. So, ONtology compositION system provides an articulation generator for resolving mismatches between different ontologies. The rules in the articulation generator express the relationship between two (or more) concepts belonging to the ontologies. Manual establishment of these rules is a very expensive and laborious task. And full automation is not feasible due to the inadequacy of natural language processing technology. The authors also elaborate on a generic relation for heuristic matches: Match gives a coarse relatedness measure and it is upon to the human expert to then refine it to something more semantic, if such refinement is required by the similar ontologies. The major drawback of this algorithm is that the way of identifying similar concepts is not specific. Also, the manual generation of articulation rules between them is tedious.

Criteria	CHIMAERA	ONION	PROMPT	FCA-MERGE	GLUE
1 Automation	Semi-automatic	Semi-automatic	Semi-automatic	Semi-automatic	Semi-automatic
2 Operation	Merging	Composition	Mapping+merging	Merging	Mapping
3 Framework	Ontolingua	Independent	Protégé 2000	Independent	Independent
4 Representation languages	Ontologua	Labeled and oriented graphs + Horn rules	Rdfs – owl	Concepts taxonomies of populated ontologies	Taxonomies
5 Mapping languages	Linguistic similarity measures	Articulation rules	Not applicable	Lexical similarity measure	Lexical similarity measure
6 External resources	No	WordNet	No	No	No
7 Lexical matching	No	No	Yes	Yes	Yes
8 Semantic matching	Yes	Yes	Yes	Yes	Yes
9 Instance matching	No	No	No	Yes	Yes
10 Name matching	No	Yes	Yes	No	Yes
11 property matching	No	No	No	No	Yes
12 Structure matching	Yes	No	Yes	Yes	Yes
13 User role	Takes decisions affecting the merging process	Validates the proposed mappings	Selects appropriate mappings from to-do list	Corrects conflicts and eliminates duplications	Selects the similarity computation function

Table 1. Comparaison of existing algorithms and tools

[2] *neighbourhood is defined to be the children, the parents or both.*

application. In their system, and after validating the suggested matches by a human domain expert, a learning component is included in the system which uses the user's feedback to generate better articulation in the future when articulating.

7. Comparison between existing algorithms and tools

So, as mentioned obove, several tools and algorithms for ontology mapping, alignment and merging exist in the literature. Most of them are semi automatic and the design of fully automated algorithms is usually a delicate issue. Herein, we compare the most known ones that exist in the literature such as: CHIMAERA, ONION, PROMPT, FCA-MERGE and GLUE, through a set of critical criteria. See Table1.

8. Conclusion

Ontology Mapping, Alignment and Merging play a key technology role to handle interoperability between source ontologies. Several tools and algorithms have been designed and developed by the researchers in the AI community. In this paper, we have surveyed the literature of ontology interoperability tools and especially, those based on mapping, alignment and merging processes. We have also, described their mapping discovery stage and the effect of the semantic integration step to promote its performances when solving the different forms of mismatches. These ones have been simply classified into three classes of mismatches, the syntactic, semantic and lexical mismatches. At the end, we have compared the algorithms of the existing tools and their performances through a set of critical criteria.

References

[1] Barwise, J., Sligman, J. (1999). Information flow: the logic of distributed systems, Cambridge University Press, ISBN 0 521 58383 1, United Kingdom.

[2] Benerecetti, M., Bouquet, P., Ghdini, C.(2000) .Contextual reasoning distilled. Journal of Théoritical and Experimental Artificial intelligence, 12(3), 279-305.

[3] Bouquet, P., Ehrig, M., Euzenat, J., Franconi, E., Hitzler, P., Krötzsch, M., Serafini, L., Stamou, G., Sure, Y., Tessaris, S. (2005). D2.2.1 Specification of a common framework for characterizing alignment. Delivrable 2.2.1, Knowledge Web.

[4] Cho, M., Kim, H., Kim, P. (2006). A new method for ontology merging based on concept using WordNet, Conference on Advanced Communication Technology, proceeding of ICACT' ,(3).

[5] Choi, N., Song, I. Y., Han, H. (2006). A Survey on Ontology Mapping, SIGMOD Record, 35(3).

[6] Doan, Madhaven, J., Domingos, P., Halevy, A. (2002). Learning to map between ontologies on the semantic web. *In*: proceedings of the 11th International World Wide Web Conference (WWW 2002), Hawaii, USA.

[7] Gangemi, A., Guarino, N., Masolo, C., Oltramari, A. (2003). Sweetening Wordnet with DOLCE. AI Magazine, 24 (3) 13-24.

[8] Kalfoglou, Y., Schorlemmer, M. (2003). Ontology mapping, the state of the art. The Knowledge Engineering Review, 18(1),1-31.

[9] Madhaven, J., Bernstein, P., Rahm, E. (2001). Generic schema matching using Cupid. *In*: the 27th International Conference on Very Large Data Bases (VLDB'01), Rome, Italy.

[10] McGuiness, D., Fikes, R., Rice, J., Wilder, S. (2000). An Environment for Merging and Testing Large Ontologies, *In*: Proceedings of the 17th International Conference on Principales of Knowledge Representation and Reasoning (KR-2000), Colorado, USA.

[11] Mitra, P., Wiederhold, G. (2002). Resolving terminological heterogeneity in ontologies. ECAI'02 workshop on ontologies and semantic interoperability, Lyon, France.

[12] NilesPease, A. (2001). Towards a standard upper ontology, second International Conference on Formal Ontology in Information Systems, (FOIS), Ogunquit, Maine.

[13] Noy, N. F., Musen, M. (1999). SMART, Automated Support for Ontology Merging and Alignment. *In*: proceedings of the 12th Workshop on Knowledge Acquisition, Modelling and Management (KAW'99), Banff, Canada.

[14] Noy, N. F., Musen, M. (2000). PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. *In*: Proceedings of the 17th National Conference on Artificial Intelligence (AAAI'00), Austin, TX, USA.

[15] Noy, N. F., Musen, M. (2002). PROMPTDIFF: A Fixed-Point Algorithm for comparing ontology versions. *In*: Proceedings of the 18th National Conference of Artificial Intelligence (AAAI'02), Edmonton, Alberta, Canada.

[16] Noy, N. F. (2004). Semantic Integration: A Survey Of Ontology-Based Approaches. SIGMOD Record, 33(4) 65-70.

[17] Rahm. E., Bernstein, P. A. (2001). A survey of approaches to automatic schema matching, VLDB Journal, 10(4).

[18] Saleem, A. (2006). Semantic Web Vision: survey of ontology mapping systems and evaluation of progress, Master Thesis, Intelligent Software Systems, Thesis no: MCS-2006:13, School of Engineering, Blekinge Institute of Technology, Box 520, SE – 372 25 Ronneby, Sweden.

[19] Stemme. G., Maedche, A. (2001). Ontology Merging for Federated Ontologies on the Semantic Web. *In*: proceedings of the International Workshop for Foundations of Models for Information Integration (FMII-2001), Viterbo, Italy.

[20] Ushold, M., Grüninger, M. (2004). Ontologies and semantics for seamless connectivity, SIGMOD Record, 33(3).

Biographies

Siham Amrouch is presently PhD student at UBMA University, Annaba, Algeria. An assistant lecturer at CUSA University, Souk Ahras, Algeria. In 2009, she obtained the Magister Degree in Artificial Intelligence. She received her engineer diploma in Artificial Intelligence in 2006 from UBMA, Annaba University. Her research interests include ontology engineering, Information Retrieval, Semantic Web and Handwritten Arabic Recognition.

Sihem Mostefai is presently associate professor in the Computing Science Department at Mentouri University, Constantine, Algeria. In 2007, she obtained a state Doctorate in Product Lifecycle Management (PLM) applied to the case of mechanical products. She received her MSc in computer graphics in 1993, in addition to an engineer Diploma in computer systems in 1989 from the same university. Her research interests include Information Systems in industry, Product modeling, feature based modeling, ontology engineering, Information Retrieval and Semantic Web.