KNOWLEDGE ACQUISITION WITH OM
A Heuristic Solution

Adolfo Guzman-Arenas and Alma-Delia Cuevas-Rasgado
Centro de Investigación en Computación Av. Juan de Dios Batiz, s/n, Zacatenco, 07738 México City
México and Instituto Tecnológico de Oaxaca Av. Ing. Victor Bravo Ahuja 125 and Calzada Tecnológico
68030 Oaxaca city, México

Keywords: Knowledge acquisition, knowledge models, ontology merging, ontology fusion, knowledge representation.

Abstract: Knowledge scattered through the Web inside unstructured documents (text documents) can not be easily interpreted by computers. To do so, knowledge contained from them must be extracted by a *parser* or a person and poured into a suitable data structure, the best form to do this, are with ontologies. For an appropriate merging of these "individual" ontologies, we consider repetitions, redundancies, synonyms, meronyms, different level of details, different viewpoints of the concepts involved, and contradictions, a large and useful ontology could be constructed. This paper presents OM algorithm, an automatic ontology merger that achieves the fusion of two ontologies without human intervention. Through repeated application of OM, we can get a growing ontology of a knowledge topic given. Using OM we hope to achieve automatic knowledge acquisition. There are two missing tasks: the conversion of a given text to its corresponding ontology (by a combination of syntactic and semantic analysis) is not yet automatically done; and the exploitation of the large resulting ontology is still under development.

1 INTRODUCTION

These days computers are not anymore isolated devices but they are important entry points in the world-wide network that interchanges knowledge and carries out business transactions. Nowadays, using Internet to get data, information and knowledge interchange is a business and an academic need. Despite the facilities to access Internet, people face the problem of heterogeneous sources, because there are no suitable standards in knowledge representation. This paper is designed for this necesity of businesses and academia.

Many answers that people require involve accessing several sources in the Internet, which are later manually merged in a "reasonable" way. Merging the information is an important task. Many languages and tools (DAML+OIL (URL 15), RDF (URL 16) and OWL (URL 14) have been developed to describe and process Internet content but, unfortunately, they don't have enough expressiveness to detail knowledge representation.

Given a document written in a natural language, it is required that the computer deciphers the information in it and converts it to a suitable notation (its knowledge base) that preserves relevant knowledge. This knowledge base can be an ontology. To describe a knowledge domain, an ontology represents the knowledge through nodes that are joined through relations. Current works that merge ontologies (Prompt (Noy, *et al*, 2000), Chimaera (McGuinness, *et al*, 2000), OntoMerge (Dou *et al*, 2002), FCA-Merge (Stumme *et al*, 2002) and If-Map (URL 1)) rely on *the user* to solve the most important problems found in the process: inconsistencies and adequate knowledge extraction. In our fusion also these inconsistencies appear buy they are solved by OM. OM, the merging algorithm that we will explain, is totally automatic. This algorithm solves by itself the inconsistencies found in the process. In some cases OM applies the confusion algorithm (Levachkine, S., and Guzman, A. 2004) at the moment other solutions are studying. Two important contributions herein presented to obtain better advantage of the Web resources are:

- A new notation to represent knowledge using ontologies, called OM (Ontology Merging) Notation, and
- An automatic algorithm to merge ontologies, called OM Algorithm.

This paper it is a summary of (Cuevas, 2006), on it we describes the second contribution. The first

Guzman-Arenas A. and Cuevas-Rasgado A. (2008).
KNOWLEDGE ACQUISITION WITH OM - A Heuristic Solution.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems - AIDSS*, pages 356-363
DOI: 10.5220/0001717703560363

contribution, the OM notation it is in (Cuevas, 2006).

OM fuses two ontologies (this is our main contribution), we are *not* doing any of: • ontology comparison, this has been done by COM (see below) and others; • ontology alignment, as Prompt (Noy & Musen 2000); • building a gigantic unique ontology, this may or may not be done (see Discussion); • an ontology server, like Protegé (Noy & Musen 2000). We neither pretend that our notation to be superior to others (RDF (URL 16), say).

Some real examples appear where the texts (unloaded by Internet) have became manually to ontologies, OM have been applied and the result of the fusion has been verified manually. We have a work in the future; to make parser that turns of text to ontologies and a deductor of intelligent questions to verify the result of the fusion.

2 KNOWLEDGE ACQUISITION

The plan to follow is to acquire many "individual" ontologies distilled from text documents, and then to fuse them, two at the time, into a larger one. The conversion of text into ontologies is hard, it is made by a parser or syntactic analyzer, and will not be covered in this work. This paper is focused to the fusion of ontologies (arising from different sources) amount computers. During this fusion the same problems (redundancy, repetition, inconsistency…) arise; the difference is that the machines have no common sense (Lenat &Guha, 1989) and the challenge is to make them to understand that *beneficial* is the same to *generous*, and that *triangle* represents: • a three-sided polygon; • a musical percussion instrument; or • a social situation involving three parts. The computer solution to fusion should be very close to people's solution.

This paper explains a process of union of ontologies in automatic and robust form. Automatic because (unaided) computer detects and solves the problems appearing during the union, and robust because it performs the union in spite of different organization (taxonomies) and when the sources are jointly inconsistent.

The fusion is demonstrated by samples taking of real Web documents and converting them by hand to ontologies. These are then fed to the computer, which produces (without human intervention) a third ontology as result, like in (Kotis, 2006). This result is hand-compared with the result obtained by a person. Mistakes are below (section 3.3, Table 1).

2.1 Ontology

Formally, an ontology is a hypergraph (C, R) where C is a set of *concepts*, some of which are *relations*; and R is a set of restrictions of the form (r c_1 c_2 … c_k) between relation r and concepts c_1 to c_k. It is said that the *arity* of r is k. Check that relations are also concepts.

An important task when dealing with several ontologies is to identify most similar concepts. We wrote COM (Olivares, 2002) that finds this similarity throught ontologies.

2.2 The Role of Ontologies

An ontology is a data structure where information is stored as nodes (representing concepts such as `hammer`, `printer`, `document`, appearing in this paper in `Courier font`) and relations (representing restrictions among nodes, such as cuts, transcribes or hair color, they appear in this paper in Arial Narrow font, how in (`hammer` cuts `wood`), (`printer` transcribes `document`). Usually, the information is stored as "high level" and it is known as knowledge.

Ontologies are useful when arbitrary relations need to be represented, because it offers more freedom to represent different types of concepts and relations.

Currently notations to represent ontologies are DAML+OIL (Connoly *et al*, 2001), RDF (URL 16) and OWL (URL 14). These languages are a notable accomplishment, but it does not have enough features:

- A relation can not be a concept. For instance, if color is a relation, it is difficult to relate color to other concepts (such as `shape`) by using other relations.
- Partitions (subsets with additional properties) can not be represented (Gómez P. A., and Suárez F. 2004).

2.3 Exploitation of Distributed Content

Works exist (McGuinness *et al,* 2000; Noy & Musen, 2000 and Dou *et al,* 2002) that perform the union of ontologies in a semiautomatic way (requiring user's assistance). Others (Kalfoglou & Schorlemmer, 2002 and Stumme & Maedche, 2002) require ontologies organized in a formal way, and to be consistent with each other. In real life, ontologies come from different sources are not likely to be similarly organized, nor they are expected to be mutually consistent. The automation of fusion needs to solve these problems.

3 INTEGRATION DESIGN

This section explains the procedure that follows OM as well as the cases in which it has been applied.

3.1 The OM Algorithm

This algorithm fuses two ontologies (Cuevas, 2006) A and B into a third ontology C = A ∪ B containing the information in A, plus the information in B not contained in A, without repetitions (redundancies) nor contradictions. OM proceeds as follows:

1. Ontology A is copied into C. Thus, initially, C contains A.

2. Using the algorithm COM (Olivares, 2002) seek in B each concept c_C of C called the *most similar concept* of C into B. The search starts from the root concept of C, taking each one of its son of this until to visit all the concepts of C. There are just two options:

   A. If c_C has a most similar concept *cms* ∊ B, then:

      i. Relations that are synonyms (section 3.2, example 2) are enriched.

      ii. New relations (inluding Partitions) that *cms* has in B, are added to c_C. Concepts that which are in the new relations which come from *cms* are copied to C (if they are not).

      iii. Inconsistencies (section 3.2) between the relations of c_C and those of *cms* are detected.

         1. If it is possible, by using *confusion* algorithm (Levachkine & Guzman 2007), to resolve the inconsistency into c_C.

         2. When the inconsistency can not be solved, OM rejects the contradicting information in B, and c_C keeps its original relation from A.

   B. If c_C does not have a *cms* ∊ B, go to step 3.

3. It takes the next descendant of c_C into C. Goes back to step 2 until all the nodes of C are visited (including the new nodes that are being added by OM). (Cuevas, 2006) explains how this works.

3.2 Problems that OM Solves

In this section, figures show only relevant parts of ontologies A, B and the resultant C, because they are too large to fit.

Example 1: Merging Ontologies with Inconsistent Knowledge. Differences between A and B could be the following: different subjects, different names of concepts or relations; repetitions; reference to the same facts but with different words; different level of details (precision, depth of description); different perspectives (people are partitioned in A into male and female, but in B they are young or old); and contradictions. For example A (URL 12) contains: The Novelist, poet and writer Don Miguel de Cervantes was born in Alcalá de Henares, Madrid while B contains: The writer, (URL 13) poet and romantic Cervantes was born in Madrid, Spain. Both ontologies duplicate some information (about Cervante's place of birth), different expressions (novelist, poet and writer versus writer, poet and romantic), different level of details (Don Miguel de Cervantes versus Cervantes), and contradictions (Alcalá de Henares, Madrid vs. Madrid, Spain). A person will have in her mind a consistent combination of information: Cervantes and Don Miguel de Cervantes are not the same person, or perhaps they are the same, they are synonyms. If she knows them, she may deduce that Don Miguel de Cervantes is the complete name of Cervantes. We solve these problems everyday, using previously acquired knowledge and common sense knowledge (Lenat & Guha, 1989), which computers lack. Also, they did not have a gradual and automatical way to grow their ontology. OM measures the inconsistency (of two apparently contradicting facts) by asking conf (Levachkine & Guzman 2007) to determine the size of the confusion in using Alcalá de Henares in instead of Madrid and vice versa, or the confusion of using Don Miguel de Cervantes instead of Cervantes.

OM does not accept two different concepts for a birthplace. If A said that *Don Miguel de Cervantes was born in Alcalá de Henares* and B says that *Cervantes was born in Madrid*, OM chooses *Alcalá de Henares* instead of *Madrid* because it is more specific place while *Madrid* that is more general (it deduces this from a hierarchy of places in Europe). Small inconsistencies cause C to retain the most specific value, while if it is large, OM keeps C unchanged (ignoring the contradicting fact from B). In case of inconsistency, A prevails. This is also because we can consider that an agent's previous knowledge is A, and that such agent is trying to learn ontology B. In case of inconsistency, it is natural for the agent to trust more its previous knowledge, and to disregard inconsistent knowledge in B as "not trustworthy" and therefore not acquired

– the agent refuses to learn knowledge that it finds inconsistent, if the inconsistency is too large.

Example 2: Joining Partitions, Synonym Identification, Organization of Subset to Partition, Identification of Similar Concepts, Elimination of Redundant Relations and Addition of New Concepts. This example is accomplished through eight cases:

a. ***Relation that are not Copied, if there are Contradictions.*** Figure 1 shows relation neutron without charge, that mean: neutron does not have charge. Another ontology containing: neutron with positive charge that contradict what it is before (see Figure 1), thus the contradicting relation will not be copied into the fused result.

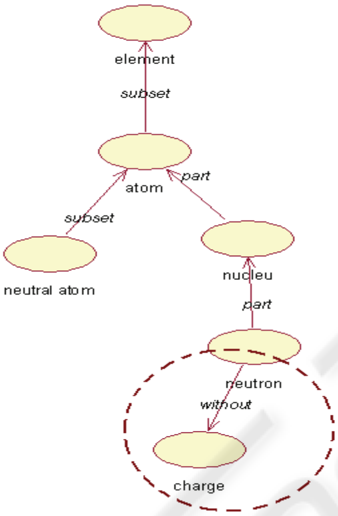

Figure 1: The concept neutron has a relation without charge that OM recognizes as absence of that property in neutron. Another ontology having relation has charge, has negative charge linked to neutron, will be an inconsistent information. When fusing both ontologies, OM will not copy neutron has negative charge into the result.

b. ***Copying New Partitions.*** building is a partition in A (indicated in the small circle) of Monte Albán, therefore it is added to the resulting ontology C (Figure 2).

c. ***Copying new concepts.*** Concepts Mixtec and Mexican Republic were not in A, but they appear in B. Therefore, they were copied by OM to C (Figure 2).

d. ***Reorganization of Relations.*** Relation located in appears twice but with different values, therefore they are added to C because it is possible that the relation to have several values (Figure 2). In case of single-valued relations, confusion algorithm (Levachkine & Guzman, 2007) is used.

e. ***Synonym Identification.*** Relation built by in A (Figure 2) and made by in B are both synonymous because in the definition of make by in B (the words that defines it, between parenthesis) we found the word *build*. OM fuses in C the relation built by of A, with both descriptive phrases *build* and *make by*.

f. ***Identification of Similar Concepts.*** In the Figure 3, concept sculpture of a jaguar in A and throne in the shape of jaguar in B have the same properties (Color and its value) therefore, OM fuses them into a single concept. The same happens with *El Castillo* and Pyramid of Kukulkan because they have the same properties and children.

g. ***Removing Redundant Relations.*** In A, Chichen Itza is member of pre-Columbian archaeological site (Figure 3), which is a member of archaeological sites. In B, Chichen Itza is member of archaeological site (which is parent of pre-Colombian archaeological site in B), therefore it is eliminated in C because it is a redundant relation. In C, pre-Columbian archaeological site is parent of Chichen Itza. The same occurs with Isotope subset of chemical element in figure 5, where A shows that Isotope is a subset of chemical element and B shows that Isotope is a subset of atom. OM check that Isotope is a subset of atom that is at the same time a subset of chemical element and Isotope is a subset of chemical element (OM erases this last relation because it is redundant). But not in the Figure 4 where the relation isotope subset of chemical element is different to isotope member of atom and atom subset of chemical element.

h. ***Organization of Subset to Partition.*** In the building partition in A there are six subsets (Figure 3): Ballcourt, Palace, Stage, Market and Bath. OM identifies them in B, where they appear as subsets of Chichen Itza. OM copies then into C like a partition and not as simple subsets. OM prefers the partition to just subset.

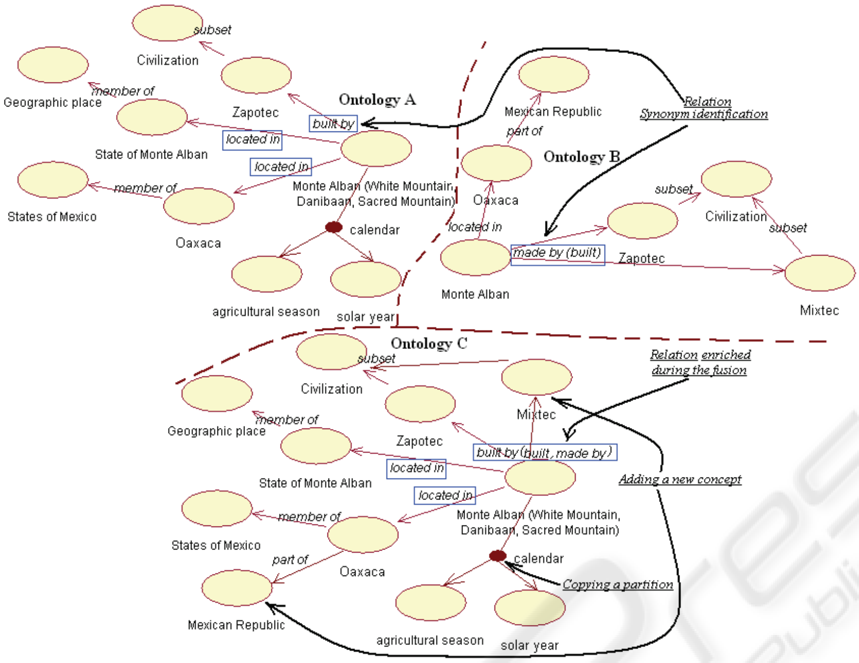

Figure 2: Ontology A describes `Monte Alban`. From a different view point, ontology B does the same. Ontology C is the result of OM fusing them. The relation built by in ontology A and make by in B are identified (case **e**) as synonyms, hence it is enriched during the fusion (into C). Only relevant parts of A, B and C are shown.

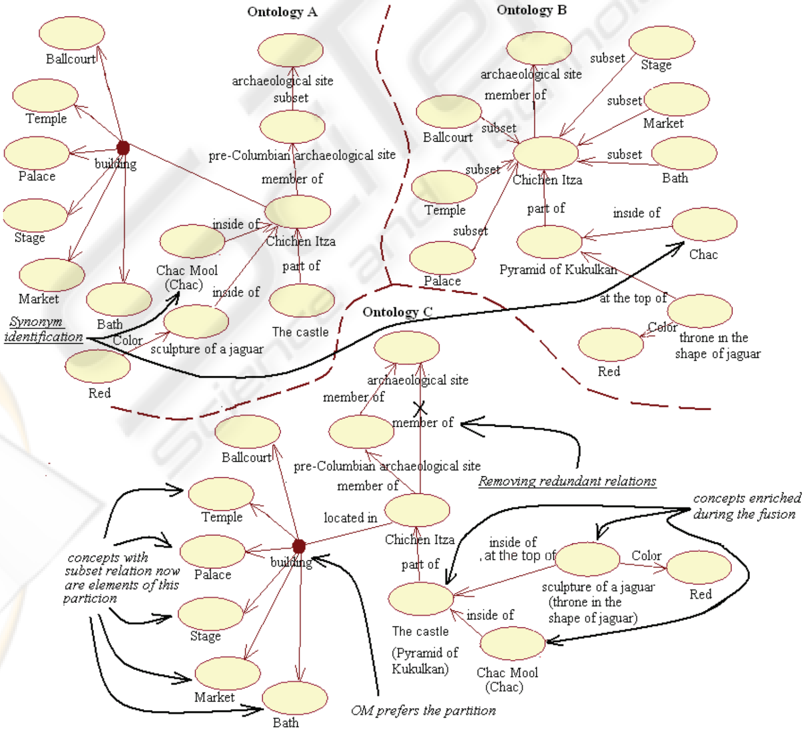

Figure 3: Ontology A and B describe `Chichen Itza`, where concepts `Chac Mool` in ontology A and `Chac` in B are identified (case **e**) as synonyms. A more interesting case is case **e**, that identifies `sculpture of a jaguar` in A as a similar concept (a synonym) to `throne in the shape of jaguar` in B. Also `The Castle` in A and `Pyramid of Kukulkan` in B are found to be the same. Case f removes redundant relations (marked with an X in the result C). Case g (see text) upgrades a set of subsets into a partition.

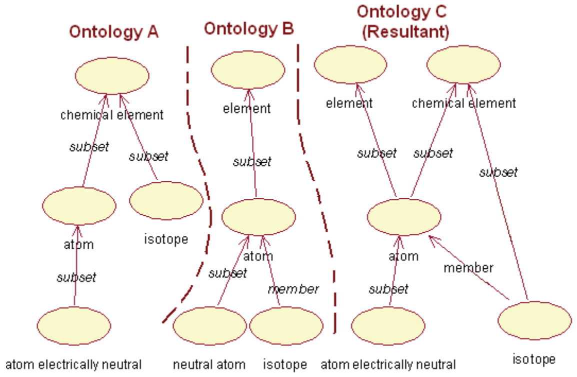

Figure 4: In ontology A concept isotope is a subset of Chemical element, but in B isotope is a member of atom; into C, the resultant ontology OM provides isotope with both ancestors Chemical element and atom.

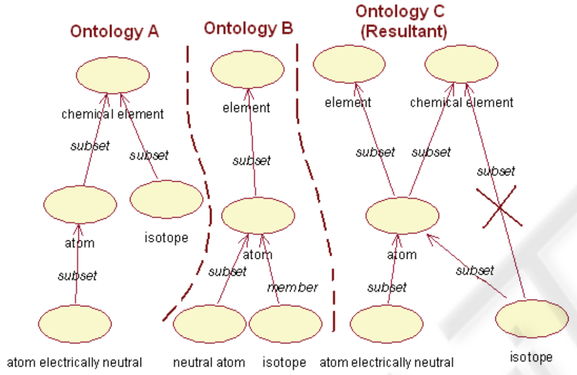

Figure 5: In B, isotope is subset of atom, whereas in A isotope is subset of chemical element. This last relation is redundant in C, and OM does not add it to C. OM fuses carefully, eliminating redundant relations.

3.3 More Applications of OM in Real Cases taken from the Web

OM has merged ontologies derived from real documents. The ontologies were obtained manually from several documents: 100 Years of Loneliness (URL 9 and 11], Oaxaca (URL 5 and 10), poppy (URL 2 and 4) and turtles (URL 6 and 7), describing the same thing. These ontologies were merged (automatically) by OM, and the product was manually validated, obtaining good results (Table 1).

Table 1: Performance of OM in some real examples: The columns "error" give the ratio of (number of wrong relations) / (total number of relations) and (number of wrong concepts) / (total number of concepts), respectively. More real examples in (Cuevas, 2006).

Ontologies	Error in the merging of relations	Error in the merging of concepts
Turtles	0	0
Hammer	0	0
Poppy	0	0
100 Years of Loneliness	2.7%	5.3%
Oaxaca	0	0.3%

4 DISCUSSION

Is it possible to keep fusing of several ontologies about the same topic, in order to have a larger ontology that faithfully represents and join the knowledge in each of the formant ontologies? OM say "yes, it is possible." What are the main roadblocks? As we perceive them, they are:

a. *Exploitation of hypergraphs*. Although we define ontologies as hypergraphs (section 2.1), the restrictions (r c_1 c_2 … c_k), where r is a relation, are lists, and consequently, order matters. For instance, it is not the same (kills; Cain; Abel; jaw of donkey) that (kills; Abel; Cain; jaw of donkey). However, the role of each "argument" or element of the restriction (such as jaw of donkey) must be explained –in the example it is the instrument used in the killing. Restrictions have different number of arguments, each one with different roles: consider (born; Abraham Lincoln; Kentucky; 1809; log cabin). We can expect a lot of arguments in a fragment of the text. The role of each argument must be explained or described in a transparent (not confuse) form –ideally, we suggest OM notation explain in (Cuevas, 2006)-, where OM can understand such explanations, *manipulate* them and *create* new ones. For instance, from a given argument, it should be able to take two different explanations (coming from ontologies A and B, respectively) and fuse them into a third explanation about such argument, to join into C. Ways to do all of this should be devised.

b. *A good parser*. Documents are now transformed by hand into ontologies, although fusion is totally automatic, but the work of verify the fusion is hard because it is also by hand. It has been found difficult to build a parser that reliably transforms a natural language document into its suitable ontology, due to the ambiguity of natural language and to the difficulty of representing relations (verbs, actions, processes) in a transparent way (see next point). Probably a good parser will profit from *the current knowledge that OM has stored in the ontology that was built before,* as well as in additional knowledge sources (point c below).

c. Additional language-dependent knowledge sources could effort enhance OM. For instance, WordNet, WordMenu, automatic discovery of ontologies by analyzing titles of conferences, university departments (Makagonov, 2007).

d. A query-answerer that queries a large ontology and makes deductions. (Botello, 2007) works on this for databases, not for ontologies. He has obtained no results for real data, yet.

In addition, some *caveats* are:

e. OM does not have a way to know *what is true* and what is false. All it does is to compute ontology C as the fusion of A and B, in a consistent form. If A and B say *the same lies,* these will go into C.

f. Probably the first ontologies should be carefully done by hand (even if parser existed), like, first documents (their ontologies, that is) to be fed to OM ("the first things OM will learn") have to be consistent, clear, and at a "low level."

g. The formal support behind OM and OM notation should be clearly adhered to.

5 CONCLUSIONS AND FUTURE WORK

This paper presents an automatic procedure (the OM algorithm) to fuse two ontologies about the same topic, which produces good results.. Thus, it is an important improvement to the computer-aided merging editors currently available (section 2.3). OM is an automatic, robust algorithm that fuses two ontologies into a third one, which preserves the knowledge obtained from the sources, solving some inconsistencies, detecting synonyms and homonyms, and expunging some redundant relations.

The examples shown, as well as others in (Cuevas, 2006; Cuevas & Guzman, 2007), provide evidence that OM does a good job, in spite of very general or very specific of joining ontologies. This is because the algorithm takes into account not only the words in the definition of each concept, but its semantics [context, synonyms, resemblance (through conf) to other concepts…] too. In addition, its base knowledge (some pre-built knowledge, such as synonyms, external language sources, stop words, words that change the meaning of a relation, among others) helps.

OM has not been tried on extensive, "real" ontologies (for instance, an ontology describing the complete work "100 Years of Loneliness"), due to the tedious work to hand-craft such ontology from the written document. Section 5.b addresses this.

REFERENCES

Botello, A. 2007. *Query resolution in heterogeneous data bases by partial integration*. Ph. D. Thesis in progress, Centro de Investigación en Computación (CIC), Instituto Politécnico Nacional (IPN), Mexico. In Spanish.

Cuevas, A. 2006 *Merging of ontologies using semantic properties.* Ph. D. thesis. CIC-IPN. In Spanish. http://148.204.20.100:8080/bibliodigital/ShowObject.jsp?idobject=34274&idrepositorio=2&type=recipiente

Cuevas, A. and Guzman, A. 2007. A language and algorithm for automatic merging of ontologies. Chapter of the book *Handbook of Ontologies for Business Interaction,* Peter Rittgen, ed. Idea Group Inc. Pp. 381-404.

Dou D., McDermott, D., and Qi. P. 2002. Ontology Translation by Ontology Merging and Automated Reasoning. *Proc. EKAW Workshop on Ontologies for Multi-Agent Systems*, Spain. Pp. 3-18.

Gómez P. A., and Suárez F. M. 2004. Evaluation of RDF(S) and DAML+OIL Import/Export Services within Ontology Platforms. MICAI 2004. Pp. 109-118.

Kalfoglou, Y., and Schorlemmer, M. Information-Flow-based Ontology Mapping. 2002. *Proceedings of the International Conference on Ontologies, Databases and Application of Semantics for Large Scale Information Systems*, volume 2519 of Lecture Notes in Computer Science. Pp. 1132-1151.

Kotis K., Vouros G. and Stergiou, K. 2006. Towards Automatic of Domain Ontologies: The HCONE-merge approach. *Journal of Web Semantics* (JWS), Elsevier, vol. 4:1, Pp. 60-79.

Lenat, D., and Guha, V. *Building Large Knowledge-Based Systems.* (Addison-Wesley 1989).

Levachkine, S., and Guzman, A. 2004 Hierarchy Measuring Qualitative Variables. *Lecture Notes in Computer Science* LNCS 2945. Computational Linguistics and Intelligent Text Processing, Springer-Verlag. Pp. 262-274. ISSN 0372-9743.

Makagonov, P. 2007. Automatic formation of ontologies by analysis of titles of conferences, sessions and articles.Work in preparation.

McGuinness, D., Fikes, R., Rice, J., and Wilder, S. 2000. The Chimaera Ontology Environment. *Proceedings of the Seventeenth National Conference on Artificial*

Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence. ISBN: 0-262-51112-6. Pp. 1123-1124.

Noy, N., and Musen, M. 2000. PROMPT: Algoritm and Tool for Automated Ontology Merging and Alignment. In *Proc. Of the National Conference on Artificial Intelligence*. Pp. 450-455, Austin, TX, USA.

Olivares, J. 2002. An *Interaction Model among Purposeful Agents, Mixed Ontologies and Unexpected Events*. Ph. D. Thesis, CIC-IPN. México. www.jesusolivares.com/interaction/publica.

Stumme, G., and Maedche, A. 2002. Ontology Merging for Federated ontologies on the semantic web. In: E. Franconi, K. Barker, D. Calvanese (Eds.): *Proc. Intl. Workshop on Foundations of Models for Information Integration*, Viterbo, Italy. LNAI, Springer 2002.

URLs:

1. Accessed 5 march 2008. AKT: plainmoor.open.ac.uk/ocml/domains/aktive-portal-ontology/techs.html
2. Accessed 5 march 2008. es.wikipedia.org/wiki/Amapola
3. Accessed 5 march 2008. es.wikipedia.org/wiki/Miguel_%C3%81ngel
4. Accessed 5 march 2008. www.buscajalisco.com/bj/salud/herbolaria.php?id=1
5. Accessed 5 march 2008. www.elbalero.gob.mx/explora/html/oaxaca/geografia.html
6. Accessed 5 march 2008. www.damisela.com/zoo/rep/tortugas/index.htm
7. Accessed 5 march 2008. www.foyel.com/cartillas/37/tortugas_-_accesorios_para_acuarios_i.html
8. Accessed 5 march 2008. www.historiadelartemgm.com.ar/biografiamichelangelobuonarroti.htm
9. Accessed 5 march 2008. www. monografias.com/trabajos10/ciso/ciso.shtml
10. Accessed 5 march 2008. www.oaxaca-mio.com/atrac_turisticos/infooaxaca.htm
11. Accessed 5 march 2008. www.rincondelvago.com/cien-anos-de-soledad_gabriel-garcia- marquez_22.html
12. Accessed 5 march 2008. es.wikipedia.org/wiki/Miguel_de_Cervantes
13. Accessed 5 march 2008. www.biografiasyvidas.com/monografia/cervantes/
14. Accessed 5 march 2008. http://www.w3.org/TR/2004/REC-owl-ref-20040210/ WWW page
15. Accessed 6 march 2008. http://www.w3.org/TR/2001/NOTE-daml+oil-reference-20011218. WWW page
16. Accessed 6 march 2008. http://www.w3.org/TR/2004/REC-rdf-primer-20040210/. WWW page.