Concept Mapping: Connecting Educators
Proc. of the Third Int. Conference on Concept Mapping
Tallinn, Estonia & Helsinki, Finland 2008

A SUPPORT TO FORMALIZE A CONCEPTUALIZATION FROM A CONCEPT MAP REPOSITORY

Alfredo Simón-Cuevas[1], Luigi Ceccaroni[2], Alejandro Rosete-Suárez[1], Amhed Suárez-Rodríguez[1], Rodolfo Victoria-Oquendo[3]
[1]Centro de Estudios de Ingeniería de Sistemas (CEIS), Facultad de Ingeniería Informática, Instituto Superior Politécnico "José Antonio Echeverría", C. Habana, Cuba
[2]Departament de Llenguatges i Sistemes Informàtics (LSI), Universitat Politècnica de Catalunya (UPC), Campus Nord, Edif. Omega, C. Jordi Girona, 1-3, 08034 Barcelona, Spain
[3]Oficina Central División Sol Meliá Cuba, C. Habana, Cuba
Email: {asimon, rosete, asuarez}@ceis.cujae.edu.cu, luigi@lsi.upc.edu, rodolfo.victoria@solmeliacuba.com

Abstract. Operations focused on the integration and filtering of concepts and propositions from various concept maps are presented as a Concept Maps Query Language (CMQL), representing a novel approach to automatically obtain knowledge from a concept map repository. Additionally, the mapping between concept maps and ontologies is described as a formal transformation method, which semantically analyzes the relationships among concepts in the map. A context in which CMQL can be useful is as support in the construction of the concept map (conceptualization) to be formalized in a preliminary ontology.

1 Introduction

In most scientific domains, information needs sometimes to be analyzed and processed by machines. In the knowledge representation oriented to the semantic analysis and processing by machines, context in which a certain degree of formalization is required, the development and use of *ontologies* is increasingly common. In this paper, we adopt the following definition for *ontology*: a formal and explicit specification of a conceptualization, which is readable by a computer; which is derived from Gruber (1993), Borst (1997) and Studer et al. (1998). *Concept maps* (CMs) are human-friendly, graphically-rich tools for organizing and representing knowledge (Novak & Gowin, 1984), and several works suggest that ontologies and CMs can be integrated (Gómez, Díaz, & González, 2004; Hayes, Eskridge, Saavedra, Reichherzer, Mehrotra, & Bobrovnikoff, 2005; Brilhante, Macedo, & Macedo, 2006).

Concept mapping is a knowledge elicitation technique that consists of enumerating a list of concepts and determining the *linking-phrases* that should connect the concepts to form meaningful propositions. This process can be carried out semi-automatically, when the system retrieves information and suggests concepts, relationships between concepts, or propositions to a human (Reichherzer, Cañas, Ford, & Hayes, 1998; Cañas, Hill, Carff, Suri, Lott, Gómez, Eskridge, Arroyo, & Carvajal, 2004; Richardson, Goertzel, & Fox, 2006), or automatically, when the CM is constructed without the aid of a human. In any case, the use of some information source, such as texts (Richardson, Goertzel & Fox, 2006), the Web or CM repositories (Cañas, Hill, Carff, Suri, Lott, Gómez, Eskridge, Arroyo, & Carvajal, 2004), is required. In case a CM repository is used as information source, the set of query operations that can be used for information retrieval has not been formally defined in the literature. Eskridge et al. (2006) argue that CMs are very good at organizing knowledge about a wide variety of subjects; however, they present some difficulties when it comes to the retrieval of concept maps based on individual query terms. Moon et al. (2006) report how to integrate many CMs to create a new CM that they call "master map", doing it in part manually, and in part using CmapTools (Cañas, Hill, Carff, Suri, Lott, Gómez, Eskridge, Arroyo, & Carvajal, 2004). From a study of the current state of the art, the authors believe that efforts should be dedicated to extend the scope of knowledge management in relation to CM repositories.

In this paper, we propose an extension of the process reported by Simón et al. (2008) to obtain a formal preliminary ontology from a CM. Specifically, a CM query language (CMQL) to be applied to a CM repository is proposed in order to support the user in obtaining a *conceptualization* (e.g., a CM to be transformed into a preliminary ontology). The CMQL defined here can be useful to the *knowledge engineering process* that is carried out for the creation of ontologies from CMs, in which obtaining *potentially useful knowledge* (coming from existing knowledge) for the construction of the conceptualization by the user or the knowledge worker is a required task. The CMQL allows users and knowledge workers to know about *concepts* and *propositions* that have been shared by other users in a CM repository, through several operations focused on the integration and filtering of *concepts* and *propositions*. The result of these processes is represented in a new CM.

The paper begins with an overview of ontologies and the languages defined to formalize them (section 2). In section 3, the process of obtaining a conceptualization from a set of CMs is studied and a query language to achieve this is formally presented; an example is included for better understanding. Section 4 describes the basic aspects of the CM-ontology mapping according to the formalization method reported by Simón et al. (2008); an example, in which the portion of the CM formalized is obtained from a query operation of CMQL, is again included for better understanding. Finally, the comparison with the state of the art is reported in section 5.

2 Ontologies and their languages

In artificial intelligence, ontologies were introduced to share and reuse knowledge. They provide the common reference frame for communication languages in distributed environments (such as multi-agent systems or the semantic Web) and a formal description for automatic knowledge processing. Several languages have been defined to implement them; OWL (Smith, Welty, & McGuinness, 2004) is the latest, standardized ontology language. OWL is based on RDF and RDFS, and includes three specifications, with different expressiveness levels: OWL Lite, OWL DL and OWL Full. The code obtained by the method described in this paper is generated according to OWL DL specifications. OWL DL is so named due to its correspondence with *description logics*. *Description logic* (DL) is the name for a family of knowledge representation formalisms that represent the knowledge of a domain by first defining the relevant concepts of the domain (its terminology), and then using these concepts to specify properties of objects and individuals occurring in the domain (Baader & Nutt, 2003). The terminology specifies the vocabulary of a domain, which consists of concepts and roles, where the concepts denote individuals while roles denote binary relationships between individuals.

3 Obtaining knowledge from concept maps

In this section, the operations for obtaining knowledge from a CM repository are studied and a formal query language (CMQL) to achieve this is presented. The result of the application of each operation is the automatic construction of a new CM. The system retrieves information (*concepts* and *propositions*) from a repository of CMs, through the following operations:

1. union of a CM set;
2. intersection of a CM set;
3. closed sub-map, guided by a concept set;
4. open sub-map, guided by a concept set;
5. open sub-map of radio R, guided by a concept set;
6. closed extension of a CM, guided by another CM and a concept set;
7. open extension of a CM, guided by another CM and a concept set;
8. open extension of radio R of a CM, guided by a concept set.

For the information retrieval in the repository, the system uses one or several of the query operations included in CMQL. The information that is retrieved from the repository is formed by *concepts* and *propositions*, and is expressed as the automatic construction of a new CM, in which those *concepts* and *propositions* are related. The query operations defined in CMQL are formalized in terms of the combination of graph theory and set theory, and may have as input one or more CMs (and a concept set in some cases) (as shown in Table 1). The CM is represented as a *directed graph* (Johnsonbaugh, 1999), that is, $G = (V, E)$, where V is the set of vertices (*concepts*) and E the set of directed edge (*propositions*). This allows taking advantage of the operations that have been defined in both fields (graph theory and set theory) for the automatic processing of CMs.

<table>
<tr><td colspan="3">Basic definitions:
M^x is a concept map, $M^x = (C^x, P^x)$; <i>c</i> is a concept;
C^x and CS are concept sets;
P^x is a proposition set, $P^x = \{\ldots, (c_o, l\text{-}p_j, c_d), \ldots\} \mid l\text{-}p$ is a <i>linking-phrase</i> and $c_o, c_d \in C^x$;
CMS is a set of concept maps.</td></tr>
<tr><td>Query operations</td><td>Expression</td><td>Results</td></tr>
<tr><td>Union of a CM set</td><td>$UM(CMS) = \cup M^i \mid M^i \in CMS$</td><td>A new CM formed by: all <i>concepts</i> and <i>propositions</i> represented in the CMs included in CMS.</td></tr>
<tr><td>Intersection of a CM set</td><td>$IM(CMS) = \cap M^i \mid M^i \in CMS$</td><td>A new CM formed by: the <i>concepts</i> presented in all CMs in CMS, and the <i>propositions</i> in which they are related.</td></tr>
<tr><td>Closed sub-map, guided by a concept set</td><td>$SM^-(M^x, CS) = M^{x \cap y} \mid M^y = (CS, \{\})$</td><td>A new CM formed by:
• the common <i>concepts</i> between CS and M^x; and
• the <i>propositions</i> in M^x in which they are related.</td></tr>
</table>

Open sub-map, guided by a concept set	$SM^{+}(M^{x}, CS) = M^{1} = (C^{1}, P^{1}) \mid P^{1} = \{(c_o, \mathit{l\text{-}p}_{j}, c_d) \mid (c_o, \mathit{l\text{-}p}_{j}, c_d) \in P^{x}, c_o \in (CS \cup C^{x}) \vee c_d \in (CS \cup C^{x})\}, C^{1} = \{c_i \mid (c_i, \mathit{l\text{-}p}_{j}, c_d) \in P^{1}\} \cup \{c_i \mid (c_o, \mathit{l\text{-}p}_{j}, c_i) \in P^{1}\}$	A new CM formed by: • the *concepts* in CS and their neighbors in M^{x} (Two concepts are neighbors if they are related by a proposition.); • the *propositions* in M^{x} in which the previous concepts are related.
Open sub-map of radio R, guided by a concept set	$SM^{+,R}(M^{x}, CC) = \begin{cases} SM^{+}(M^{x}, CC) & \text{si } R = 1 \\ SM^{+,R-1}(M^{x},C1), M1= (C1,P1) = SM^{+}(M^{x},CC) & \text{si } R>1 \end{cases}$	A new CM formed by: • the common *concepts* between CS and M^{x} and all *concepts* in M^{x} to which a path with length ≤ R can be created from some concept in CS; • the *propositions* in M^{x} in which those concepts are related.
Closed extension of a CM, guided by another CM and a concept set	$Ext^{-}(M^{x}, M^{y}, CS)$	A new CM formed by: • the *concepts* in M^{x} and the *concepts* included in the CM obtained from $SM^{-}(M^{y}, CS)$; • the *propositions* in M^{x} and the *propositions* included in the CM obtained from $SM^{-}(M^{y}, CS)$.
Open extension of a CM, guided by another CM and a concept set	$Ext^{+}(M^{x}, M^{y}, CS)$	A new CM formed by: • the *concepts* in M^{x} and the *concepts* included in the CM obtained from $SM^{+}(M^{y}, CS)$; • the *propositions* in M^{x} and the *propositions* included in the CM obtained from $SM^{+}(M^{y}, CS)$.
Open extension of radio R of a CM, guided by another CM and a concept set	$Ext^{+,R}(M^{x}, M^{y}, CS)$	A new CM formed by: • the *concepts* in M^{x} and the *concepts* included in the CM obtained from $SM^{+,R}(M^{y}, CS)$; • the *propositions* in M^{x} and the *propositions* included in the CM obtained from $SM^{+,R}(M^{y}, CS)$.

Table 1: CM query operations included in CMQL

The CM query operations allow automatically obtaining a new CM, which can be edited later by the user, from knowledge represented in other CMs. This is a novel contribution with respect to current retrieval proposals, in which *concepts* and *propositions* are retrieved independently and have to be integrated by the user (Cañas, Hill, Carff, Suri, Lott, Gómez, Eskridge, Arroyo, & Carvajal, 2004). With the proposed method, CMs developed by persons focused on different aspects of a domain can be integrated, as in the case of "master maps" (Moon, Pino, & Hedberg, 2006), which can be automatically obtained using the operation *Union of a CM set*.

As an example, six query operations (intersection, union, open sub-map, closed sub-map, open extension and open extension of radio R) are applied to the simple CMs shown in Figure 1 and 2. Results are shown in Figures 3-8.

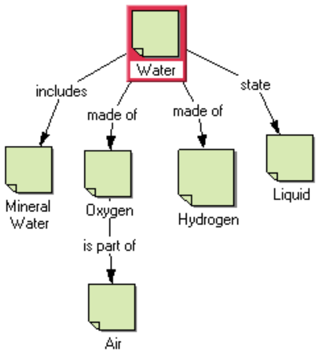

Figure 1. CM about "Water"

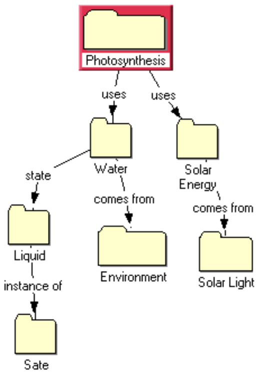

Figure 2. CM about "Photosynthesis"

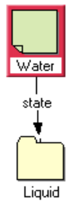

Figure 3. Intersection of Water’s CM and Photosynthesis’s CM:

IM({Water, Photosynthesis})

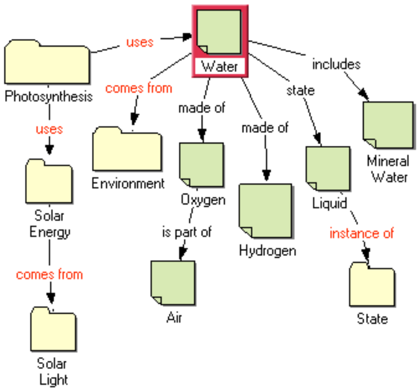

Figure 4. Union of Water’s CM and Photosynthesis’s CM:

UM({Water, Photosynthesis})

Figure 5. Open sub-map of Water’s CM, guided by Oxygen and Air concepts:

SM^{+}(Water,{Oxygen, Air})

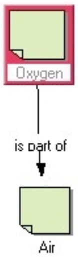

Figure 6. Closed sub-map of Water’s CM, guided by Oxygen and Air concepts:

SM^{-}(Water,{Oxygen, Air})

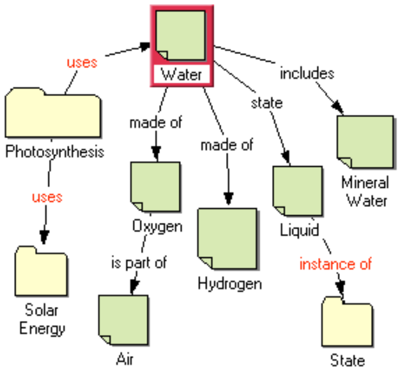

Figure 7. Open extension of Water’s CM, guided by Photosynthesis’s CM and by Photosynthesis and State concepts:

Ext^{+}(Water, Photosynthesis, {Photosynthesis, State})

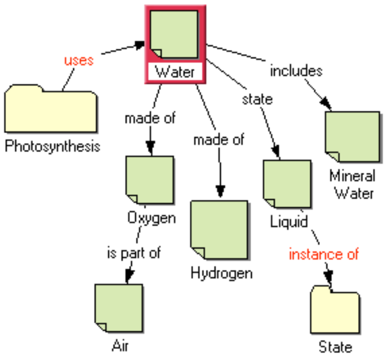

Figure 8. Open extension of R = 1 of Water’s CM, guided by Photosynthesis’s CM and by Photosynthesis and State concepts:

Ext$^{+,1}$(Water, Photosynthesis, {Photosynthesis, State})

CMQL allows the user to know about *concepts* and *propositions* that have been shared by other users in the CM repository and provides the capability for the analysis of the interrelations of existing knowledge. The availability of this type of operations may be used as an indicator of the potential inference capability of a given CM tool kit. The CMQL defined can be useful to obtain a *conceptualization* from a CM repository, which may be later translated into OWL in order to formalize the informal knowledge of a CM into an ontology.

4 Basic aspects of CM-OWL mapping

Knowledge, in OWL ontologies, is expressed as *classes*, *subclasses*, *properties* and *instances* (Smith, Welty, & McGuinness, 2004), while in CMs much of this formal and explicit specification does not exist, and has to be inferred. Nonetheless, some initial structural mapping between CMs and OWL can be easily established:

- *Concepts* correspond to: *classes* and *instances*;
- *linking-phrases* correspond to: *properties*, considering this as a binary relation between instances of classes in OWL (Smith, Welty, & McGuinness, 2004);
- *propositions* correspond to *classes* and *properties’ restrictions* or other OWL constructs.

Some type of semantic relation, such as *class-subclass*, *class-property*, *class-property-value*, *class-instance*, can be inferred from certain *linking-phrases* used in CMs, in accordance with others authors (Hayes, Eskridge, Saavedra, Reichherzer, Mehrotra, & Bobrovnikoff, 2005; Brilhante, Macedo, & Macedo, 2006). In this section, how to map a CM (synthesized from a CM repository with the use of CMQL) to an ontology will be described.

In addition to the CM to be formalized, two external knowledge sources are used in this work: WordNet (Miller, Beckwidth, Fellbaum, Gross, & Miller, 1990) and another CM repository. WordNet is a lexical knowledge base, whose basic structure is the *synset. Synsets* form a semantic network and are interconnected among themselves by several types of relations, some of which are used in the proposed algorithm, such as *hypernymy-hyponymy* (class/subclass) and *meronymy-holonymy* (part/whole). The *synset* defines the meaning of a word, which, in the case of *polysemy*, can be found in various *synsets*. WordNet can be used as an ontology if its links are associated to a formal semantics. The CM repository used here is the included in ServiMap (CMs Server) (Simón, Estrada, Rosete, & Lara, 2006), which stores several CM of different domains constructed using the Macosoft CM editor (Simón, Estrada, Rosete, & Lara, 2006).

The mapping and semantic inference leading to OWL coding are carried out combining (Simón, Ceccaroni, & Rosete, 2008) the analysis of:

- the syntax of the *propositions;*
- the occurrence of similar semantic relations in WordNet and the external CM repository.

Initially, some frequently used *linking-phrases* are defined and organized in four *categories*, according to the semantics that can be associated to them and their correspondence with the semantic relations in WordNet. They are:

- *Classification (CC)*, for *linking-phrases* that may indicate *(super class - class)* relations between concepts in a *proposition* (e.g., *includes*) in the proposition *(Water, includes, Mineral Water)* in Figure 1); corresponding to *hypernymy* and *hyponymy* relations in WordNet;
- *Instance (IC)*, for *linking-phrases* that may indicate *(class-instance)* relations between concepts in a *proposition* (e.g., *instance of* in the proposition *(Liquid, instance of, State)* in Figure 2);
- *Property (PC)*, for *linking-phrases* that may indicate *(class-property)* relations between concepts in a *proposition* (e.g., *has*); corresponding to *has_meronym* and *has_holonym* relations in WordNet;
- *Property-Value (PVC)* for *linking-phrases* that may indicate *(class-property-value)* relations between concepts in a *proposition*, such as nouns (e.g., *state* in the proposition *(Water, state, Liquid)* in Figure 1); corresponding to basic *meronymy* and *holonymy* WordNet's relations, and different from the more specific *has_meronym* and *has_holonym* relations (e.g. *has_mero_madeOf* and *has_holo_madeOf*).

This method allows everyday natural language to be used at CM construction time. Lexemes are used to avoid duplications due to verb forms' variability. The *linking-phrases* are continually and automatically enriched: if the proposition's semantics is inferred by some semantic relations from WordNet, then the *linking-phrase* used in this proposition is added to the corresponding category.

In the mapping method, the CM under consideration is analyzed as a structured text. A concept sense-disambiguation algorithm (Simón, Ceccaroni, & Rosete, 2007) with 89.9 % accuracy, is used to infer the most rational sense (in terms of WordNet's *synsets*), for the concepts in the CM. Once inferred a *synset* for each concept in a *proposition*, the semantics of the CM relation among them can be inferred, through a similar semantic relation between the *synset* of these *concepts* represented in WordNet (if one exists).

The method for obtaining OWL-DL ontologies form CM is organized in three phases (Simón, Ceccaroni, & Rosete, 2008): *preprocess*, *mapping* and *codification* and four components are defined for the implementation of the system:

- *parser*: it analyzes the CM to be translated to OWL, identifying propositions and their parts (*concepts* and *linking-words*) and obtains knowledge related to the CM from a CM repository;
- *disambiguator*: it infers the most rational *sense* (in terms of WordNet *synsets*) of the concepts in the CM, using the algorithm defined by Simón et al. (2007), and identifies the semantic relations between these *synsets* in WordNet;
- *semantic interpreter*: it applies a set of heuristic rules on the propositions obtained by the *parser*, to infer several semantic descriptions from the CM, such as: *classes*, *relations between classes*, *relations between classes and instances*, *property restriction (by value)*, *object properties, symmetric and functional properties*, *union classes* and *intersection classes*.

- *OWL codifier*: it uses the semantic description inferred by the *semantic interpreter* and writes out the corresponding OWL constructs according to W3C recommendations (Smith, Welty, & McGuinness, 2004).

As an example, we use the CM shown in Figure 7, which was obtained as result of the query operation *Open extension* on the *Water* CM (shown in Figure 1). This is a case in which a user needs to obtain an ontology about *water*; he has some knowledge about it, which allows him to construct the CM shown in Figure 1, but his needs require more information. In this situation, the query operation *Open extension* is used to enrich the *Water* CM with related knowledge that has been shared by other authors in a CM repository, which has only one CM (the *Photosynthesis* CM) in this case. A new CM (shown in the Figure 7), extending the *Water* CM, is automatically constructed. Finally, the *OWL codifier* is used to obtain an ontology from the CM:

```
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns="http://www.owl-ontologies.com/unnamed.owl#"
xml:base="http://www.owl-ontologies.com/unnamed.owl">
<owl:Ontology rdf:about="Water"/>
<owl:Class rdf:ID="Hydrogen"/>
<owl:Class rdf:ID="Oxygen"/>
<owl:Class rdf:ID="State"/>
<owl:Class rdf:ID="Air"/>
<owl:Class rdf:ID="Photosynthesis"/>
<owl:Class rdf:ID="Solar Energy"/>
<State rdf:ID="Liquid"/>
<owl:Class rdf:ID="Water">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#state"/>
      <owl:hasValue rdf:resource="#Liquid">
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#made of"/>
      <owl:someValuesFrom rdf:resource="#Hydrogen"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#made of"/>
      <owl:someValuesFrom rdf:resource="#Oxygen"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Mineral Water"/>
   <rdfs:subClassOf rdf:resource="#Water"/>
</owl:Class>
<owl:ObjectProperty rdf:about = "#hasPart">
   <rdfs:domain rdf:resource="#Air"/>
   <rdfs:range rdf:resource="#Oxygen"/>
</owl:ObjectProperty>
```

5 Related work

Gómez et al. (2004) report a transformation mechanism from CM into OWL language, to apply in the case-based reasoning context. In that mechanism, the CM is constructed by a user and is coded in XTM (XML Topic Maps) (Biezunski, Newcomb, & Bryan, 2002): *concepts* and *linking-phrases* are represented by *topic* tags and the *propositions* are represented by *association* tags (using the label of *linking-phrases* in the proposition as identified), and a set of rules for obtaining OWL are applied to it. In the OWL coding process, *topics* corresponding to *concepts* are coded as *owl:class* and the ones corresponding to *linking-phrases* are coded as *owl:ObjectProperty*, and *association* tags (corresponding to *proposition*) are coded as *property restrictions: by value*. XTM is a language lacking explicit semantics; therefore the direct mapping from XTM to OWL is very limited without a previous semantic analysis of the relations in the CM. COE is a collaborative ontology environment (Hayes, Eskridge, Saavedra, Reichherzer, Mehrotra, & Bobrovnikoff, 2005), which includes a mechanism of visualization-generation of OWL ontologies based on CM. Several graphical conventions (*templates*) are used to specify the semantics of *concepts* and *propositions* in the concept mapping. A set of *linking-phrases* is predefined to represent types of relations between concepts, e.g., "are" and "is a" (to represent class relations); "at most" and "at least" (to represent cardinality restrictions) and "cannot be" (to define negation). These aspects are oriented to increase the formalization of the CM by restricting the natural language to be used in the concept mapping. COE can show concepts from existing ontologies that are relevant to COE user's current focus; it search through some existing ontologies to locate potentially useful, contextually relevant concepts, to aid the user's comprehension of existing ontologies and lead to "fortuitous" reuse opportunities (Hayes, Eskridge, Saavedra, Reichherzer, Mehrotra, & Bobrovnikoff, 2005). Brilhante et al. (2006) report a method to translate an individual CM into an ontology. The translation is carried out by employing several heuristic rules designed to establish the representational correspondences between CMs features and OWL constructs. Concepts and relations in CMs are mapped into *classes*, *object properties*, *property restrictions* and *individuals*. The process is based on a set of predefined *linking-phrase*, e.g., "has a", "has part", "is part of" (to identify composition relations) and "is a", "can be" (for the identification of subclasses and superclasses), and on the use of *hypernymy* and *meronymy* relations from WordNet. In this approach the authors do not consider the concept's ambiguities in the use of WordNet; therefore the efficiency and effectiveness in the use of it can be low in several cases.

6 Conclusions

To realize a semantic Web, tools are required that allow users with little technical background to generate their own ontologies and collaborate in the construction of distributed knowledge bases. The work presented here is an extension of a method to formally obtain ontologies codified in the OWL-DL language from an informal knowledge representation, such as concept maps. In this paper, operations focused on the integration and filtering of concepts and propositions from various concept maps are presented as a query language (CMQL), representing a novel approach to obtain knowledge from a concept map repository. The query language presented allows the user to know about concepts and propositions that have been shared by other users in the repository, in order to support the user in obtaining a *conceptualization*. At the current state of development, the application of the operations presented may generate meaningless and contradictory propositions in the new maps constructed, for consistency and sense checking is not yet implemented; we are considering this aspect as future work. The formalization method presented advances the state of the art through the use of CMQL, as an alternative to automatically obtain potentially useful knowledge from a CM repository. The use of tools and techniques from natural language processing, such as the use of WordNet and a concept sense disambiguation algorithm, are other contributions of the formalization method presented. This, combined with the topological analysis of concept maps, allows maintaining a greater flexibility and more independence during concept mapping, which is an important aspect for less-expert users in ontology construction. Additionally, the increase in the formalizations level of the *linking-phrases* and the use of external knowledge representations, allow to augment the semantic description inference in concept maps and to obtain more expressiveness in the resulting OWL than the ones reported by other authors.

7 Acknowledgements

The research described is partly supported by the EC project Laboranova (IST-5-035262-IP). The opinions expressed in this paper are those of the authors and are not necessarily those of Laboranova partners.

References

Baader, F., & Nutt, W. (2003). Basic Description Logics, The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, 47-100.

Biezunski, M., Newcomb, S., & Bryan, M. (2002). Guide to the topic map standards. ISO/IEC 13250 Projects.

Borst, P. (1997). Construction of Engineering Ontologies for Knowledge Sharing and Reuse. Ph. D. thesis, University of Twente, The Netherlands.

Brilhante, V., Macedo, G., & Macedo, S. (2006). Heuristic Transformation of Well-Constructed Conceptual Maps into OWL Preliminary Domain Ontologies, Workshop on Ontologies and their Applications, WONTO’2006, Brazil.

Cañas, A. J., Hill, G., Carff, R, Suri, N., Lott, J., Gómez, G., Eskridge, T. C., Arroyo, M., & Carvajal, R. (2004) CmapTools: A Knowledge Modeling and Sharing Environment. In A. J. Cañas, J. D. Novak, F. M. González (Eds.), Concept Maps: Theory, Methodology, Technology, Proceedings of the First International Conference on Concept Mapping (Vol. 1, pp. 125-133). Pamplona, Spain: Universidad Pública de Navarra.

Eskridge, T. C., Granados, A., & Cañas, A. J. (2006). Ranking Concept Map Retrieval in The CmapTools Network. In A. J. Cañas & J. D. Novak (Eds.), Concept Maps: Theory, Methodology, Technology. Proceedings of the Second International Conference on Concept Mapping (Vol. 1, pp. 477-484). San José, Costa Rica: Universidad de Costa Rica.

Gómez, H., Díaz, B., & González, A. (2004). Two layered approach to knowledge representation using conceptual maps description logic. In A. J. Cañas, J. D. Novak & F. M. González (Eds.), Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping (Vol.1, pp. 205-212). Pamplona, Spain: Universidad Pública de Navarra.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition, 5(2), 199-220.

Hayes, P. J., Eskridge, T. C., Saavedra, R., Reichherzer, T., Mehrotra, M., & Bobrovnikoff, D. (2005). Collaborative knowledge capture in ontologies. In Proceedings of the 3rd International Conference on Knowledge Capture (K-CAP 2005) (pp. 99-106). Canada.

Johnsonbaugh, R. (Ed.) (1999). Matemáticas Discretas. Fourth Edition, Prentice Hall Hispanoamericana. S. A.

Miller G.A, Beckwidth, R., Fellbaum, C., Gross, D., & Miller, K. J. (1990). Introduction to WordNet: An On-line Lexical Database, International Journal of Lexicography, 3(4), 235-244.

Moon, B. M., Pino, A. J., & Hedberg, C. A. (2006). Studying Transformation: The Use of CmapTools in Surveying the Integration of Intelligence and Operations. In A. J. Cañas & J. D. Novak (Eds.), Concept Maps: Theory, Methodology, Technology. Proceedings of the Second International Conference on Concept Mapping (Vol. 1, pp. 527-533). San José, Costa Rica: Universidad de Costa Rica.

Novak, J. D., & Gowin, D. B. (1984). Learning How to Learn. New York: Cambridge University Press.

Reichherzer, T. R., Cañas, A. J., Ford, K. M., & Hayes, P. J. (1998). The Giant: An Agent-based Approach to Knowledge Construction & Sharing. American Association for Artificial Intelligence.

Richardson, R., Goertzel, B., & Fox, E. A. (2006). Automatic Creation and Translation of Concept Maps for Computer Science-Related Theses and Dissertation. In A. J. Cañas, & J. D. Novak (Eds.), Concept Maps: Theory, Methodology, Technology. Proceedings of the Second International Conference on Concept Mapping (Vol. 2, pp. 32-35). San José, Costa Rica: Universidad de Costa Rica.

Simón, A., Estrada, V., Rosete, A., & Lara, V. (2006). GECOSOFT: Un Entrono Colaborativo para la Gestión del Conocimiento con Mapas Conceptuales. In A. J. Cañas & J. D. Novak (Eds.), Concept Maps: Theory, Methodology, Technology. Proceedings of the Second International Conference on Concept Mapping (Vol. 2, pp. 114-117). San José, Costa Rica: Universidad de Costa Rica.

Simón, A., Ceccaroni, L., & Rosete, A. (2007). Concept sense disambiguation in concept maps using WordNet, Technical Report LSI-07-44-R, Departament de Llenguatges i Sistemes Informàtics (LSI), Universitat Politècnica de Catalunya (UPC), Campus Nord, Edif. Omega, c. Jordi Girona, 1-3, 08034 Barcelona, Spain.

Simón-Cuevas, A., Ceccaroni, L., & Rosete-Suárez, A. (2008). An approach to formal modeling of environmental knowledge via concept maps and ontologies, In M. Sànchez-Marrè, J. Béjar, J. Comas, A. Rizzoli & G. Guariso (Eds.), Proceedings of International Congress on Environmental Modeling and Software 2008 (iEMSs’08), Barcelona, Spain.

Smith, M., Welty, Ch., & McGuinness, D. (2004). OWL Web Ontology Language Guide, Official site of W3C.

Studer, R., Benjamins, V. R., & Fensel D. (1998). Knowledge engineering: principles and methods. Data and Knowledge Engineering, 25(1-2), 161–197.