Toward a Cooperatively Built Ontology of Knowledge Engineering

Philippe MARTIN
Griffith University
School of ICT
PMB 50 Gold Coast MC, QLD 9726
AUSTRALIA

http://www.phmartin.info

Michel EBOUEYA
University of La Rochelle
Laboratoire Informatique Image et Interaction
Av. Michel Crépeau, 17042 La Rochelle Cedex 1
FRANCE

http://univ-lr.fr/labo/l3i/

Abstract: - Nowadays, it is difficult and inefficient to publish, retrieve, compare, evaluate and learn ideas and techniques about knowledge engineering since they are not organized into a semantic network but stored within informal documents and hence scattered and described in various ways across millions of such documents (research articles, documentations, emails, etc.). Our knowledge server WebKB-2 supports the collaborative building of a formal or semi-formal semantic network. We have begun creating such a network to permit a scalable sharing of information about knowledge engineering. This article illustrates and discusses this work.

Key-Words: Knowledge engineering, Knowledge sharing, Knowledge retrieval, Ontology, CSCW

1 Introduction

Nowadays, as in any other domain, publishing information about knowledge engineering (KE) most often involves writing sentences in a document. This is a lengthy process which implies summarizing or describing ideas or facts that have already been summarized or described by countless other persons and also implies making rather arbitrary choices and compromises about which information to describe, at which level of detail, in which order, etc. Furthermore, the result of this exercise only adds to the volume of poorly structured and heavily redundant data that she and other persons later have to sift through to find information.

The problem is that information about KE is currently not structured into a *semantic network of techniques or ideas* that a Web user could (i) navigate to get a synthetic view of a subject or, as in a decision tree, quickly find its path to relevant information, and (ii) easily update to publish a new idea (or the explanation of an idea at a new level of detail) and link it to other ideas via semantic relations. Various small steps toward that goal can be observed.

The most well known is that Wikipedia has a page about KE and many pages about KE related objects. However, using Wikipedia (in connection with other wikis since the content of Wikipedia is meant to remain of "encyclopaedic" nature, that is, not too technical) is not a scalable approach. Indeed, current wikis, even semantic wikis such as Semantic MediaWiki, do not provide minimal supports for the collaborative building of a large well organized semantic network: no initial large lexical ontology, no intuitive expressive notation, no structural and ontological guidelines, no editing/sharing protocols, and extremely limited knowledge checking, querying and browsing features. Thus, current semantic wikis remain mostly informal and poorly structured. For example, the knowledge representation language (KRL) of Semantic MediaWiki does not permit to express quantifiers, collections, meta-information (even to represent the author of a statement, a kind of information that is essential to support editing/sharing protocols and filtering mechanisms) and it only permits to represent relations within hyperlinks and with source the object of the page (hence, for example, to represent the semantic content of a table, a user would have to create as many pages as there are columns or rows in the table).

The same restricted approach (and similar KRL within hyperlinks) was used in the well-publicized KA2 project [1] which re-used Ontobroker and aimed to let Knowledge Acquisition (KA) researchers index their KA resources within their Web pages. (The pages of the registered researchers were loaded from time to time into Ontobroker and the various bits of knowledge were then aggregated when possible). Furthermore, the provided ontology was extremely small (only 37 domain names) and could not be directly updated by users. Thus, this approach was extremely limiting, was not followed

by many KA researchers, and could not support the representation or indexation of research ideas.

Finally, Fact Guru (the commercial successor of CODE4 [11]), a knowledge base (KB) server with a semi-formal English-like syntax supporting minimal knowledge processing, once proposed users to access and complement a small KB on Object-Oriented Software Engineering. There are many informal states of the art about KE, some home pages gathering information about projects related to KE (e.g., [2]) and also surveys about tools (e.g., [3]) but we found no KB server (nor static ontology) about KE research ideas, technique or tools.

[10] showed how our KB server WebKB-2 provides the above cited minimal supports for the collaborative building of a large well organized KB or semantic network (with formal or informal nodes) and how the approach advantageously compares with less structured ones (e.g., [13]) for knowledge retrieval and comparison, or for supporting learning and research. [10] used examples from our representation of teaching materials. In this article, after a short summary of WebKB-2's approach, we illustrate the ontology that we have begun to permit a scalable sharing of information about KE. More precisely, we illustrate each of the sections which, to support readability, search, checking and systematic input, we used to modularise the input files that we created for this ontology. These sections have names such as "Domains and Theories", "Tasks and Methodologies", "Structures and Languages", "Tools", "Journals, Conferences and Mailing Lists", "Articles, Books and other Documents" and "People: Researchers, Specialists, Teams/Projects, ...". The input files [9] have names such as "Fields of study", "Systems of logic", "Information Sciences", "Knowledge Management", "Conceptual Graph" and "Formal Concept Analysis" (the last three files specialize the others).

Summary of WebKB-2's approach

[5] introduces three notations used by WebKB-2 - FL (For-links), Formalized English (FE) and FCG (Frame-CG) - derived from the Conceptual Graph linear form (CGLF) [12] to improve on its readability, expressivity and "normalizing" characteristics (their combination is what made Conceptual Graphs famous). Their expressivity is respectively similar to RDF+OWL, CGLF and KIF. FL is adapted to the case of "links" (simple relations between categories or statements) and permits to represent a large volume of knowledge in a structured way and a small amount of space, which is important for browsing a large KB. In the three notations, the connected objects can be formal statements (written in FE or FCG) as well as informal statements (mere strings of characters), thus permitting the users to choose the level of detail that suits their goals and to refine their representations incrementally (if and when they wish to).

The example below is needed for the understanding of later examples. It shows translations of English (E) sentences into FL (note: "<" means "subtype of" and ">" means "subtype"). The first example uses informal terms. The second example shows the creator of each formal term and relation. For example, "wn#body" is an identifier for the Wordnet concept that has for *names* "body", "organic_structure" and "physical_structure". Hence, another identifier for this concept is "wn#body__organic_structure__physical_structure" . Since a name (an informal term) can have many meanings, it can be shared by many categories (concepts or relations). The KB of WebKB-2 was created by transforming WordNet 1.7 into a genuine lexical ontology and extending it with several top-level ontologies and domain-related ontologies [7]. In WebKB-2, the "wn" creator may be left implicit (it will be omitted in all other examples).

E: Any human body is a body and has at most 2 arms, 2 legs and 1 head. Any arm, leg and head belongs to at most 1 human body. Male_body and female_body are exclusive subtypes of human_body and so are juvenile_body and adult_body.

```
FL: human_body  < body,
      part:  arm[0..1,0..2]   leg[0..1,0..2]   head[1,1],
      > {male_body female_body}  {juvenile_body adult_body};
```

E: According to Jun Jo (who has for user id "jj"), a body (as understood in WordNet 1.7) may have for part (as understood by "pm") a leg (as defined by "fg") and exactly 1 head (as understood by "oc").

```
FL: wn#body   pm#part:   fg#leg (jj)   oc#head[1](jj);
```

The example below shows two small extracts from a "structured discussion" about the use of XML for knowledge representation, a topic that leads to recurrent debates on many KE related mailing lists. The parenthesis are used for two purposes: (i) allowing the direct representation of links from the destination of a link, and (ii) representing meta-information on a link, such as its creator (for example, the user registered as "pm") or a link on this link (e.g., an objection by "pm" on the use of an objection link by "fg", without stating anything about the destination of this link). The content of the sentences and the *indentation* in the

example below should permit the understanding of these two different uses. (Note that in this example the creators of the statements are left implicit but that prefixes such as "pm#" could be used exactly as in the first example above). The use of dashes to list joint arguments/objections (e.g., a rule and its premise) should also be self-explanatory. The use of specialization links between informal statements may seem odd but such links are used in several argumentation systems: they are essential for modularising purposes and for checking the updates of argumentation structures, and hence guiding or exploiting these updates (e.g., the (counter-)arguments for a statement also apply to its specializations and the (counter-)arguments of the specializations are (counter-)examples for their generalizations). Few argumentation systems allow links on links (ArguMed is one of the exceptions) and hence most of these systems force incorrect representations of discussions. Even fewer provide a textual notation that is not XML-based, hence a notation readable and usable without an XML editor or a graphical interface. All our structured discussions are in [9].

```
"XML is useless for knowledge representation and exchange"
  argument:
    ("using XML tools for KBSs is a useless extra task"
        argument: "KBSs do not use XML internally" (pm,
          objection: "XML can be used for knowledge
                      exchange or storage" (fg,
            objection: "it is as easy to use other formats for
                        knowledge exchange or storage" (pm),
            objection: "a KBS has to use other formats for
                        knowledge exchange or storage" (pm)))
    )(pm);

"XML can be used for knowledge exchange or storage"
  argument: - "an XML notation permits classic XML tools
              (parsers, XSLT, ...) to be re-used" (pm)
            - "classic XML tools are usable even if a
              graph-based model is used" (pm),
  argument of:
    ("a KRL should (also) have an XML notation"
        specialization: "the Semantic Web KRL should have
                         an XML notation" (pm),
        specialization of: "a KRL (Knowledge Representation
                            Language) can have an XML notation" (pm)
    )(pm);
```

The approach of WebKB-2, which is based on a KB shared by all its users, supports and encourages knowledge re-use, precision and connectivity, more than any other current approach [6]. Here is a summary of its principles.

Each category has an associated creator who is also represented by a category and thus may have associated statements. Each statement also has an associated creator and hence, if it is not a definition, may be considered as a belief. Any object (category or statement) may be re-used by any user within her statements. Only the creator of an object may remove it but any user may "correct" a belief by connecting it to another belief via a "corrective relation" (e.g., pm#corrective_specialization). (Definitions cannot be corrected since they cannot be false; similarly, definitions from *different* users cannot be inconsistent with each other, they simply define different categories/meanings). If entering a new belief introduces a redundancy or an inconsistency that is detected by the system, it is rejected. The user may either modify her belief or re-enter it again connected by a "corrective relation" to each belief it is redundant or inconsistent with: this makes explicit the disagreement of one user with (her interpretation of) the belief of another user. Knowledge filters exploiting those relations and details about the creators may then be specified by a user for an application or to ease browsing. For example, a user may specify that during her browsing of the KB, she does not want to see statements that have been corrected nor those from people belonging to certain organizations.

Finally, in order to encourage users to enter precise and original statements, in [10] we proposed an algorithm to evaluate the popularity and originality of each contribution and contributor based on votes on statements and argumentation relations from them. This algorithm would ideally be used with parameters given by each user to specify her own view about which statements or users are interesting to view, and hence better filter the KB during her browsing.

The notations, protocols and large ontology proposed by WebKB-2 are necessary to ease and normalize the cooperative construction of a KB but are insufficient: an initial ontology for the targeted domain is also necessary for people to know how to represent their pieces of information so that the KB remains well organized. The next sections discuss this initial ontology for KE.

Domains and Theories

Names used for domains ("fields of study") are very often also names for tasks. Task categories are more convenient for representing knowledge than domain categories because (i) organizing them is easier and less arbitrary, and (ii) many relations (e.g., case relations) can then be used. Since for normalization purposes a choice must be made, whenever suitable we have represented tasks instead of domains. When

names are shared by domain categories and task categories (in WebKB-2, categories can share names but not identifiers), we advise the use of the task categories for indexing or representing resources.

When studying how to represent and relate document subjects/topics (e.g., technical domains), [14] concluded that representing them as *types* was not semantically correct but that mereo-topological relations between *individuals* were appropriate. Our own analysis confirmed this and we opted for (i) an interpretation of theories and fields of study as large "propositions" composed of many sub-propositions (this seems the simplest, most precise and most flexible way to represent these notions), and (ii) a particular part relation that we named ">part" (instead of "subdomain") for several reasons: to be generic, to remind that it can be used in WebKB-2 as if it was a specialization relation (one of the advantages is that the destination category needs not be already declared) and to make clear that our replacement of WordNet hyponym relations between "synonym sets" about fields of study by ">part" relations refines WordNet without contradicting it. Our file on "Fields of study" [9] details these choices. Our file on "Systems of logics" [9] illustrates how for some categories the represented field of study *is* a theory (not a *reference* to it) thus simplifying and normalizing the categorization. Below is an example of relations from WordNet category #computer_science, followed by an example about logical *domains/theories*. When introducing general categories in Information Sciences and Knowledge Management, and links that do not come from WordNet, we used the "generic users" "is" and "km" (anyone can add knowledge for these users).

```
#computer_science__computational_science
   annotation: "engineering science that ...",
   >part:  #artificial_intelligence,
   >part:  is#software_engineering_science (is),
   >part:  is#database_management_science (is),
   >part of: #engineering_science
   part:  #information_theory,
   part of: #information_science;

km#substructural_logic
  annotation: "system of ...",
  >part of: km#intuitionist_logic,
  >part: km#relevance_logic  km#linear_logic;

km#CG_domain__Conceptual_Graphs
  >part of: km#knowledge_management_science,
  object: km#CG_task  km#CG_structure
          km#CG_tool  km#CG_mailing_list,
  url: http://www.jfsowa.com/cg/;
```

To provide a core ontology that will guide the sharing, indexation or representation of techniques in Knowledge Management, hundreds of categories will need to be represented. We have only begun this work. In the KA2 project [1], the ontology was predefined and a good part of it was a hierarchy of 37 Knowledge Acquisition (KA) domains, the names of which also allude to tasks, structures, methods (PSMs) and experiments. E.g., this hierarchy included:

```
reuse_in_KA > ontologies  PSMs;
PSMs > Sysiphus-III_experiment;
```

Tasks and Methodologies

In most model libraries for KA (e.g., the library of KADS), each non-primitive task is linked to *techniques* that can be used for achieving this *task*, and conversely, each technique combines the results of more primitive tasks. We tried this organization but at the level of generality of our current modeling it turned out to be inadequate: it led (i) to arbitrary choices between representing sometimes as a task (a kind of process) or a technique (a kind of process description), or (ii) to the representation of both notions and thus to introduce categories with names such as KA_by_classification_from_people; both cases are problematic for readability and normalization. Similarly, instead of representing *methodologies* directly, that is, as another kind of process description, it seems better to represent the *tasks* advocated by a methodology (including their uppermost supertask: following the methodology). Furthermore, with tasks, many relations can then be used directly: similar relations do not have to be introduced for techniques or methodologies (the relation hierarchy should be kept small, if only for normalization purposes). Hence, we represented all these things as tasks and used multi-inheritance. This considerably simplified the ontology and the source files. Below are some extracts. (Note. The relation "object" has different meanings depending on the connected categories. In FL, FE and FCG, relation names may be used instead of relation identifiers when there is no ambiguity. In this example, the curly brackets enclose open subtype partition of exclusive subtypes.)

```
km#KM_task__knowledge_management_task
  <  is#information_sciences_task,
  >  km#knowledge_representation
     km#knowledge_extraction_and_modelling
     km#knowledge_comparison   km#knowledge_retrieval_task
     km#knowledge_creation   km#classification
     km#KB_sharing_management
     km#mapping/merging/federation_of_KBs
     km#knowledge_translation   km#knowledge_validation
     {km#monotonic_reasoning  km#non_monotonic_reasoning}
     {km#consistent_inferencing  km#inconsistent_inferencing}
     {km#complete_inferencing   km#incomplete_inferencing}
     {km#structure-only_based_inferencing
       km#rule_based_inferencing}
     km#language/structure_specific_task
     km#teaching_a_KM_related_subject
     km#KM_methodology_task,
  object of: km#knowledge_management_science,
  object: km#KM_structure;

  km#knowledge_retrieval_task  < is#IR_task,
    >  {km#specialization_retrieval   km#generalization_retrieval}
       km#analogy_retrieval  km#structure_only_based_retrieval
       {km#complete_retrieval  km#incomplete_retrieval}
       {km#consistent_retrieval  km#inconsistent_retrieval};
```

Structures and Languages

In WebKB-2's top-level ontology [7], pm#description_medium (top supertype of concept types for languages, data structures, etc.) and pm#description_content (top supertype for fields of studies, theories, document contents, softwares, etc.) have for supertype pm#description because (i) such a general type grouping both notions is needed for the signatures of many basic relations, and (ii) classifying WordNet categories according to the two notions would have often led to arbitrary choices. We chose to represent the default ontology of WebKB-2 as being "a part of" WebKB-2 and hence we allowed pieces of information to be related by part relations. To further ease knowledge entering, WebKB-2 allows the use of generic relations such as part, object and support when the intended more precise relations (e.g., pm#subtask or pm#physical_part) can be automatically found.

For similar reasons, to represent "sub-versions" of ontologies, softwares, and more generally, documents, we use types connected by subtype relations. Thus, for example, km#WebKB-2 is a type (not an individual) and hence can be used with quantifiers.

```
km#KM_structure  < is#symbolic_structure,
  >  {km#base_of_facts/beliefs  km#ontology
      km#KB_category  km#KB_statement}
     km#KB  km#KA_model  km#KR_language
     km#language_specific_structure;

  km#ontology
     >  km#domain_ontology km#top_level_ontology
        km#lexical_ontology km#language_ontology
        km#concept_ontology km#relation_ontology
        km#multi_source_ontology__MSO,
     part: 1..* km#KB_category   1..* km#category_definition;

  km#KR_language__KRL__KR_model_or_notation
     >  {km#KR_model/structure  km#KR_notation}
        km#frame_oriented_language
        km#predicate_logic_oriented_language
        km#graph_oriented_language
        km#KR_language_with_query_commands
        km#KR_language_with_scripting_features,
     attribute: km#semantics;

  km#language_specific_structure  > km#CG_structure;

     km#CG_structure  > km#CG_statement  km#CG_language;
```

Tools

The example below illustrates some specialization relations between tools. In our ontology we use FCG for complex descriptions of tools.

```
km#CG_related_tool
  <  km#language/structure_specific_tool,
  >  km#CG-based_KBMS  km#CG_graphical_editor
     km#NL_parser_with_CG_output;

  km#CG-based_KBMS  <  km#KBMS,
    > {km#CGWorld  km#PROLOG\+CG
       km#CoGITaNT  km#Notio  km#WebKB};

     km#WebKB  >  {km#WebKB-1  km#WebKB-2},
                  url: http://www.webkb.org;
```

Articles and other Documents

This example shows a simple document indexation using Dublin Core relations (we have done this for all the articles of ICCS 2002). Representing ideas from articles would be more valuable. Examples of representations of conferences, publishers, mailing lists, researchers and research teams are in [9].

```
[an #article,
   dc#Coverage: km#knowledge_representation,
   pm#title: "What is a Representation?",
   dc#Creator: "R. Davis, H. Shrobe and P. Szolovits",
   pm#object of: (a #publishing, pm#time:1993,
                    pm#place:(the #object_section"14:1 p17-33",
                                  pm#part of: is#AI_Magazine)),
   pm#url:medg.lcs.mit.edu/ftp/psz/k-rep.html];
```

Conclusion

In his description of a "Digital Aristotle", [4] describes a "Knowledge Web" in which researchers could add ideas or explanations of ideas "at the right place" (that is, without introducing redundancies), and suggests that this Knowledge Web should "include the mechanisms for credit assignment, usage tracking, and annotation that the Web lacks", thus supporting a much better re-use and evaluation of the work of a researcher than via the system of article publishing and reviewing. [4] did not give any indication about such mechanisms but the approach of WebKB-2 approach seems to provide a template for them. However, in addition to the guidance provided by the large general ontology, checking mechanisms, edition protocols, notations and knowledge entering forms, our experiments showed that an initial domain specific ontology is also required to guide and normalize the cooperative construction of a knowledge repository in a domain such as KE.

This article showed the principles of our modelling and what this entails for an ontology of KE. Directly representing sentences from documents would not lead to an organised KB: categorising the underlying objects and their relationships is necessary. The approach of dividing each input file into sections corresponding to one major conceptual category eases the search, cross-checking and systematic input of knowledge. This is a scalable scheme: whenever a section grows too big it can be further divided according to subcategories.

The demand for comparing the dozens existing ontology editing tools cannot be satisfied with informal superficial surveys such as [3]. In [8] we categorized 7 CG-related tools according to 160 criteria organized by subtype relations and grouped into six sections and tables. So far, a wiki is used to store this comparison and let CG researchers complement it. We plan to extend this categorization to 50 ontology tools and 250 features, and then formalize it. In addition to supporting conceptual browsing, this will permit us to answer conceptual queries about these tools and generate tables to compare them. Once this work is done, we shall invite KE researchers to represent or index their research tools or ideas into WebKB-2.

References:

[1] Benjamins V.R., Fensel D, Gomez-Perez A., Decker S., Erdmann M., Motta E. and Musen M. *Knowledge Annotation Initiative of the Knowledge Acquisition Community: (KA).* Proceedings of KAW98, Banff, Canada, April 1998.

[2] Clark P. Ongoing KBS Projects and Groups. http://www.cs.utexas.edu/users/mfkb/related.html

[3] Denny M. Ontology Tools Survey, Revisited. http://www.xml.com/pub/a/2004/07/14/onto.html July 14, 2004.

[4] Hillis W.D. *"Aristotle" (The Knowledge Web).* Edge Foundation, No 138, May 2004.

[5] Martin P. *Knowledge representation in CGLF, CGIF, KIF, Frame-CG and Formalized-English.* Proceedings of ICCS 2002, 10th International Conference on Conceptual Structures (Springer Verlag, LNAI 2393, pp. 77-91), Borovets, Bulgaria, July 15-19, 2002.

[6] Martin P. *Knowledge Representation, Sharing and Retrieval on the Web.* Chapter of a book titled "Web Intelligence", (Eds: N. Zhong, J. Liu, Y. Yao; Springer-Verlag, pp. 263-297), January 2003.

[7] Martin P. *Correction and Extension of WordNet 1.7.* Proceedings of ICCS 2003 (Springer Verlag, LNAI 2746, pp. 160-173), Dresden, Germany, July 2003.

[8] Martin P. *CG tools.* http://www.anykb.org/wiki/index.php/CG_tools

[9] Martin P. *Semantic classification of some resources.* http://www.webkb.org/kb/classif/

[10] Martin P., Eboueya M., Blumenstein M. and Deer P. *A Network of Semantically Structured Wikipedia to Bind Information.* Proceedings of E-learn 2006, (pp. 1684-1702), AACE Conference on E-learning in Corporate, Government, Healthcare and Higher Education, Honolulu, Hawaii, October 13-17, 2006.

[11] Skuce D. and Lethbridge T.C. *CODE4: A Unified System for Managing Conceptual Knowledge*. Int. Journal of Human-Computer Studies (42), pp. 413-451, 1995.

[12] Sowa J.F. Conceptual Structures: Information Processing in Mind and Machine. Addison-Wesley, Reading, MA, 1984.

[13] Stutt A. and Motta E. *Semantic Learning Webs*. Journal of Interactive Media in Education, Special Issue on the Educational Semantic Web, 10, 2004.

[14] Welty C.A. and Jenkins J. *Formal Ontology for Subject.* Journal of Knowledge and Data Engineering, 31(2), pp. 155-182, September 1999.