Panel discussion notes:knowledge maintenance
The role of Formal Ontologies

Yannis Kalfoglou
University of Edinburgh,
Division of Informatics,
School of Artificial Intelligence,
Institute for Representation and Reasoning,
80 South Bridge, Edinburgh, EH1 1HN, Scotland
Email: yannisk@dai.ed.ac.uk

Ontologies have become popular in the recent years in the field of artificial intelligence. There exist different types of ontologies and numerous ways of constructing them ([9],[23]). The interpretation of the term varies across different communities and [12],[13] elaborates on terminological clarifications. The engineering community, and particular the KBS community has adopt a short definition proposed by [10]:"An ontology is a specification of a conceptualisation". An elaboration of ontology definition is presented in [23] and [25] where the authors point out the differences between types of ontologies and surveying uses of them in different domains.

The type of ontologies in which we are interested in are the formal ontologies. A formal ontology is a language with a precisely defined syntax and semantics(which may be determined via model theory, proof theory or in terms of another formal ontology). The inferences permitted in the language are constrained by one or more sets of proof rules accompanied by appropriate proof strategies. The forms of description allowed by those using the ontology are required to be consistent with a set of axioms limiting its use, which we call 'ontological constraints'. The aim of the ontology is to provide a language which allows a stipulated group of people to share information reliably in a chosen area of work.

A variety of ontologies have been reported in the literature with emphasis on their intended use for knowledge sharing and reuse. There exist tools for browsing and editing ontologies(i.e.: Ontolingua [7], Ontosaurus([21] and [27])) as well as guidelines and methodologies to be followed on constructing them(i.e.: [23],[8],[2]).

Areas to which ontologies have been applied include: enterprise modelling by using an ontology in textual format(Enterprise ontology [24]) and in computational form as in the TOVE project([6]); business process modelling using the PIF ontology as reported in [20]; requirements engineering by reusing domain and business ontologies as described in [14]; ontology's reuse as reported in [22]; knowledge acquisition by supporting sharing and reuse([1]).

Ontologies provide a set of characteristics that can be used in various ways. Apart from their intended purpose of knowledge sharing and reuse, they are beginning to be used in software design and in particular to support verification and formal evaluation in the early phases of it. This approach, although in its infancy, has already been explored in research experiments([22]). The authors report that: "despite the effort involved the approach was cost-effective and that it would have taken significantly longer to design the knowledge content of the ontology from scratch in the application". Other researchers have used similar techniques([18]) and pointed out the benefits of using an ontology as a starting point in the design of a software product([17]).

By using ontologies as a starting point of software development we hope to gain a higher level of assurance that the early phases of development(i.e. specification) are well defined and evaluated with respect to the ontology that conform to. This assumes that the syntax and semantics of an ontology can be checked and verified against axioms. Note that should one choose to follow this path, it is not necessary strictly to use only the ontology's constructs in the specification. In fact, it is normally impractical to construct an executable specification by using only the ontology's constructs. Other constructs should be included as well, which do not directly benefit from the presence of ontological constraints but will be checked for errors using normal debugging techniques.

In these proceedings([15]), we describe an extension to the role of ontological constraints: we explore their application in the area of software development and in particular the early phases of it. This makes it possible to detect whether parts of the software specification which uses ontological constructs stray out the limits of the ontology. This will reveal a potential error that is difficult to detect since it conforms to the syntax and semantics of the ontology. Ontological constraints are used to restrict possible interpretations that these constructs can have. By using techniques from the logic programming community, as we describe in [16], and these ontological constraints we are able to detect such errors which contributes to the area of software testing by using ontologies as a basis.

However as pointed out in [26] there is a dearth of well-developed applications based on formal ontologies. This contradiction is visible in the field of AI where the few applications that are discussed are intended applications which are yet to be built, or small research prototypes. According to [26] the reason for this is that the more well-developed applications seem to be for ontologies which do not have rich representations of meaning. On the contrary, the AI ontology community concentrates its efforts to semantically rich representations. This, perhaps, explains the greater success of ontology application in areas such as semantic modelling in databases and data warehouses where some of the applications are fielded commercially([26]).

Another open issue in the application of ontologies is the selection of the right ontology for the application to be built. There are tools that help the developer to familiarise himself with the terminology used in a domain represented in an ontology(i.e. Ontosaurus project, [21]) and tools that support collaboratively construction by using dialogue techniques(i.e. Tadzebao project, [5]). However, it is crucial for the prospective user of an ontology to investigate whether it supports the intended task of the application and to what extent. Towards this direction, in [3] the authors elaborate on the usage of specific tasks and methods as ontologies of problem-solving methods that can be reused in similar applications.

The appropriateness of the ontology can be checked by a set of questions derived from the domain which the ontology should be able to answer. This technique has been explored in [11] where the author describes how a set of questions, called 'competency questions', were used to evaluate the expressiveness of the ontology that is required to represent them and to characterise their solutions. A similar approach is described in [4] where a set of 'sample questions' were used to evaluate and debug ontologies that were built by different teams as part of an ontology sharing experiment. The underpinning theory behind these questions is the ontological commitment as defined in [13]: "An ontology is a logical theory accounting for the intended meaning of a formal vocabulary, i.e. its ontological commitment to a particular conceptualisation of the world. The intended models of a logical language using such a vocabulary are constrained by its ontological commitment." The author states that the ontological commitment should be made explicit when applying the ontology in order to facilitate its accessibility, maintainability, and integrity. It is claimed that this will lead to an increase of transparency for the application software which based on that ontology.

We argue that ontologies can be useful for knowledge maintenance tasks. As Menzies point out in this panel's abstract([19]) ontologies can be used as a "pointer tool" that facilitates exploration of a new domain. It helps developers to "kick start" an application. It is crucial for the success of this method to preserve the support of the underlying ontology throughout the life-cycle of the application. As the application evolves over time the underlying ontology should reflect these changes. This might be possible by having the application conform to several domain-specific ontologies which in turn are part of a broad ontology. This will make it, arguably, easier to reflect a change in the application by updating a domain-specific ontology or choosing a new one.

Ontologies as a mean of representing meta-knowledge have been investigated over the last 10-12 years. Recently, they have begun to be used in more specific and practical ways in the software development life-cycle itself.

Acknowledgements

The research described here is supported by a European Union Marie Curie Fellowship (programme: Training and Mobility of Researchers).

References

[1] R. Benjamins and D. Fensel. The Ontological Engineering Initiative-KA2. In N. Guarino, editor, *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, Trento, Italy*, pages 287–301. IOS Press, June 1998.

[2] M. Blazquez, M. Fernadez, J.M. Garcia-Pinar, and A. Gomez-Perez. Building Ontologies at the Knowledge Level using the Ontology Design Environment. In *Proceedings of the 11th Knowledge Acquisition Workshop, KAW98, Banff, Canada*, April 1998.

[3] B. Chandrasekaran, J.R. Josephson, and R. Benjamins. The Ontology of Tasks and Methods. In *Proceedings of the 11th Knowledge Acquisition Modeling and Management Workshop, KAW'98, Banff, Canada*, April 1998.

[4] Cohen,P. and Chaudhri,V. and Pease,A. and Schrag,R. Does prior knowledge facilitate the development of knowledge-based systems? In *Proceedings of the Sixteenth National Conference on Artificial Intelligence, Orlando, FL, USA, AAAI'99*, July 1999.

[5] J. Domingue. Tadzebao and WebOnto: Discussing, Browsing, and Editing Ontologies on the Web. In *Proceedings of the 11th Knowledge Acquisition, Modelling and Management Workshop, KAW'98, Banff, Canada*, April 1998.

[6] Enterprise Integration Laboratory, University of Toronto, Canada. TOVE Project. available from http://www.ie.utoronto.ca/EIL/tove/ontoTOC.html, July 1995.

[7] A. Farquhar, R. Fikes, W. Pratt, and J. Rice. The Ontolingua Server: a Tool for Collaborative Ontology Construction. In *proceedings of the 10th Knowledge Acquisition Workshop, KAW'96,Banff,Canada*, November 1996. Also available as KSL-TR-96-26.

[8] M. Fernandez, A. Gomez-Perez, and N. Juristo. METHONTOLOGY: From Ontological Arts Towards Ontological Engineering. In *Proceedings of the AAAI-97 Spring Symposium Series on Ontological Engineering, Stanford, USA*, pages 33–40, March 1997.

[9] N. Fridman Noy and C.D. Hafner. The State of the Art in Ontology Design: A Survey and Comparative Review. *AI Magazine*, pages 53–74, 1997.

[10] T.R. Gruber. A Translation Approach to Portable Ontologies. *Knowledge Acquisition*, 5(2):199–220, 1993.

[11] M. Gruninger. Designing and Evaluating Generic Ontologies. In *Proceedings of the 12th European Conference of Artificial Intelligence*, August 1996.

[12] N. Guarino and P. Giaretta. Ontologies and Knowledge Bases: Towards a Terminological Clarification. *Towards Very Large Knowledge Bases*, 1995. IOS Press, Amsterdam.

[13] Guarino,N. Formal Ontology and Information Systems. In N. Guarino, editor, *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, Trento, Italy*, pages 3–15. IOS Press, June 1998.

[14] Z. Jin, D. Bell, F.G. Wilkie, and D. Leahy. Automatically Acquiring Requirements of Business Information Systems by Reusing Business Ontology. In Gomez-Perez,A. and Benjamins,R., editor, *Proceedings of Workshop on Applications of Ontologies and Problem Solving Methods, ECAI'98, Brighton, England*, August 1998.

[15] Kalfoglou,Y. and Robertson,D. A Case Study in Applying Ontologies to Augment and Reason about the Correctness of Specifications. In *Proceedings of the 11th International Conference in Software Engineering and Knowledge Engineering, SEKE'99, Kaiserslauten, Germany*, June 1999. Also as Dept. of Artificial Intelligence, University of Edinburgh, Research Paper No.927.

[16] Kalfoglou,Y. and Robertson,D. Use of Formal Ontologies to Support Error Checking in Specifications. In D. Fensel and R. Studer, editors, *Proceedings of the 11th European Workshop on Knowledge Acquisition, Modelling and Management(EKAW99), Dagstuhl, Germany*, pages 207–220, May 1999. Also as: University of Edinburgh, Dept. of AI, Research Paper No.935.

[17] W. Mark. Ontologies as Representation and Re-Representation of Agreement. In *Proceedings of the 5th International Conference on Principles of Knowledge Representation and Reasoning, KR'96, Massachusetts, USA*, 1996. Position paper presented on the panel: *Ontologies: What are they and where's the research.*

[18] W. Mark, S. Tyler, J. McGuire, and J. Schossberg. Commitment-Based Software Development. *IEEE Transactions on Software Engineering*, 18(10):870–884, October 1992.

[19] T. Menzies. Knowledge Maintenance Heresies: Meta-Knowledge Complicates KM. In *Proceedings of the 11th International Conference in Software Engineering and Knowledge Engineering, Kaiserslauten, Germany*, June 1999. Notes of Panel: "Does Meta-Knowledge Confuse or Complicate Maintenance?".

[20] S. Polyak, J. Lee, M. Gruninger, and C. Menzel. Applying the Process Interchange Format(PIF) to a Supply Chain Process Interoperability Scenario. In A. Gomez-Perez and R. Benjamins, editors, *Proceedings of Workshop on Applications of Ontologies and Problem Solving Methods, ECAI'98, Brighton, England*, August 1998.

[21] B. Swartout, R. Patil, K. Knight, and T. Russ. Toward Distributed Use of Large-Scale Ontologies. In *Proceedings of the 10th Knowledge Acquisition, Modeling and Management Workshop(KAW'96),Banff,Canada*, November 1996.

[22] M. Uschold, P. Clark, M. Healy, K. Williamson, and S. Woods. An Experiment in Ontology Reuse. In *Proceedings of the 11th Knowledge Acquisition Workshop, KAW98, Banff, Canada*, April 1998.

[23] M. Uschold and M. Gruninger. Ontologies: principles, methods and applications. *The Knowledge Engineering Review*, 11(2):93–136, November 1996.

[24] M. Uschold, M. King, S. Moralee, and Y. Zorgios. The enterprise ontology. *Knowledge Engineering Review*, 13(1), February 1998. Also available as AIAI-TR-195 from AIAI, University of Edinburgh.

[25] Uschold,M. Knowledge level modelling: concepts and terminology. *The Knowledge Engineering Review*, 13(1):5–29, February 1998.

[26] Uschold,M. Where are the Killer Apps? In Gomez-Perez,A. and Benjamins,R., editor, *Proceedings of Workshop on Applications of Ontologies and Problem Solving Methods, ECAI'98, Brighton, England*, August 1998.

[27] A. Valente, T. Russ, R. MacGrecor, and W. Swartout. Building and (Re)Using an Ontology for Air Campaign Planning. *IEEE Intelligent Systems*, 14(1):27–36, January 1999.