PATTERNS IN ONTOLOGY ENGINEERING: CLASSIFICATION OF ONTOLOGY PATTERNS

Eva Blomqvist
School of Engineering, Jönköping University
P.O. Box 1026, SE-551 11 Jönköping, Sweden
eva.blomqvist@ing.hj.se

Kurt Sandkuhl
School of Engineering, Jönköping University
P.O. Box 1026, SE-551 11 Jönköping, Sweden
kurt.sandkuhl@ing.hj.se

Keywords: Ontology Engineering, Ontology Patterns, Pattern Classification.

Abstract: In Software Engineering, patterns are an accepted way to facilitate and support reuse. This paper focuses on patterns in the field of Ontology Engineering and proposes a classification scheme for ontology patterns. The scheme divides ontology patterns into five levels: Application Patterns, Architecture Patterns, Design Patterns, Semantic Patterns, and Syntactic Patterns. Semantic and Syntactic Patterns are quite well-researched but the higher levels of pattern abstraction are so far almost unexplored. To illustrate the possibilities of patterns on these levels some examples are discussed, together with ideas of future work.

1 INTRODUCTION

Recent developments in the area of Ontology Engineering involve semi-automatic ontology construction to reduce time and effort of constructing an ontology. In particular for small-scale application contexts, reduction of effort and expert involvement is an important requirement.

Ways of reducing this effort are to further facilitate semi-automatic construction of ontologies, but also to introduce reuse in Ontology Engineering. Patterns have proved to be a fruitful way to handle the problem of reuse. In Software Engineering it is already the commonly accepted way to build software and there are not only Software Design Patterns, but also Data Model Patterns, generic Problem Solving Methods, Object System Models, and many other approaches. Classification of ontology patterns is a first step towards a structured use of patterns in Ontology Engineering. The main contribution of our research, and focus of this paper, is the classification and categorisation of patterns for Ontology Engineering.

In section 2 the background of the research is presented. Further, in section 3, our approach for classifying ontology patterns is described and some examples of the different levels are discussed. In the final part, section 4, our research is summarised and some ideas for future work are presented.

2 ONTOLOGY PATTERNS TODAY

Ontology is a popular term today, used in many areas and defined in many different ways. In this paper ontology is defined as:

An ontology is a hierarchically structured set of concepts describing a specific domain of knowledge that can be used to create a knowledge base. An ontology contains concepts, a subsumption hierarchy, arbitrary relations between concepts, and axioms. It may also contain other constraints and functions.

Even using this definition, ontologies can be used for many different purposes, and they can be constructed and structured in many different ways. One of the most common ways to describe the level of generality of an ontology is by using the structure in (Guarino, 1998). The structure shows how a general Top-level Ontology can be specialised into a Domain Ontology or a Task Ontology, which in turn can be combined into an Application Ontology (describing a certain task specialised for a specific domain).

Another categorisation of ontologies can be obtained by classifying them by their intended use (van Heijst et al., 1997). There are three main levels, Terminological Ontologies, Information Ontologies and Knowledge Modelling Ontologies, where each level adds further complexity to the ontology structure. For example, in a Terminological Ontology often a simple taxonomy of terms is enough, while in a Knowledge Modelling Ontology more advanced axioms and constraints are very often required.

The ontology community has not yet adopted the pattern idea on a broader scale. There exist a few patterns for ontologies, and all of them are specialised for some specific kind of ontology (with respect to both level of generality and usage area, as defined above).

For specific ontology languages, patterns are being developed, e.g. the OWL-patterns of (W3C-SWBPD, 2004). Examples of meta-structures for ontologies, which can be denoted patterns, are Semantic Patterns for describing implementation independent logical constructs (Stuckenschmidt and Euzenat, 2001)(Staab et al., 2001). These patterns can for example describe notions like locally inverse relations and composition of relations. Similar to Software Design Patterns are the Ontology Design Patterns discussed in (Gangemi, 2004), and the Design Patterns developed for ontologies in Molecular Biology (Reich, 1999). These patterns are quite general and describe for example how to keep track of updates between different partitions of an ontology.

3 CLASSIFICATION OF ONTOLOGY PATTERNS

In order to study the use of ontology patterns in a broader perspective, there is need for a general classification of patterns. In this section such a classification is presented and discussed.

3.1 Classification

Our classification of ontology patterns uses the following levels (an illustration of the levels and their interdependencies can be seen in Figure 1):

- Application Patterns - Purpose, scope, usage and context of the implemented ontology or ontologies, including interfaces and relations to other systems.
- Architecture Patterns - A description of how to combine or arrange implemented Design Patterns in order to fulfill the overall goal of the ontology.
- Design Patterns - A small collection of Semantic Patterns that together create a common and generic construct for ontology development.
- Semantic Patterns - Language independent description of a certain concept, relation or axiom. A meta-description of a Syntactic Pattern.
- Syntactic Patterns - Language specific ways to arrange representation symbols, to create a certain concept, relation or axiom.

To really benefit from ontology patterns they should be studied, and used, at all these levels. As noted in section 2, there are some ontology pattern approaches present today, but these are mostly connected to the lower levels, such as Syntactic Patterns (W3C-SWBPD, 2004), Semantic Patterns (Staab et al., 2001) and Design Patterns (Gangemi, 2004) (Reich, 1999).

Ontology Application Patterns aim to describe generic ways to use the implemented ontologies, in terms of purpose, context, interfaces etc. This idea of abstracting the best ways to apply and use an ontology, or several ontologies, within some context or application is an important issue. Here many models and common practices exist, but not formalised as patterns. A simple generalisation of existing models could result in initial Application Patterns for ontologies. The application level patterns will be very much dependent on the ontology usage area, and may also be domain dependent.

Ontology Architecture Patterns aim to describe a generic way to design the overall structure of an ontology, in order to fulfill the goal of the ontology in question. So far, no Architecture Patterns for ontologies have been presented. Important questions are whether to divide the ontology into components or modules, or to divide it into layers or use other construction principles. On the architecture level there could be great possibilities to draw from experiences in other areas. Architectures tend to be similar whether it comes to Problem Solving Methods, software or ontologies. Architectures also do not tend to be application or domain specific, but the patterns created certainly need to be adapted to either manual or automatic use.

Ontology Design Patterns aim to describe a generic recurring construct in ontologies. When implemented, the construct will form a small piece of the complete ontology (perhaps a component or module). There exist patterns that are generic and domain independent, for manual construction of domain ontologies. Such patterns can be very useful but there is also a need for more specific patterns, e.g. domain dependent patterns, both for manual and automatic use. The main idea of patterns for automatic use is to be specific enough, allowing them to be used automatically or semi-automatically, and at the same time generic enough to be useful in several ontologies of a certain domain.

A continuous application of all levels of the pattern classification above would require defined patterns for all different kinds of ontologies (both level of generality and usage area as defined in section 2). Yet another aspect to be taken into account is whether the patterns will be applied manually or automatically. In Ontology Engineering today, the approaches tend more and more toward semi-automatic construction, even more so than in Software Engineering. Automatically applied patterns will probably play a very important role in the future. Thus, when exploring and developing a full-covering system of patterns for ontologies, it is important to consider both the ontology categorisation, the pattern categorisation and the notion of manual or automatic pattern usage.

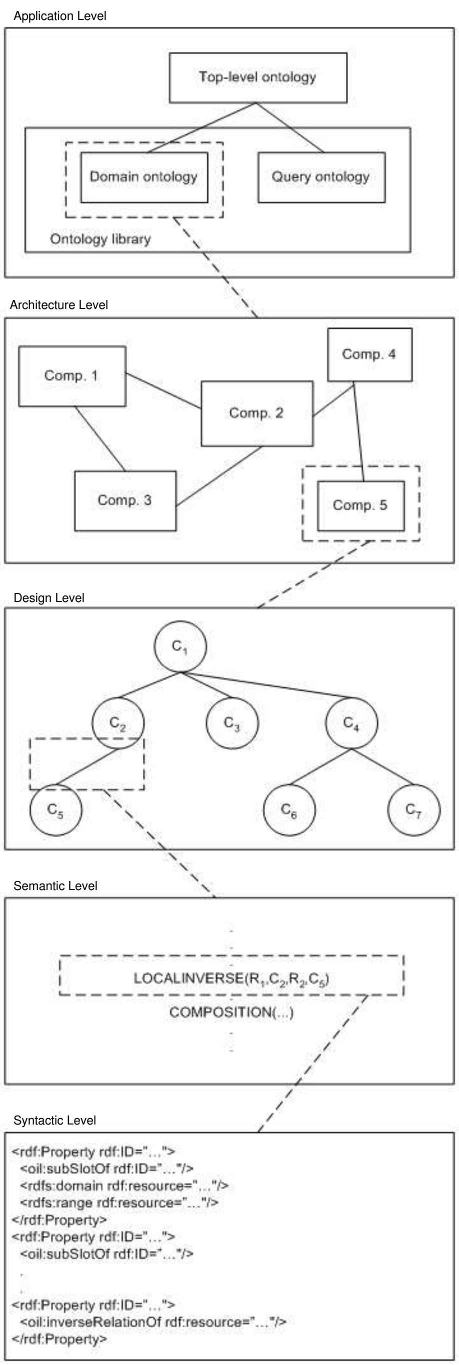

Figure 1: Illustration of the classification levels as a continuous increase of granularity.

3.2 Discussion of Selected Examples

Our research mainly has a focus on ontology patterns for automatic use, and specifically for Information Ontologies on the domain or application level within enterprises.

One can imagine two different approaches when addressing this problem. The first one is to take all existing ontologies and derive patterns from them. The other one is to develop criteria of "good design" and construct patterns that reflect these principles. Unfortunately, the second approach is not practically feasible since it is almost impossible to derive criteria of how to design ontologies for all situations. The first approach is the common way, in for example Software Engineering, to "discover" patterns using existing structures. Unfortunately, enterprise ontologies are so far quite scarce and often sparsely documented, which makes it hard to extract patterns this way.

One "middle course" is to draw from knowledge already accumulated in other areas. Many patterns in Computer Science also describe some kind of knowledge, although they might have a different purpose. Our approach is to study those patterns in order to develop ontology patterns. So far the focus has been to develop an overall picture of the ontology area, by developing the classification structure discussed previously, and to put existing approaches into this categorisation. In the following sections first ideas for populating the higher pattern levels are presented.

3.2.1 Ontology Application Patterns

No Application Patterns have so far been formalised for ontologies, as noted earlier, but there exist many models of ontology usage. The models typically show how different ontologies interact and depend on each other in a specific system. As a starting point, these types of models (for example like in (Levashova et al., 2003)) could be abstracted into Ontology Application Patterns.

3.2.2 Ontology Architecture Patterns

At the next level, some Software Architecture Patterns might be quite easily adopted by the ontology community. There are several approaches to modularisation of ontologies (Stuckenschmidt, 2003) (Rector, 2003), which in combination with a Software Architecture Pattern, for example similar patterns to those developed by Shaw (Shaw, 1996), could become an architecture for ontologies. An example could be the Implicit Invocation Architecture Pattern (Shaw, 1996), which instead of objects or processes could be used with ontology modules.

3.2.3 Ontology Design Patterns

The third level in focus is the Ontology Design Patterns. Here patterns similar to Software Design Patterns have already been suggested, so the interesting areas to explore involve more domain and application specific patterns for semi-automatic use. For testing the suitability of different patterns in Computer Science to act as templates when constructing Ontology Design Patterns, some example ontologies were created. These were evaluated against real-world applications in order to decide which patterns to focus on. Some patterns considered interesting for constructing Ontology Design Patterns are Data Model Patterns (Hay, 1996) (Silverston, 2001), Object System Models (Sutcliffe, 2002) and Analysis Patterns (Fowler, 1997).

3.2.4 Remaining Issues

So far our focus has been on Ontology Design Patterns in practice, and currently an evaluation is being conducted to see how useful the ontologies suggested above can be, and how they might be generalised into ontology patterns. The evaluation has shown that one problem is to transfer constraints in the existing patterns into axioms of the ontologies in a structured way. Another problem is the differences in generality of the existing patterns. Also, the underlying assumptions that the patterns are based on might differ between the ontology field and the field where the pattern originated.

4 SUMMARY AND FUTURE WORK

The main contribution of this research, so far, is the classification of ontology patterns into five different levels. When using these levels it is also important to consider the kind of ontology to be built, its intended usage, and especially how it will be constructed. The notion of semi-automatic construction poses new and different requirements on possible patterns, they need to be more specific, appropriately formalised and maybe even domain dependent.

Our work has so far focused on testing patterns from other Computer Science areas in order to evaluate their usefulness when adapted to the ontology area. When considering future work, it has been shown that many areas of the ontology pattern field require more research efforts, in order to reach a full-covering system of patterns. Our research will continue to focus on the patterns intended for automatic use since they are the natural next step in further facilitating semi-automatic ontology construction.

REFERENCES

Fowler, M. (1997). *Analysis Patterns - Reusable Object Models*. Addison-Wesley.

Gangemi, A. (2004). Some design patterns for domain ontology building and analysis. Available at: http://www.loa-cnr.it/Tutorials/OntologyDesign Patterns.zip, downloaded 2004-10-04.

Guarino, N. (1998). Formal Ontology and Information Systems. In *Proceedings of FOIS'98*, pages 3–15.

Hay, D. C. (1996). *Data Model Patterns - Conventions of Thought*. Dorset House Publishing.

Levashova, T. V., Pashkin, M. P., Shilov, N. G., and Smirnov, A. V. (2003). Ontology Management, II. *Journal of Computer and Systems Sciences International*, 42(5):744–756.

Rector, A. (2003). Modularisation of Domain Ontologies Implemented in Description Logics and related formalisms including OWL. In *Proceedings of the international conference on Knowledge capture*, pages 121–128, Sanibel island. ACM Press.

Reich, J. R. (1999). Ontological Design Patterns for the Integration of Molecular Biological Information. In *Proceedings of the German Conference on Bioinformatics GCB'99*, pages 156–166.

Shaw, M. (1996). Some Patterns for Software Architectures. In Vlissides, J. M., coplien, J. O., and Kerth, N. L., editors, *Pattern Languages of Program Design*, volume 2, pages 255–269. Addison-Wesley.

Silverston, L. (2001). *The Data Model Resource Book, Revised Edition*, volume 1. John Wiley & Sons.

Staab, S., Erdmann, M., and Maedche, A. (2001). Engineering Ontologies using Semantic Patterns. In O'Leary, D. and Preece, A., editors, *Proceedings of the IJCAI-01 Workshop on E-business & The Intelligent Web*, Seattle.

Stuckenschmidt, H. (2003). Modularization of Ontologies. WonderWeb Deliverable D21, available at: http://wonderweb.semanticweb.org/deliverables /D21.shtml.

Stuckenschmidt, H. and Euzenat, J. (2001). Ontology Language Integration: A Constructive Approach. In *Proceedings of the Workshop on Application of Description Logics at the Joint German and Austrian Conference on AI, CEUR-Workshop Proceedings*, volume 44.

Sutcliffe, A. (2002). *The Domain Theory - Patterns for Knowledge and Software Reuse*. Lawrence Erlbaum Associates.

van Heijst, G., Schreiber, A. T., and Wielinga, B. J. (1997). Using explicit ontologies for KBS development. *International Journal of Human-Computer Studies*, 46(2-3):183–292.

W3C-SWBPD (2004). Semantic Web Best Practices and Deployment Working Group. Available at: http://www.w3.org/2001/sw/BestPractices/.