Integrating Knowledge Bases Using UML Metamodels

Thies Pfeiffer, Ian Voss
{tpfeiffe,ivoss}@techfak.uni-bielefeld.de
SFB 360; University of Bielefeld; 33594 Bielefeld, Germany

1 Introduction

When merging different knowledge bases one has to cope with the problem of classifying and linking concepts as well as the possibly heterogeneous representations the knowledge is expressed in. We are presenting an implementation that follows the Model Driven Architecture (MDA) [Miller and Mukerji, 2003] approach defined by the Object Management Group (OMG). Metamodels defined in the Unified Modeling Language (UML) are used to implement different knowledge representation formalisms. Knowledge is expressed as a Model instantiating the Metamodel. Integrating Metamodels are defined for merging knowledge distributed over different knowledge bases.

2 Applying the Model Driven Architecture

UML has its origins in software engineering. It has been used not only for modeling applications, but also for modeling of processes and, e.g., database design. UML is well known in the knowledge modeling community, [Cranefield and Purvis, 1999] have already used UML as representation language for ontology modeling by defining an appropriate semantic. Our approach is different in that it does not try to define a semantic for UML itself, but uses the concepts of UML Metamodels to describe existing knowledge representation formalisms with a well defined (and established) semantic. This can easily be shown, e.g., for KL-ONE [Brachman and Schmolze, 85].

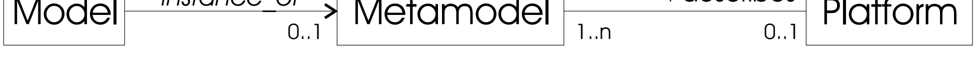

Figure 1: According to the MDA, the Model Driven Architecture, a Model is an instance of a certain Metamodel. Metamodels are describing different Platforms. A knowledge base (Model) for linguistic concepts can, e.g., be an instance of a KL-ONE Metamodel. The Platform in this situation is a KL-ONE runtime written in Prolog or any other suitable programming language.

Based on these Metamodels additional integrating Metamodels can be defined, which subsume the original Metamodels and allow for a linking of concepts between Models based on different Metamodels. From our point of view this approach has the following additional advantages:

Metamodels (M2)	**integrating Metamodel**			
	KL-ONE	Scenegraph	COAR	MURML
Models (M1)	Ontology	Scene	COAR Knowledgebase	Gestuary
Instances (M0)	Speech	Visual Objects	Building Blocks	Gestures

Table 1: DOORS uses an integrated Metamodel based on four different knowledge representation formalisms which have been reformulated as UML Metamodels. COAR and MURML are formalisms developed in our AI lab at the University of Bielefeld.

- Both the formalism AND the knowledge itself can be represented in UML.
- Different formalisms can be expressed in the same standardized language.
- Knowledge bases different in content and structure can be edited within the same environment.
- Links between instances of different Metamodels can be described on basis of an integrating Metamodel.
- Mappings between different Metamodels (here different knowledge representation formalisms) can be defined for migrating knowledge from one representation to another.

3 Conclusion

This approach has been implemented in the DOORS system for distributed ontologybased object reference resolution at the University of Bielefeld [Pfeiffer et al., 2003]. In this system dynamic knowledge bases are merged in a realtime Virtual Reality (VR) setting. Evaluation of the different knowledge bases (see Table 1) is done to provide a cognitive equipment for an anthropomorphic interface agent in the VR environment [Voss and Wachsmuth, 2003].

Generally when following the MDA approach, existing knowledge representation formalisms can be described and content can be instantiated in an integrated manner. Mappings between formalisms and integrating Metamodels can then be used to transform or merge heterogeneous knowledge bases.

References

[Brachman and Schmolze, 85] Brachman, R. and Schmolze, J. (85). An Overview of the KL-ONE Knowledge Representation System. *Cognitive Science*, 2:171–216.

[Cranefield and Purvis, 1999] Cranefield, S. and Purvis, M. (1999). UML as an Ontology Modeling Language. In *Proceedings of the Workshop on Intelligent Information Integration, 16th Int. Joint Conference on AU (IJCAI-99)*, Germany.

[Miller and Mukerji, 2003] Miller, J. and Mukerji, J. (2003). MDA Guide Version 1.0. Technical report, Object Management Group.

[Pfeiffer et al., 2003] Pfeiffer, T., Voss, I., and Latoschik, M. E. (2003). Resolution of Multimodal Object References using Conceptual Short Term Memory. In Schmalhofer, F. and Young, R., editors, *Proceedings of the EuroCogSci03*, page 426. Lawrence Erlbaum Associates Inc.

[Voss and Wachsmuth, 2003] Voss, I. and Wachsmuth, I. (2003). Anticipation in a VR-based Anthropomorphic Construction Assistant. In Jacko, J. and Stephanidis, C., editors, *Human-Computer Interaction, Theory and Practice (Part I)*, pages 1283–1287, London. Lawrence Erlbaum Associates.