MOMENT: a formal MOdel manageMENT tool♦

Artur Boronat, José Á. Carsí, Isidro Ramos

Department of Information Systems and Computation
Polytechnic University of Valencia
Camí de Vera s/n
46022 Valencia-Spain
{aboronat | pcarsi | iramos}@dsic.upv.es

Model-Driven Engineering considers models as the main assets in the software development process. Models collect the metadata that describes the information system at a high level of abstraction, which permits the development of the application in an automated way following generative programming techniques. Traditionally, the tasks that are involved in this process (such as model integration or model transformation) have usually been solved in an ad-hoc manner for a specific context or metamodel: relational databases, XML schemas, ontologies, aspect-oriented programming, etc. Nowadays, Model Management [1] is a new emergent discipline that pursues an abstract reusable solution for problems of this kind. Model management was presented by Bernstein as an approach to deal with software artifacts by means of generic operators that do not depend on metamodels by working on mappings between models. Operators of this kind deal with models as first-class citizens, increasing the level of abstraction of the solution by avoiding working at a programming level and improving the reusability of the solution.

Based on our experience in formal model transformation and data migration [2,3], we are working on the application of the Model Management trend to the context of the Model-Driven Engineering, regarding the guidelines that are specified in the Model-Driven Architecture initiative. We are developing a framework, called MOMENT (MOdel manageMENT), which is embedded into the Eclipse platform and that provides a set of generic operators to deal with models through the Eclipse Modeling Framework (EMF). EMF provides a close implementation to the MDA guidelines. This framework enables the automatic importation of software artifacts from heterogeneous datasources: UML models (by means of visual modeling environments), relational schemas of any relational database management system (through the Rational Rose tool) and XML schemas. Moreover, third-party researchers and developers are bringing new tools to work on ontologies through EMF and graphical Domain Specific Languages. Therefore, EMF has become an industrial framework for MDE.

The algebra of model management operators, which was proposed by Bernstein to deal with models and mappings between models as first-class citizens, has been adapted and directly specified as a generic algebra by using the algebraic specification formalism in the MOMENT framework. This algebra has been specified in a parametric module that receives the algebraic signature of a specific metamodel as

♦ This work was supported by the Spanish Government under the National Program for Research, Development and Innovation, DYNAMICA Project TIC 2003-07804-C05-01.

argument. The instantiation of the parameterized module with a specific metamodel results in an algebra that contains the constructors for the metamodel concepts and model management operators that can be directly applied to all the models that conform such a metamodel.

This formalism enables the specification of software artifacts as terms of an algebra, providing the following features:

- Modularization. Algebraic specifications are defined in modules so that they can be reused by other modules.
- Composition of modules by means of an importation mechanism that preserve monotonicity. This ensures that the meaning of the imported modules is preserved, even though when new functionality is added. This enables the definition of generic model management operators that can be customized to a certain metamodel.
- Scalability of the operators. Their declarative definition can be customized or modified by simply adding axioms to its semantics.
- Well-defined support for parameterized algebraic specifications through the *Pushout* concept of category theory. This constitutes the key feature for the definition of generic model management operators in the MOMENT framework.

To take advantage of these features we have used Maude [4]. This is an algebraic specification language that belongs to the OBJ family and that provides support for the above features through equational logic. We have developed a plug-in that embeds the Maude environment into the Eclipse framework so that we can use it for our purposes. Our framework also provides support for traceability in order to trace the specific transformations that have been applied to a set of models.

We have developed a set of bridges between the technical spaces EMF and Maude by using generative techniques. These technical bridges provide interoperability between an efficient robust environment for algebraic specifications (Maude) and a well-known industrial modeling tool (EMF). On the one hand, the algebraic specifications formalism provides some desired features: abstraction, modularization, subtyping, semantic validity, genericity by means of parameterization, etc. On the other hand, visual modeling tools provide the interface that is needed to provide usability and industrial application to our model management approach.

References

1. Bernstein, P.A: Applying Model Management to Classical Meta Data Problems. pp. 209-220, CIDR 2003.
2. Boronat, A., Pérez, J., Carsí, J. Á., Ramos, I.: Two experiencies in software dynamics. *Journal of Universal Science Computer*. Special issue on Breakthroughs and Challenges in Software Engineering. Vol. 10 (issue 4). April 2004.
3. Boronat, A., Carsí, J.Á., Ramos, I.: Automatic Reengineering in MDA Using Rewriting Logic as Transformation Engine. IEEE Computer Society Press. 9th European Conference on Software Maintenance and Reengineering. Manchester, UK. 2005.
4. Clavel, M., Durán, F., Eker, S., Lincoln, P., Martí-Oliet, N., Meseguer, J., Quesada, J.F.: Maude: specification and programming in rewriting logic. Theoretical Computer Science, 285(2):187-243, 2002.