

Moritz Balz: Embedding Model Specifications in Object-Oriented Program Code: A Bottom-up Approach for Model-based Software Development

Promotion: Universität Duisburg-Essen, Fakultät für Wirtschaftswissenschaften

Erstgutachter: Prof. Dr. Michael Goedicke, Universität Duisburg-Essen

Zweitgutachter: Prof. Dr. Bruno Müller-Clostermann, Universität Duisburg-Essen

Datum der Prüfung: 31. August 2011

Veröffentlichung: <http://duepublico.uni-duisburg-essen.de/>

Kurzfassung:

Models in software engineering are descriptive structures so that transformations can connect their contents at a semantic level. In model-based software development, algorithmic program code usually exists alongside models derived from them or with the purpose to amend them. While thus both kinds of notations must be considered by developers, no consistent mapping is given since transformations between models and code are usually unidirectional for code generation. This impedes a continuous integration of both, limits the applicability of models, and prevents error tracking and monitoring at run time with respect to models.

In this thesis, the approach of embedded models is introduced. Embedded models define patterns in program code whose elements have formal relations to models and can be executed by reflection at the same time. Model specifications are thus embedded in implementations and can be accessed by bidirectional transformations for design, verification, execution, and monitoring. The thesis focuses on the development of such patterns and their precise description as well as on the connection to other program code surrounding embedded models. Implementations are described for two modeling domains, state machines and process models, including tools for design, verification, execution, monitoring, and design recovery. The approach is evaluated with two case studies, the modeling of a real-world load generator for performance tests and the development of model-based educational graphical scenarios for university teaching.

Both case studies show that the approach is valid and fulfills its purpose for a certain class of applications. Focusing on the integration in implementations, embedded models are thus a bottom-up approach for model-based software development.