eMoflon: Leveraging EMF and Professional CASE Tools

Anthony Anjorin, Marius Lauder, Sven Patzinay and Andy Schürr

{fanthony.anjorin, marius.lauder, sven.patzina, andy.schuerrg}@es.tu-darmstadt.de

Abstract: The primary goal of Model-Driven Development Software Development (MDSD) is to improve productivity by providing tools that are tailored for a specific domain. Such domain specific Computer Aided Software Engineering (CASE) tools exploit domain knowledge to further raise abstraction levels and automate complex, but routine programming tasks whenever possible. Anticipated advantages include an increase in speed of development and quality of software, improved response to change, support for interoperability, and a reduced gap between requirements and the actual specification of a solution.

MOFLON[1] is a tool for building tools: a meta-CASE tool that supports the MDSD process by providing a metamodeling and (bidirectional) model transformation framework. Development on MOFLON started in 2002 with major goals including conformity with standards, reuse of stable external components, and industrial relevance implying scalability, usability and the possibility of integrating Commercial Off-The-Shelf (COTS) tools and established workflows with MOFLON.

In recent years, there has been a lot of progress in the MDSD community and especially the Eclipse Modeling Framework (EMF) has established itself as a de facto standard, offering stable and well-tested components. We are convinced that it is high time to reengineer MOFLON and leverage modern MDSD technology to be able to concentrate fully on our core competencies including (bidirectional) model transformations via graph transformations.

eMoflon is a complete re-engineering of MOFLON that tailors Enterprise Architect (EA), a professional industrial-strength UML CASE tool, as our frontend for metamodeling and specification of model transformations, as opposed to handcrafting a new editor with numerous editor frameworks such as the Graphical Modeling Framework (GMF).We leverage relevant Eclipse technologies and integrate the standard EMF codegenerator with CodeGen2, a model transformation engine used for generating method implementations from graph transformations.

Although we made major changes, we retained many design decisions and principles that MOFLON has always been known for. In the near future, we aim to support substituting the underlying model transformation engine with alternative engines. We plan to implement various extensions to Triple Graph Grammars and to our integration environment, and to provide a textual concrete syntax and textual editors as a free alternative to our EA frontend.

[1] www.moflon.org