

Declarative mapping between concrete and abstract syntax of domain-specific visual languages¹

István Ráth

Nowadays, the relevance of domain-specific visual languages is rapidly increasing due to the fact that they enable engineers to better concentrate on the problem itself.

In this paper, I present the VIATRADSM framework, a tool developed by Dávid Vágó and myself at the Department of Measurement and Information Systems of the Budapest University of Technology and Economics, which utilizes VIATRA2[2] to provide uniform support for creating editors, transformations, simulators and code generators for domain specific visual languages within the Eclipse framework.

The VIATRADSM framework is based on a plug-in architecture in order to enable the user to view the same modelspace from different domain-specific perspectives, which is an important advantage over current DSM implementations, since those tools focus on generating a separate editor program for each domain. VIATRADSM's tree view-based, syntax-driven editors can be constructed by simply specifying the domain metamodel (abstract syntax).

In most of the current DSM tools, such as MetaEdit+[3], concrete syntax representation is directly mapped to the abstract syntax, meaning that logical entities are always visualised as nodes, and logical relationships as edges. This is acceptable for simple languages, however our experience has shown that using more complex metamodels, especially those conceived for automated model transformations, not only results in visual models being too complicated to overview, but it can also drain system resources heavily.

Thus, modern approaches, such as Eclipse's Graphical Modeling Framework[4], employ a separate *visualisation metamodel*, which describes the structural appearance of diagrams. This technique allows the toolsmith to hide unnecessary detail, however it is still limited in the sense that classes can only be mapped to nodes and references to edges.

The declarative mapping technology presented in this paper, developed for the VIATRADSM framework, extends this idea by using VIATRA2's graph pattern matching engine to give the language engineer complete freedom to define how models are visualized. Thus, complex mappings such as aggregations can be easily defined using VIATRA2's native pattern description language (based on the Visual and Precise Metamodeling[1] language). The goal of the research is to provide full declarative support for specifying these mappings, meaning that language engineers should be able to construct visually appealing and effective tools without any manual coding.

References

- [1] D. Varró and A. Pataricza. VPM: A visual, precise and multilevel metamodeling framework for describing mathematical domains and UML (The Mathematics of Metamodeling is Metamodeling Mathematics), in: *Journal of Software and Systems Modeling*, October, 2003.
- [2] VIATRA2 Framework, *An Eclipse GMT Subproject*, <http://www.eclipse.org/gmt>.
- [3] MetaCase MetaEdit+, <http://www.metacase.com>
- [4] General Modeling Framework, <http://www.eclipse.org/gmf>

¹This work was partially supported by the SENSORIA European project (IST-3-016004).