

Generative Model Transformer : An Open Source MDA Tool Initiative

Jorn Bettin
SoftMetaWare
PO Box 617
Waiheke Island
Auckland, New Zealand

jorn.bettin@softmetaware.com

Ghica van Emde Boas
Bronstee Software & Services
Franz Lisztlaan 5
2102 CJ Heemstede
Netherlands

emdeboas@bronstee.com

E.D. Willink
Thales Research and
Technology Limited
Reading, England

ed.willink@thalesgroup.com

ABSTRACT

The Generative Model Transformer (GMT) project is an Open Source initiative to build a Model Driven Architecture™ tool that allows fully customisable Platform Independent Models, Platform Description Models, Texture Mappings, and Refinement Transformations. The project should result in (a) a tool that fulfils the MDA promise for faster/more accurate/better maintainable application development, (b) a tool for industrial use, and (c) MDA related research—which is encouraged and needed. A distinctive feature of GMT is the emphasis of model transformations as "first-class model citizens". The implementation of model transformations is envisaged to be in conformance with the future OMG modeling standard for Queries, Views, and Transformations (QVT).

Categories and Subject Descriptors

D.2.2 [Design Tools and Techniques]: Model Driven Architecture - *domain-specific architectures*.

General Terms

Design, Standardization, Languages.

Keywords

Model Driven Architecture (MDA), Generative Model Transformer (GMT), model transformation, QVT, domain-specific languages, Open Source.

1. INTRODUCTION

The GMT project [1] is the result of a BOF session following the OOPSLA'02 workshop *Generative Techniques in the Context of MDA* [2]. The goals of GMT are aligned with the OMG's objectives for MDA [3] and can be summarised as

- (a) facilitating a paradigm shift away from the current paradigm where "traditional" source code is the ultimate specification of software systems to a paradigm where "models and transformations between models" are the source

specifications of software and

- (b) making available an MDA platform that is fully open, allowing Platform Independent Models (PIMs), Platform Description Models (PDMs), and model mappings to be shared in a vendor-independent format—just as traditional text based source can be shared today.

GMT represents the interest of those who would like to see a much more widespread acceptance of model-driven and highly automated approaches in the software community, and who don't believe that this goal can be achieved with commercial MDA tools alone—especially since none of the current MDA tools provide the extensive support for fully customizable model transformations as envisaged by GMT. An open source platform should make MDA accessible and practical for the average software development organization.

GMT will provide a related set of components that can be used by other MDA tool components, and will consist of four main components:

- A mapping component that can combine two XMI-encoded models into one new XMI-encoded model.
- A model transformation component using XMI as input and output.
- A text generation component, using XMI as input and text (code) as output.
- A workflow component that provides the required glue between the three functional components above, any additional user-developed MDA tool components, and popular IDEs/tool platforms such as Eclipse.

2. THE MDA PROCESS WITH GMT

Any component that fulfils the basic requirement of allowing XMI input should be usable as a GMT tool component.

Figure 1 shows how the GMT tool might be used in practice. In the diagram we've accommodated the fact that a complex transformation may be better represented as a series of simpler transformations.

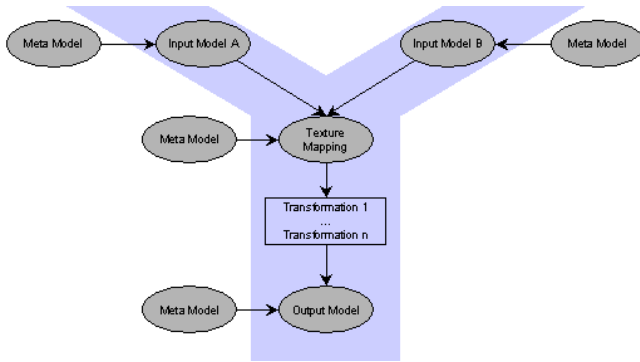


Figure 1

The terms *Platform Independent Model*, *Platform Specific Model*, and *Platform Description Model* are fundamental to the standard terminology used in the project. To differentiate between the loose specification of traditional software design patterns and the precise specification of patterns required for automation we use the term *texture* for "a pattern specification in an unambiguous UML-based format" [4]. Note that at each stage in the model transformation process one of the inputs is a set of textures—some textures may be very simple, others may be more complex. Thus it is appropriate to speak of *texture mappings*.

2.1 The Target Audience of GMT

The large breadth of the audience makes it necessary to define a tool architecture that is based on plug-ins and relies on XMI as the format for information exchange between tools.

GMT will be useful for those who need to

- apply MDA techniques for a specific application area (for example order management using relational database and JSP according to enterprise-specific infrastructure standards), i.e. Software Analysts and Software Architects with expert knowledge of the target domain and/or platform.
- develop real deployable applications, i.e. Software Developers with average knowledge and experience of the target domain and/or programming language.
- design domain specific languages for the purpose of significantly automating development of a product family in a specific domain, i.e. Language Designers with deep knowledge in specific vertical or horizontal domains.

2.2 Relationship to the OMG QVT RFP

There is a very strong relationship between the goals of the Queries, Views and Transformations (QVT) initiative [5] and the goals of the GMT project. Currently we are evaluating the initial submissions to the RFP [6], [7]. Although compatibility with QVT is a goal, in some areas GMT development may outpace the QVT standardisation process. This may mean that QVT compliance will have to be retrofitted into GMT.

As a side effect our project may help to raise the bar of a future OMG QVT standard, so that the commercial MDA vendors are forced towards a higher degree of inter-operability.

3. CURRENT PROJECT STATUS

The GMT project has just started, and we are currently concentrating on the design of the core architecture that will allow various MDA tool components to plug-in to GMT, such that GMT manages the workflow and co-ordinates model transformations across a chain of MDA components. By October 2003 we expect to be in a position to demonstrate the GMT backbone and initial prototype MDA tool components. In parallel with the practical work on the GMT tool, the project is fuelled by MDA-related research from a number of institutions [8], [9].

4. FURTHER WORK AND OUTLOOK

Significant work lies ahead in order to achieve the ambitious goals of the GMT initiative. The initial QVT submissions provide a useful starting point in terms of abstract and concrete syntax for model transformations, but only a tool that implements the suggested standards will be able to highlight deficits and practical usability issues.

We believe that development tools—including advanced MDA tools—are destined to go down the same path as operating systems, they become part of the basic infrastructure. We are convinced that the Open Source path provides the best route to achieve the paradigm shift away from textual source code to models and model transformations as the source specification for software.

5. REFERENCES

- [1] Generative Model Transformer project.
<http://www.eclipse.org/gmt/>
- [2] OOPSLA'02 workshop "Generative Techniques in the Context of MDA".
<http://www.softmetaware.com/oopsla2002/mda-workshop.html>.
- [3] OMG, Model Driven Architecture.
<http://www.omg.org/mda/>
- [4] Original GMT project proposal.
<http://www.softmetaware.com/oopsla2002/mdatoolproposal.pdf>.
- [5] OMG, "Request For Proposal: MOF 2.0/QVT", OMG Document, ad/2002-04-10.
- [6] DSTC, IBM, "MOF Query/Views/Transformations, Initial Submission",
<http://www.dstc.edu.au/Research/Projects/Pegamento/publications/ad-03-02-03.pdf>.
- [7] QVT Partners, "Initial submission for MOF 2.0 Query/Views/Transformations RFP",
<http://www.qvtp.org/downloads/1.0/qvtpartners1.0.pdf>.
- [8] Wei Zhao, B.R. Bryant, C.C. Burt, R.R. Raje, A.M. Olson. A Generative and Model Driven Framework for Automated Software Product Generation.
<http://www.csse.monash.edu.au/~hws/cgi-bin/CBSE6/Proceedings/proceedings.cgi>
- [9] E.D. Willink, UMLX : A graphical transformation language for MDA. (May 2003)