MDD with OMG Standards
MOF, OCL, QVT &
Graph Transformations

Andy Schürr
Real-Time System Group
Darmstadt University of Technology

andy.schuerr@es.tu-darmstadt.de

20th Feb. 2007, Trento
Outline of Presentation

• Languages and Tools for Model-Driven Development
  - OMG’s Model Driven Architecture (MDA)
  - Model-Driven Software Development (MDD)
  - MDD requirements derived from industrial case study

• From MDD to the World of Graph Transformations
  - Comparison of Meta-Case, Model/Graph Transformation Tools
  - MOFLON = OMG standards + graph transformation technology
  - MOFLON architecture and sublanguages

• … and Back Again
  - Status quo and future of MOFLON
  - Status quo of MDA/MDD/DSL/Meta-Case/… tools in general
Motivation
An Industrial Case Study

Magnetic Resonance Imaging System
- Real-Time
- Safety-Critical
- ...

- 3D volume scan
- Quickly evolving technology
  - Scan speed
  - Image resolution
- More clinical applications
  - Motionless tissues
  - […]
  - Heart surgery

- 3.5++ MLOC, 3 computers, 80 processes
- 200++ SW developers (engineers, scientists)
Source code organization

source code files ⇒ leaves of a building block hierarchy
A **Building Block** is a separated unit of

- Ownership and Responsibility
  - Software Architect for the System
  - Senior Designer for a Subsystem
  - …
- Product-related documents
  - Requirements Specification
  - Design & Interface Specification
  - Test Specification & Reports
  - …
- Functionally related code
- Encapsulation (information hiding)
- Hierarchy of Subblocks
Different Levels of Abstraction

High-Level Architecture
(Domain-Specific Language)

Consistency?

Low-Level Architecture
(Unified Modeling Language)

Consistency?

Implementation
(C-Code, ...)

Trento, 20th February 2007
OMG’s **Model Driven Architecture**

- **CIM** = Computer **Independent** Model
- **PIM** = Platform **Independent** Model
- **PSM** = Platform **Specific** Model
- **DSL** = Domain-**Specific** Language
- **MOF** = Meta **Object** Facility
- **UML** = Unified **Modeling** Language

OMG often recommends:

- DSL₁ = DSL₂ = DSL₃ = UML
OMG's Model Driven Architecture

(Meta-)Modeling Layers

Infrastructure = UML ∩ MOF
MDD/MDA Tool Requirements

Rapid Development (generating) of

- Local model analysis/transformation support
- Inter-model consistency checking
- Traceability link management support
- Bidirectional model update propagation
- Model import / export (code generators, parser, …)
- Tool wrappers for „COTS“ tools
- New tools for domain-specific languages
- Integrated model version management
- …
Categories of MDD Tools

- „Pure“ Model Transformation Tools
  - AMMA / ATL (INRIA)
  - ArcStyler (Interactive Objects Software GmbH)
  - ...

- „Pure“ DSL Editor Generators
  - Microsoft DSL
  - MetaEdit+ (MetaCase)
  - ...

- Integrated Approaches
  - GME (Vanderbilt University)
  - (OMG Standards)
  - MOFLON / Fujaba (Uni. Darmstadt, Paderborn, Kassel, …)
  - …
Model Transformation with AMMA

- Code Generator Def. Language
- Metamodel Def. Language (Abstract Syntax)
- Constraint Def. Language (Static Semantics)
- Model-to-Model Translation Def. Language
- Transformation Def. Language (Dyn. Semantics)
- GUI Def. Language (Concrete Syntax)
- ATL

TCS

Real-Time System Group

Trento, 20th February 2007
The Microsoft DSL Tools

- Code Generator
  - Def. Language

- Concrete Syntax
  - Def. Language

- Metamodel
  - Def. Language

- (API)
  - Analysis Report
    - Def. Language

- (API)
  - M2M Translation
    - Def. Language

- (API)
  - Transformation
    - Def. Language
Microsoft DSL - Metamodellierung
<shape name="FactoryShape" geometry="Rectangle">
  <decorators>
    <shapeText name="Name" position="Center"/>
    <expandCollapse position="InnerTopRight"/>
  </decorators>
  <fillColor color="DarkOrange"/>
  <outlineColor color="DarkOrange"/>
</shape>

<shapeMap>
  <class>DomainModel.SPL/Factory</class>
  <mc1CollectionExpression>
    <role>DomainModel.SPL/SimplePatternModel/PatternElements</role>
  </mc1CollectionExpression>
  <shape>Designer.SPLDiagram/Shapes/FactoryShape</shape>
  <textMaps>
    <textDecorator>Designer.SPLDiagram/Shapes/FactoryShape/Decorators/Name</textDecorator>
    <valueExpression>
      <valueProperty>DomainModel.SPL/Factory/Name</valueProperty>
    </valueExpression>
  </textMaps>
</shapeMap>
The Meta-CASE Tool MetaEdit+

- Code Generator
  Def. Language

- Concrete Syntax
  Def. Language

- Metamodel
  Def. Language

- Analysis Report
  Def. Language

- (API)
  Transformation
  Def. Language

- (API)
  M2M Translation
  Def. Language
Meta-Edit+ DSL Definition

Abstract Syntax Definition

Concrete Representation Definition

Class

Attribute
The Meta-CASE Tool GME

- Code Generator
  - Def. Language

- GUI
  - Def. Language
    - (Concrete Syntax)

- Metamodel
  - Def. Language
    - (Abstract Syntax)

- Constraint
  - Def. Language
    - (Static Semantics)

- Model-to-Model Translation
  - Def. Language

- Transformation
  - Def. Language
    - (Dyn. Semantics)

- GReAT
Elements of OMG´s MDA World

- Model to Text Transformation
  - RFP

- MOF View to Diagram
  - RFP

- MOF 2.0
  - Metamodel
  - Def. Language

- OCL 2.0
  - Constraint
  - Def. Language

- QVT 1.0
  - M2M Translation
  - Def. Language

- (QVT 1.0)
  - Transformation
  - Def. Language
### MDD / DSL Tools - Summary

<table>
<thead>
<tr>
<th></th>
<th>OMG Languages</th>
<th>AMMA (INRIA)</th>
<th>MS DSL (Microsoft)</th>
<th>GME (Vanderbilt)</th>
<th>...</th>
<th>Fujaba (UPB, UKa)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metamodel Def. Lang.</td>
<td>MOF</td>
<td>KM3</td>
<td>+</td>
<td>GME 5.0</td>
<td></td>
<td>UML 1.x</td>
</tr>
<tr>
<td>GUI Def. Lang.</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>GME 5.0</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>Constraint Def. Lang.</td>
<td>OCL</td>
<td>ATL / OCL</td>
<td>-</td>
<td>-</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>Model Trafo Def. Lang.</td>
<td>QVT¹</td>
<td>ATL</td>
<td>-</td>
<td>GReAT</td>
<td></td>
<td>SDM</td>
</tr>
<tr>
<td>M2M Trans. Def. Lang.</td>
<td>QVT</td>
<td>ATL / AMW²</td>
<td>-</td>
<td>GReAT²</td>
<td></td>
<td>TGG</td>
</tr>
<tr>
<td>Code Gen. Def. Lang.</td>
<td>-</td>
<td>TCS</td>
<td>+</td>
<td>-</td>
<td></td>
<td>Velocity</td>
</tr>
</tbody>
</table>

1: QVT has been designed for model-to-model translation purposes
2: ATL and GReAT are unidirectional model translation languages
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MOF</td>
<td>-</td>
<td>OCL</td>
<td>-</td>
<td>QVT</td>
<td>-</td>
<td>UML 1.x</td>
<td>???</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>???</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>???</td>
<td>SDM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>QVT</td>
<td>TGG</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Velocity</td>
</tr>
</tbody>
</table>
OMG Standard + Graph Transformation

- **Velocity, XSLT**
  - Codegenerierung

- **DiaMeta**
  - Editorgenerierung

- **MOF 2.0**
  - Metamodel
  - Def. Language

- **OCL 2.0**
  - Constraint
  - Def. Language

- **TGG (QVT Subset)**
  - M2M Translation
  - Def. Language

- **SDM**
  - Fujaba Graph Transformations

- **University of Techn. Dresden**
- **University of Techn. Darmstadt**
- **University of Kassel**
- **University of Paderborn**
- **University of BW Munich**

Trento, 20th February 2007

Real-Time System Group
Back to the Running Example
Simplified Running Example

High-Level Architecture
(Domain-Specific Language)

Consistency

Low-Level Architecture
(Unified Modeling Language)

Consistency

Implementation - PSM
(C-Code, … )
Forward Transformation Scenario

Diagram:
- Model Representation
  - Model Parser
  - High-Level ADL Objects
    - Model Translator
    - Low-Level ADL Objects
      - Code Generator
      - Code Fragments
      - Velocity / XSLT Templates
      - MOF Meta Model
    - MOF Meta Model
    - Analysis Results
  - MOF Meta Model
  - Model Analyser
  - OCL Constraints
  - SDM Graph Transformation
  - TGG (QVT-like) Translations

Legend:
- TGG (QVT-like)
- EBNF / XSLT
- MOF
- OCL
- SDM Graph Transformation
- TGG (QVT-like) Translations

Trento, 20th February 2007
Meta-Modeling with MOF 2.0 - 1

ADL DSL Meta-Model
Implementation Meta-Model

contains association:

• “heavy-weight“ association (implemented as relation)
• association owns association ends (and not class)
• association ends are not navigable (from classes)
• …
Unique Features of MOF 2.0

- Different sorts of associations
  - pointers versus real associations
  - navigability (for API method selection)
  - ownership of association ends (for DBMS schemata)

- Excellent support for model refinement
  - class inheritance hierarchies
  - refinement of associations (subset, redefines, …)
  - refinement of packages

- Powerful modularization concepts
  - hierarchies of packages
  - import/export relationships
  - merging of packages
We added 86 constraints

From 90 MOF constraints are

- 48 (53%) correct
- 42 (47%) erroneous

We added 86 constraints

- 50 (50%) additional
- 51 (50%) modified

Reasons for 51 bug fixes are

- 16 (29%) erroneous metamodel reference
- 19 (34%) erroneous semantics
- 21 (37%) Erroneous syntax
Forward Transformation Scenario

MOF Meta Model

Analysis Results

Model Analyser

OCL Constraints

SDM Graph Query

High-Level ADL Objects

Low-Level ADL Objects

MOF Meta Model

Scenario

ADL Objects

MOF Meta Model

ADL Objects

High-Level ADL Objects

Low-Level ADL Objects

OCL Constraints

SDM Graph Query

Model Analyser

Analysis Results

MOF Meta Model
Consistency rules for ADL

- A block uses interfaces that are required interfaces
- A block uses blocks that own required interfaces
Constraint Definition with Graph Query
Forward Transformation Scenario

- MOF Meta Model
- Analysis Results
- OCL Constraints
- SDM Graph Query
- SDM Graph Transformation

1. High-Level ADL Objects
2. Model Translator
3. Low-Level ADL Objects
4. MOF Meta Model

- Model Analyser
- OCL Constraints
- SDM Graph Query
- SDM Graph Transformation
Forward Transformation - Context

source model

traceability link

target model

performForwardTransformation(RefObject)
Created target model extension (plus new traceability link)
• Needed model transformation rule sets:
  ➢ forward transformation
  ➢ backward transformation
  ➢ create traceability links only
  ➢ check traceability link consistency
  ➢ forward/backward attribute propagation
  ➢ remove traceability links
  ➢ forward/backward deletion propagation
  ➢ …

• Generate all transformation rule sets from single declarative bidirectional model integration rules
  ➢ QVT core/relational = Triple Graph Grammars (TGG)
Forward Transformation Scenario

High-Level ADL Objects

Model Translator

Low-Level ADL Objects

MOF Meta Model

Analysis Results

Model Analyser

OCL Constraints

SDM Graph Query

SDM Graph Transformation

TGG (QVT-like) Translations

generates
TGG (QVT-like) Specification

Declaration of Traceability Relationships (Mappings) with associated bidirectional translation rules
TGG (QVT-like) Rule for Blocks

Architecture model

File system model
TGG (QVT-like) Rule for Subblocks

Architecture model

Inherited rule pattern

File system model
Generated Forward Transformation

```plaintext
SubBlockDirectory::performForwardTransformation(inputObject, RefObject): Void

parentBlock::Block

blockdirectory::BlockDirectory

parentDir::Directory

directory::Directory

srcDir::Directory

name = "src"

inDir::Directory

name = "inc"

containingBlock

contains

entailsBlock

containsBlock

containsArtifact
```

Trento, 20th February 2007

Real-Time System Group
Summary of MOFLON MDD World

- Text Templates
- MOF Meta Model
- Analysis Results

- Model Representation
- Model Parser & Generator
- High-Level ADL Objects
- Model Transformer
- Model Analyser
- Model Editor

- Code Fragments
- Code Parser & Generator
- Low-Level ADL Objects

- Representation Grammar
- Velocity / XSLT Templates
- MOF Meta Model

- OCL Constraints
- SDM Graph Transformation
- DiaMeta Ed. Specification
- TGG (QVT-like) Translations

Trento, 20th February 2007

Real-Time System Group
Conclusions
• Model-Driven Development (MDD) is a “hot topic” of the Software Engineering Community
  ➢ with all the resulting pros and cons
  ➢ …

• MDD combines established technology
  ➢ meta-modeling / meta-case tool technology
  ➢ compiler compiler technology
  ➢ …

• Currently available (commercial / academic) MDD tools
  ➢ support only subsets of all MDD activities
  ➢ lack precise definition (available for graph transformations)
  ➢ …
OMG Standard + Graph Transformation

- **Velocity, XSLT**
  - Codegeneratorierung

- **DiaMeta**
  - Editorgeneratorierung

- **MOF 2.0**
  - Metamodel
  - Def. Language

- **OCL 2.0**
  - Constraint
  - Def. Language

- **TGG (QVT Subset)**
  - M2M Translation
  - Def. Language

- **SDM**
  - Fujaba Graph Transformations

- University of Technology Dresden
- University of Technology Darmstadt
- University of Kassel
- University of Paderborn
- University of BW Munich

Trento, 20th February 2007
• system engineering tool integration  
  (ToolNet project with DaimlerChrysler et al.)
• model analysis / design guideline checking  
  (MATE project with DaimlerChrysler et al.)
• software analysis / reverse engineering  
  (based on experiences at Philips Medical Research)
• visual DSL editor development  
  (ECLIPSE plug-ins in cooperation with UniBw)
• …
• Metamodelling with MOF 2.0
  - missing UML concepts (association classes)
  - integration with UML profile definition

• Constraint Definition with OCL 2.0
  - incremental (event-driven) constraint checking
  - integration with transactions & repair actions

• Local Model Transformations with SDM
  - handling of composition hierarchies (still a problem!)
  - integrated formal definition of language mix

• Model To Model Transformations with TGGs
  - merging TGGs with QVT Relational
  - …

• Integration with Editor Generator Framework DIAMETA
Model-Driven Development with OMG Standards
Graph Transformations

Questions?

Version 1.0
(Queen)

Download/Feedback: http://www.moflon.org/