Building an MDA Tool Chain

Greg Eakman

Pathfinder Solutions

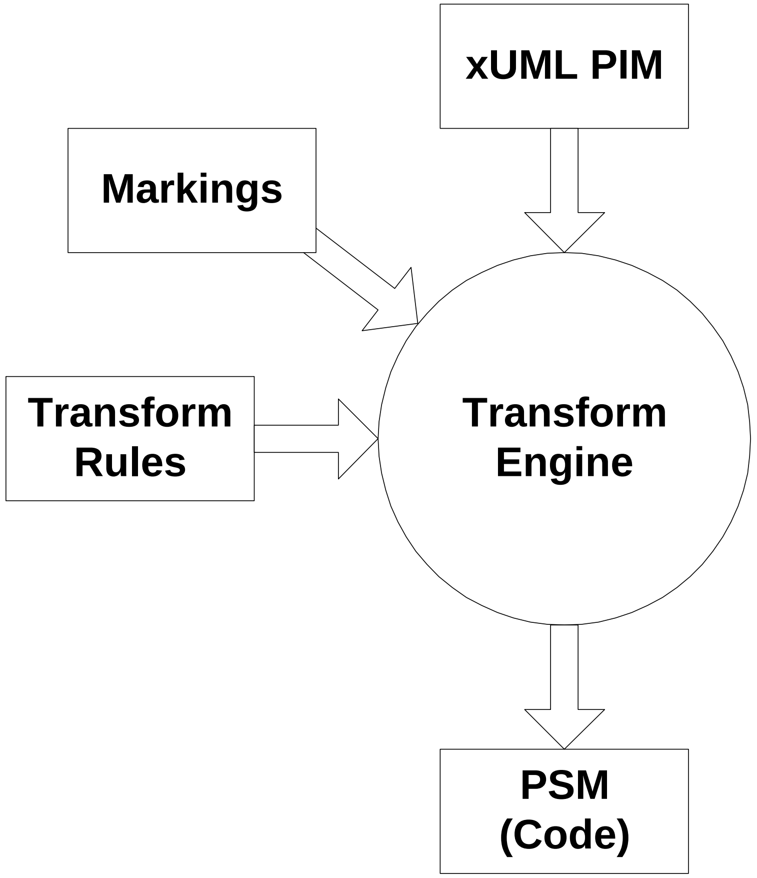
grege@pathfindermda.com

Agenda

- Model Driven Architecture
- PathMATE MDA tool chain
- Tool chain integration points
- Future directions

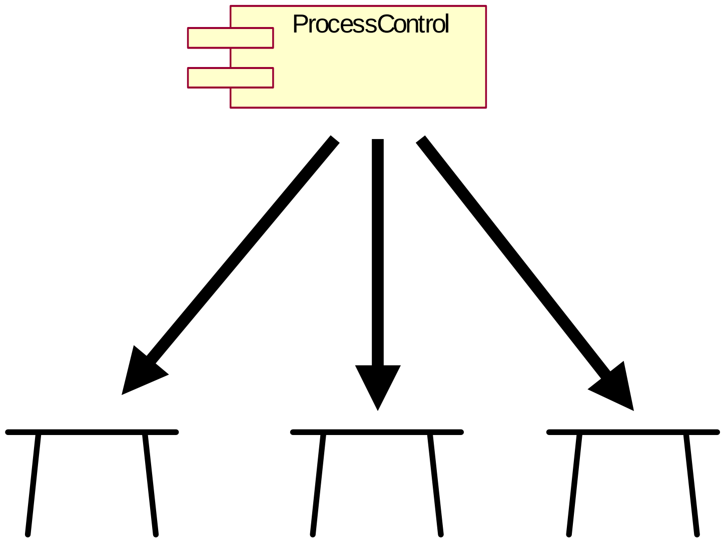
Model Driven Architecture

- Model components independent of implementation technologies
- Transformations map model to implementation
- Generates any type of text file
  - Code (C, C++, Java)
  - XML, XML Schemas
  - RDB Schemas
  - Build scripts
  - HTML



Embedded MDA

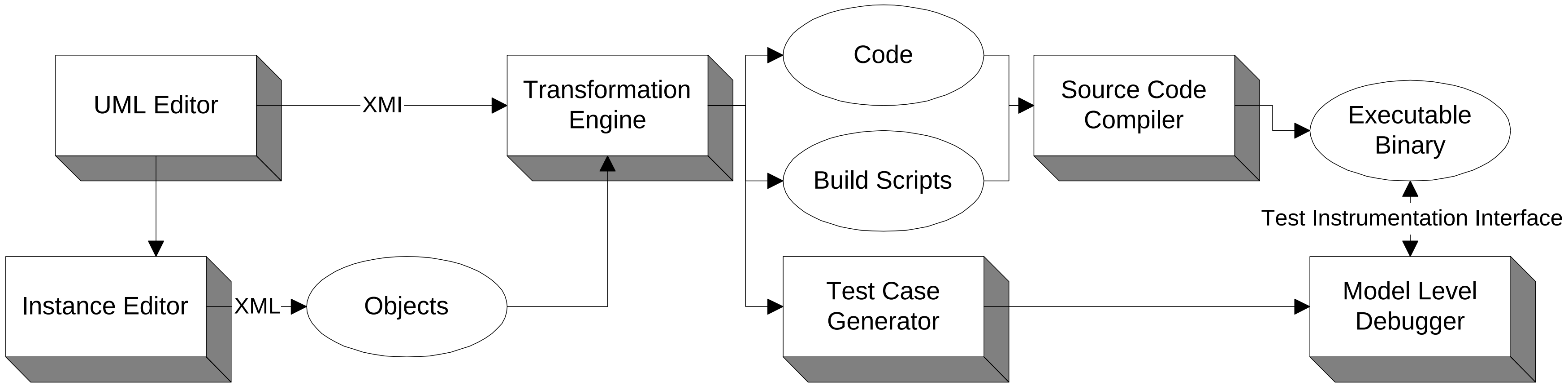
- Executable models
- Large number of embedded systems platforms
- Variations in development processes and tool requirements



Tool Integration

- Editors
  - XMI Transformation Engine
  - Interactive Requirements
- RTOS
- Build Environments

PathMATE Tool Chain



Current Approach

- Custom integration with each tool
- Templates map models to downstream target tool input format
- Upstream inputs require custom parsers and transformations

Transformations

- Domain specific language for converting model data to text
- Executed by the transformation engine
- Operate on UML metamodels natively
- Extensions for custom metamodels

Transformation Engine

- Proprietary metamodel based on UML
- Operates on UML data (MOF level 2)
- Transformation template script
- Template language extended to handle instance data (MOF level 0)
- Current output textual only (not a QVT transform)

UML Editor Integration

- PathMATE profile
- Connection to transformation engine
- Interactive requirements
- Action language
- Rename

XMI

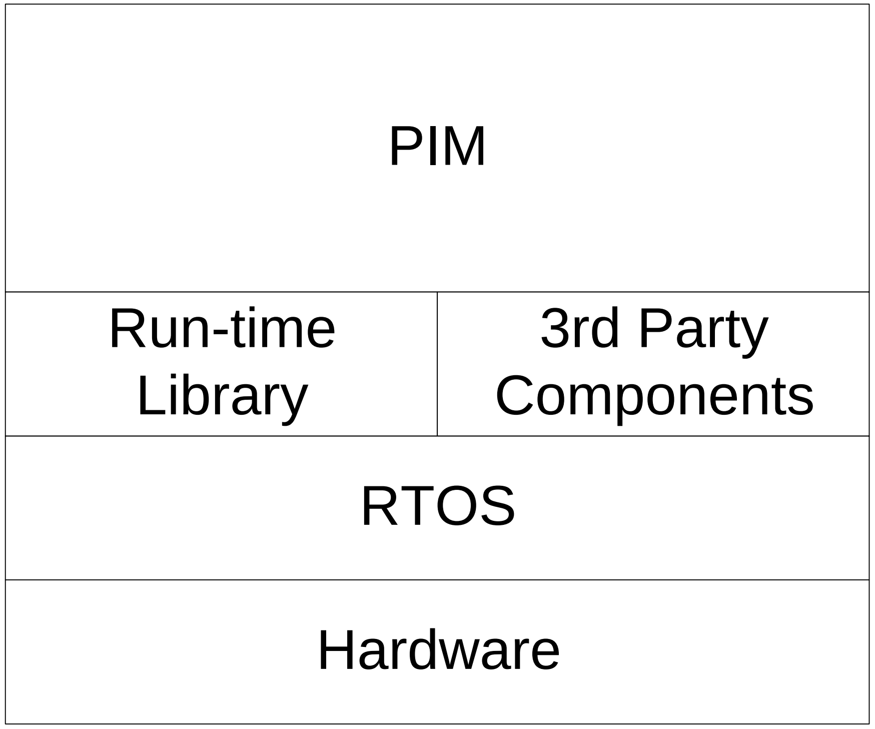
- Not quite ready as an interoperability standard
- Initial round trip via XMI lost data
- Wrote own XMI exporter for editors
- Efforts to tighten XMI in progress

Build Environments

- IDE Project files
- Makefiles
- Board specific files

RTOS and Other Components

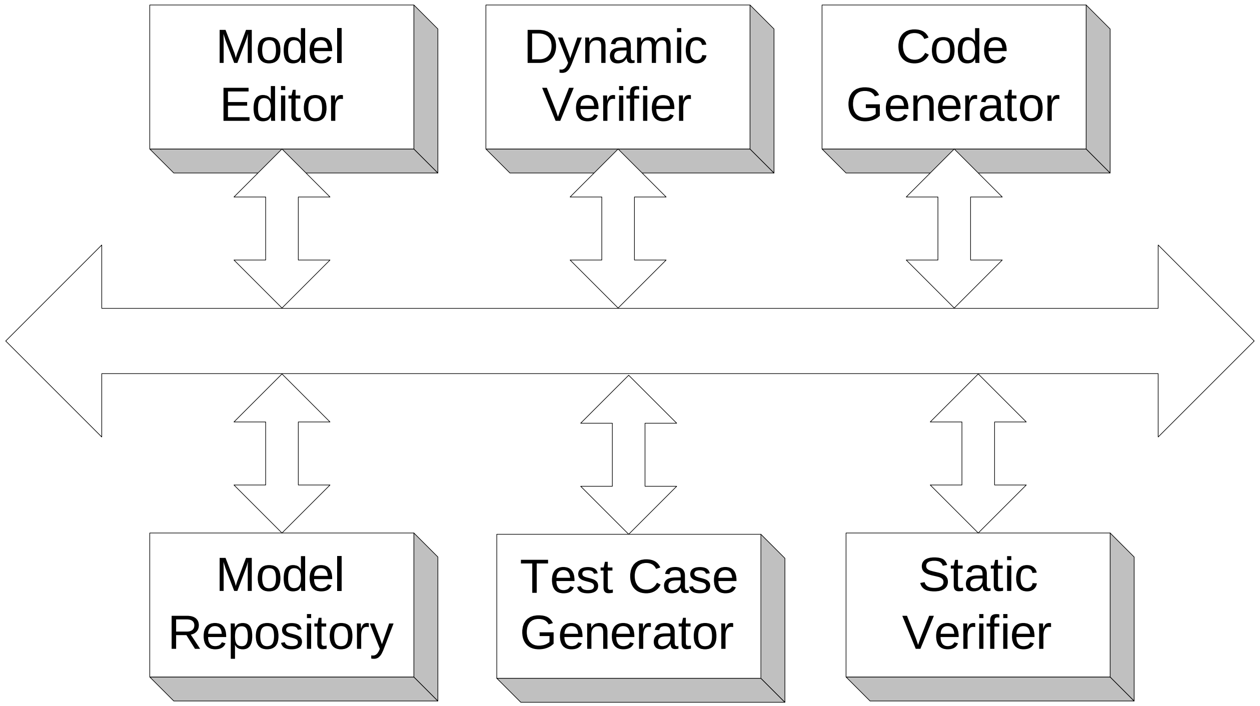
- Run-time library wraps common RTOS features
  - Threads
  - Memory management
- Transformation rules map to custom features



Instance Data

- Instance data (MOF level 0)
- Instance editor
  - Instances
  - Attribute values
  - Associations
- XML Schema
  - XMI like
  - Rename of meta-model elements

MDA Tool Chain



MDA Tool Chain Obstacles

- Tools use different models or semantics
  - Informally defined or proprietary profiles
  - Executable UML
  - Ambiguity intentionally left in UML specification
- Assume different platform semantics
- Tool input and output formats

Query View and Transformation

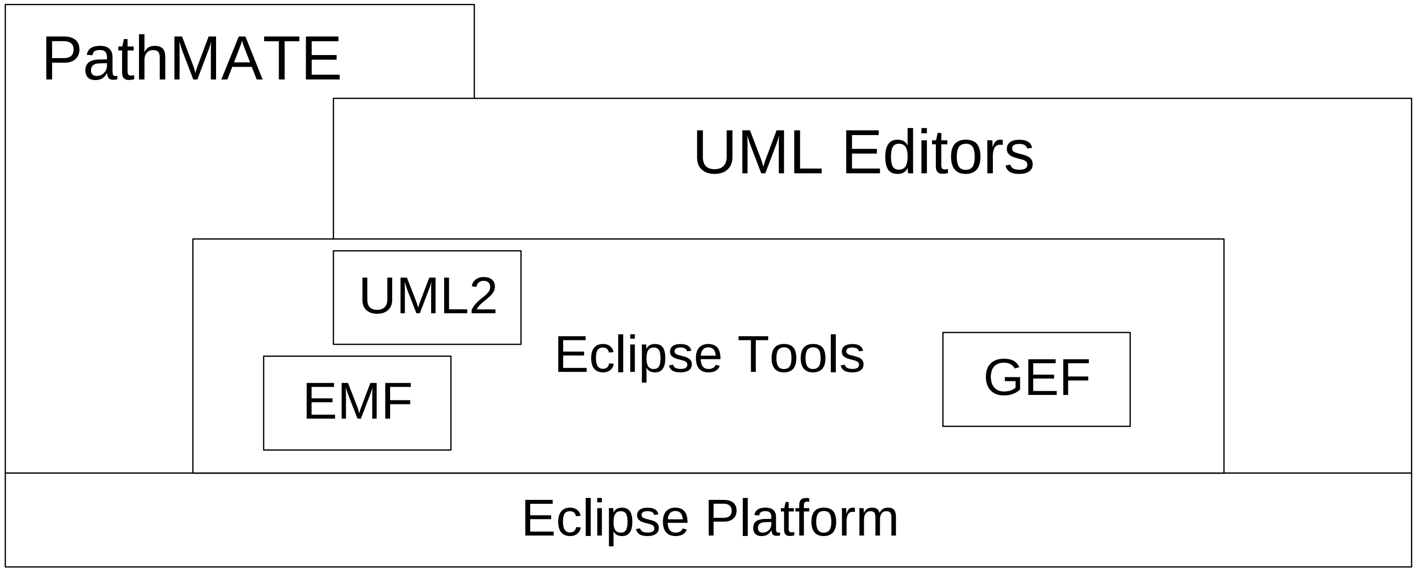
- Standard way to specify transformations
- Model to model transformations
- Model to text
- Works in progress

Eclipse

Pathfinder Solutions

- Common backplane EMF
- Plug-in architecture
- Solves some of our editor integration problems
  - UML2 project wraps XMI
  - Events connection to EMF for interactive change notification
  - Fewer editor specific integrations
  - Extendability

Eclipse


PathMATE
UML Editors
UML2
EMF
Eclipse Tools
GEF
Eclipse Platform

Conclusion

- MDA tool chain could use MIC principles
- MIC could use MDA transformation principles
- Transformation is key area of overlap

Thank You!

Pathfinder Solutions