Which MDA Tools are Right for You?

Mike Rosen
CTO, M²VP
Mrosen@m2vp.com
Model Driven Architecture

- A set of standards defining the scope, contents, creation and usage of models
- An architecture-based method for integrating models into the development process
- Core Technologies
  - UML + OCL
  - MOF + XMI
  - CWM
Agenda

- MDA context
- Development process
- Where tools fit in the process
- Requirements for tools
- Integration between tools
MDA Distilled

Business Analyst

Business Model (CIM)

Platform Independent Analysis Model (PIM)

Architect / Designer

Platform Specific Design Model (PSM)

Developer / Tester

Code
MDA Mappings

Computation Independent Business Model

Platform Independent Analysis Model

PIM → PIM Mapping

PIM → PSM Mapping

Platform Specific Design Model

PSM → PSM Mapping

PSM → Code Mapping

Code
Technology Independence

- Applications are “Future-Proof” against technology churn
- When technology evolves, a new PSM can be generated rather than rewriting it
- Artifacts can be generated for multiple platforms from the same design

Business Analyst

Business Model

Platform Independent Model

- EJB 1.1 Design Model
- EJB 2.0 Design Model
- .NET Design Model

Architect / Designer

Developer
Generation Tools

- Tools are standards based, not proprietary
- Resulting code base doesn’t require a specific runtime infrastructure
- 70-80% of the structural code can be generated
- Test Cases can be generated from OCL
MDA Under the Hood

MDA explicitly supports Enterprise Architecture
1. Define the approach
   - Integrate enterprise architecture into the development process.
   - Create meta-models and profiles

2. Define the problem
   - Create Business Models (Domain, CIM, System)

3. Define the solution
   - Refine into PIMs and PSMs

4. Leverage the solution
   - Integrate assets into a reuse repository
   - Architecture and design accommodates: reuse, customization, enhancements, versioning…
MDA Profiles

- **Computational Independent Model**
  - Simplified UML subset appropriate for business analysts. Non-UML representation.

- **Platform Independent Model**
  - Custom profiles for enterprise architecture and standards
  - Standard based profiles (EDOC, EAI)

- **Platform Specific Model**
  - Standards based profiles (CORBA, EJB, .NET)
MDA Process Review

Business Model

Platform Independent Model

Platform Specific Model

Code, Test, Doc Generation

Model Transformation

Meta & Modeling

MDA Architect

Business Analyst

Application Architect

Developer / Tester

Code
Meta and Modeling Concerns

- Basic desktop tool capabilities
- Ability to create models and meta-models in UML
  - Formally define model constructs and constraints
- Ability to create profiles in UML
  - Since most modeling tools don’t support MOF directly, profiles extend standard UML tools
- Ability to import / export meta-models and models in standard XMI format
Transformation Concerns

- Ability to formally define mapping between models
- Ability to customize standard mappings
- Ability to parameterize mappings
- Ability to modify transformation result
- Ability to trace elements in one model to the elements they were derived from in another model
Generation Concerns

- Preserves separation of concerns
  - Platform specific info is not required in PIM
  - Platform independent info in not required in PSM
- Can generate to multiple technologies and platforms
- Generation can be parameterized and modified
- Generation based on standard and custom patterns
- Completeness
  - Structural code (skeletons, etc.)
  - Procedural code
  - OCL can be compiled, generate run-time code
- Can be integrated into an automatic build process (command driven)
More About Generation

- **Procedural code**
  - Some code can be generated from OCL, but
  - Requires support for Action Semantics
  - Generally necessitates reducing scope of target environment

- **Tests**
  - Can be generated from OCL and Action Language
  - Can be integrated into a framework (e.g. JUNIT, JRUN)

- **Documentation**
  - Ability to create analysis and design documentation from information in all levels of models
  - Ability to customized document and report generation
  - Ability to create some form of user documentation
Advanced Capabilities

- Model correctness can be validated against meta-model
- Model can execute in a simulation environment
- Support for versioning of models
- Support for model synchronization
  - Reverse or Round-Trip engineering
- MOF Repository
Metadata in the MDA Process

- Business Model
- Platform Independent Model
- Platform Specific Model

Profiles → Mappings → CIMs PIMs PSMs
Profiles → Mappings → Asset Descriptors
Profiles → Mappings → Deploy. Descr. + Declarative
Profiles → Mappings → Source Code
Profiles → Mappings → Docs

MOF
MOF/UML
UML/RAS
XML
scms
dms
Tools in the MDA Process

- Business Analyst
  - Business Model
  - Modeling, Validation & Simulation
  - Business Model

- Application Architect
  - Platform Independent Model

- Model Transformation
  - Platform Specific Model

- Developer / Tester
  - Code Generation
  - Test Generation
  - Code

- Documentation Generation
  - Profiles & Mappings

- MDA Architect
  - MetaModeling
  - IDE

- Repositories + Metadata Management

MOF
MOF/UML
UML/RAS
XML
scms
dms

Mo2VP
MDA Tool Integration

Business Modeling Tool

Software Modeling Tool

Platform Independent Model

Transformation Tool

Platform Specific Model

Code and Test Generation Tool

Code

Repository and Metadata Mgmt. Tool

XMI

XMI

XMI

XMI

XMI

Business Model

Business Analyst

Application Architect

Developer / Tester

Meta-Modeling Tool

MDA Architect

Copyright © M²VP Inc, 2003. All rights reserved
Hard Criteria

- Meets *YOUR* criteria
- Addresses Full Lifecycle
  - Meta-models, transformations, generation
- Generation
  - SOC, Customization, Completeness
- Advanced Capabilities
  - Synchronization, Versioning, Validation, Simulation, MOF
- Integration with other tools
  - Modeling
  - IDE
  - Repository
- Standards Conformance
Soft Criteria

- Cost
- Product Architecture
  - Evolvability with the market
  - Support for future platforms
- Documentation, samples, training
- Ease of Use
- Company
  - Support, enhancements
  - Future looks bright
  - Product line is integral
  - Easy to work with
Questions for Your Enterprise

- What current tools and processes does MDA have to integrate with or support for development, test, reuse, documentation, etc.
- What will the MDA development lifecycle be in the organization?
- Who will perform the business modeling?
- What is the enterprise and application architecture?
- Can these be supported by standard profiles or will the organization be creating custom profiles and metamodels?
- What technology platforms will need to be supported?
The Bottom Line

- For MDA to deliver value
  - It must make models *first class development artifacts*
  - Tools must support all aspects
    - IDE for modeling
Conclusion

- No one tool does it all!
- Enterprises have different requirements
- Integration and customization important
- Standards and tools are critical to MDA success
- Lot’s of advancements in the future
  - Action Semantics, MOF 2.0, QVT, RT Engineering
- The market is still young. Expect…
- Continued expansion of MDA tools…
- … and then contraction of market
Questions