Model Driven Architecture: Concepts and Practice

Alan W. Brown

IBM Distinguished Engineer
Manager, MDD Strategy

awbrown@us.ibm.com
Agenda

- Models, Model Driven Development, and MDA
- IBM’s MDA Solutions Today
- Future Directions
- Summary
Agenda

- Models, Model Driven Development, and MDA
- IBM’s MDA Solutions Today
- Future Directions
- Summary
How Do Customers Use Models?

- **Code only**
  - M
  - C
  - “What’s a Model?”

- **Code Viz.**
  - M
  - C
  - “The code is the model”
  - visualize

- **RTE**
  - M
  - C
  - “Manage code and model”
  - synchronize

- **Model-centric**
  - M
  - C
  - “The model is the code”
  - generate

- **Model only**
  - M
  - C
  - “Let’s talk models”
Analysis, Design & Construction Solutions

*Adopt the right development style for your needs*

**Supports spectrum of development requirements**
- Development languages
- Operating environments
- Skill levels
- Modeling paradigms
A Solution for Managing IT Complexity: Model Driven Architecture (MDA)™

- An integration of best practices in Modeling, Middleware, Metadata, and Software Architecture

- Model Driven (UML, MOF, CWM…)
  - Platform Independent Models (PIM) – Technology or increasingly Business Models
  - Platform Specific Models (PSM) - J2EE, .Net, SQL
  - Mappings: PIM<->PIM, PSM<->PSM, PIM<->PSM
  - Applies across the software life cycle

- Key Benefits
  - Improved Productivity for Architects, Designers, Developers and Administrators
  - Lower cost of Application Development and Management
  - Enhanced Portability and Interoperability
  - Business Models and Technologies evolve at their own pace on platform(s) of choice

Source: OMG

MDA is a **framework** and a **process**
MDA as an ‘Architectural Style’ for Development & Integration

- **Understand** the problem domain (technology or business)
- **Model** the problem domain
- Formally represent the **models and metadata**
- Use Standard transformation (mappings & patterns)
- Use open source modeling frameworks for **model integration and management**
Agenda

- Models, Model Driven Development, and MDA
- IBM’s MDA Solutions Today
- Future Directions
- Summary
Three Primary Scenarios for MDA

- Model-driven Business Integration
  - Connecting business, IT, and operations

- Model-driven RAD for developers
  - Leveraging models for code generation

- Model-driven Design and Construction
  - Creating DSLs and domain-specific transformations
Model-driven Business Integration

Model Business Process
- Document and specify As-Is Process with Metrics
- Specify and construct goals, objectives and requirements
- Apply Technology to Improve the Process
- Model the To-Be Process

Understand Business & Elicit Requirements
- Model and simulate business processes
- Model Applications and Data
- Analyze the financials & prioritize the areas that bring maximum business value

Monitor & Manage
- Audit processes and improvements
- Make Iterative Improvements
- Model the Next As-Is and To-Be Process

Discover & Design
- Harvest existing assets
- Identify / prepare existing assets or reuse
- Design system architecture

Test & Deploy
- Manage testing, requirements, configuration, and project management

IBM Rational Unified Process
- Develop Iteratively
- Focus on Architecture
- Continuously Ensure Quality
- Manage Change & Assets
- Transform, Integrate & Build
  - Rapid integration and/or app development
  - Visual construction and programmatic code generation
  - Functional and load testing
  - Generate XML code & manage UML blueprints & Automated Workflow
  - Apply Patterns to Accelerate Development
An Example of Model-Driven Business Integration Today (Travel & Transportation Scenario)

- Paint the vision of the future of business operations with a Role Based Business Performance Management Demo
- Document and analyze the business operations Business Process Models with Metrics
- Generate the Financial Analysis, Determine Where to Focus & Build the Business Case
- Visual construction and programmatic code generation
- Monitor the Results Against Original Plan & Iteratively Improve
- Repair or Improve a new process
- Apply Patterns to Accelerate Development
- Generate XML code & manage UML blueprints
Model-driven RAD for Developers

- Leverage Rational Visualization expertise to improve WSAD developer experience
- Create and populate diagrams from WSAD Java/J2EE views
- Edit EJBs and Java classes
- Expand and elide relationships on diagrams
- Navigate from graphical views to WSAD text-based editors

Implementation-level modeling for creation and visualization
Packaged Technology-specific Transformations -1

Generate Portlets by Wizard

Test and Debug In Test Environment

Build and Customize
## Packaged Technology-specific Transformations -2

<table>
<thead>
<tr>
<th>Page</th>
<th>Transaction</th>
<th>Entity</th>
<th>Database</th>
</tr>
</thead>
<tbody>
<tr>
<td>JSP Page</td>
<td>Session Bean</td>
<td>Entity Bean or Entity Class</td>
<td>Stored Procedures or Dynamic SQL</td>
</tr>
<tr>
<td>Servlet or Session Bean</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Portlet Controller</td>
<td>JSP</td>
<td>Entity Bean or Entity Class</td>
<td>Stored Procedures or Dynamic SQL</td>
</tr>
<tr>
<td>Message</td>
<td>Message Session Bean, Msg Driven Bean, or Msg Servlet</td>
<td>Entity Bean or Entity Class</td>
<td>Stored Procedures or Dynamic SQL</td>
</tr>
<tr>
<td>Web Service</td>
<td>WS Session Bean</td>
<td>Entity Bean or Entity Class</td>
<td>Stored Procedures or Dynamic SQL</td>
</tr>
</tbody>
</table>
Model-based Information Generated at Runtime

- Perform runtime analysis within IBM WebSphere Studio and Microsoft Visual Studio .NET
- Record application execution details in real-time
- Analyze applications with or without debugger
- Selectively instrument managed languages (Java, C#, MS Visual Basic .NET)
Model-driven Design and Construction

Abstraction
Model-driven Design and Construction

Model drivers (semantic connections between models). These are candidates for automation.
The MDA Toolkit Overview

- A toolkit for the creation of MDA style transformations that involve UML models and IBM Rational XDE for Java

- Appropriate for large scale transformation (entire models to models) that are algorithmically complex

- Not a replacement for patterns
  - Patterns are most appropriate for small declarative transformations that are used tactically by developers

- A set of components for the IBM Rational XDE for Java environment that compliment the existing set of features and functionality common used in MDA style development processes

- Delivered as an Eclipse plugin itself
  - Takes advantage of the Eclipse Update/Install features

MDA Toolkit Elements

- Transformation project creation wizard (with sample code)
- Profile tools for working with UML profiles
- An API built on top of the RXE, XDK, Code Templates, Patterns and Java XDE. Includes commonly used MDA like functionality (i.e. deep copy)
- Documentation
- Reference Examples
  - UX to JSF
  - UX to Struts
  - Analysis Model to Hibernate Design
MDA Toolkit Usage

- A project’s technical experts develop the transformations
  - Define appropriate UML profile using profile tool
  - Implement the “transform” method
  - Toolkit manages creating the transforms and its UI
  - Delivers the transform as an Eclipse plugin

- Eclipse plugin is installed by the project
  - Uses standard Eclipse mechanisms
  - Creates new menu items
  - Transforms are applied by developers
Overview of the MDA process -1

- Examine the models currently being used or expected to be used in the process

- Closely examine the semantic connections between elements in one model to another

- Identify candidate transformations for automation
  - Good transformations can be clearly articulated (can be simple or complex)
  - Model to Model
  - Model to Code

- Specify (document) the transformation requirements
  - Input and output parameters
  - Model markings
  - Mapping rules (procedure and declarative)
Overview of the MDA process -2

- Create necessary XDE/UML profiles
- Develop the transformations
  - Model to Model (MDA Architect Plugin and API)
  - Model to Code (Code Templates)
  - Model to Artifact (custom Java code)
- Draft usage documents for development team using the transformations
- Package and deploy the transformations as normal Eclipse plugins
Lessons in the Practice of MDA

- Focus needs to be on productivity and predictability of running systems
  - Its about code generation

- ....but also focus on Integration, communication, and repeatability
  - Its about much more than code generation

- Don’t get confused between models, modeling, and MDA
  - Many approaches to capture and automate knowledge
  - Appropriate practices are needed in support of these

- Changing mindset to a model-drive approach takes time
  - Introduce modeling “by all means necessary”!
  - Provide the support structure necessary for all reuse approaches
Agenda

- Goals and Vision
- Industry Trends and Directions
- IBM’s MDD Solution Today
- Future Directions
- Summary
Model Driven Development Strategic Themes for 2004 - 2005

**Broaden Appeal**
- Reach Architects who have not embraced formal modeling
- Enable Architects who have embraced formal modeling and model driven development
- Accelerate the “Maintenance of Existing Value” and the “Development of New Business Value” through automations

**Raise Productivity**
- Simplify model driven workflows
- Integrated process guidance and tool assistance
- User assistance model that assists the user in developing the appropriate skills and expertise

**Extend Integration**
- Deepen integrations with change management and configuration management solutions
- Extensibility framework to support customization and extensibility by customers and partners
- Traceability from requirements thru analysis and design

**Maintain Standards Concurrency**
- Compliance with the latest standards and platform technologies
  - UML, RAS, J2EE, Web Services, JSP, JSF, SDO, XML
Direction: Integrated, Role-based Tools Portfolio

- Analyst
  - Model, simulate and monitor business operations

- Architect
  - Model applications and data

- Developer
  - Visually construct, program, and generate code

- Tester
  - Design, create, and execute tests

- Operations Manager
  - Deployment and Management (Bus. and IT)

- Analyst
  - Business Performance Monitoring and Analysis

Eclipse Platform, EMF (UML, J2EE, Web Services…) models

Team Unifying Platform

Requirements Management, Test Management, Change Management
Software Configuration Management, Rational Unified Process
Future Research Innovations and Product Ideas
Agenda

- Goals and Vision
- Models, Model Driven Development, and MDA
- IBM’s MDA Solutions Today
- Future Directions
- Summary
Summary

✓ Provide an end-to-end modeling solution that spans business, data, user interaction, and software architecture that’s traceable from design through deployment

✓ Blend RAD, MDA and IDE tools to empower developers at all skill levels with model-driven architecture

✓ Deliver “just in time, just enough” functionality to every team member

✓ Make reuse practical with asset-based development

Accelerating application deployment and integration