An SOA Solution Driven from Business Process Models using MDA: Methods and Lessons Learned

INHERIT, LLC
Robert Mitchell
Agenda

- Provide an Overview of Our Background
- Describe SOA-BPM Project Using MDA
- Our MDA Approach
- Hurdles to Overcome
- Lessons Learned
Our Background

- **1997 - We founded INHERIT, LLC**
  - Since its inception we were focused on driving software development from UML models

- **1999 – Developed initial version of our toolset**
  - Transformation engine built upon a declarative rules engine

- **2001 – OMG Coins term “MDA”**
  - We realized that OMG’s definition of MDA was very much in line with the software development approach we were pursuing
    - *Good news and bad news*

- **2003 – Developed MDA Express**
Current Version of MDA Express

- Our transformation engine is called MDA Express
  - Provides a flexible environment that supports complete customization of architectural frameworks
  - Supports software development in any language
  - Provides multiple ways to preserve changes to generated artifacts during re-generation
  - Captures and incorporates system details from all types of UML Diagrams
  - Supports a combination of declarative rules that then execute a series of imperative statements
  - Works with Eclipse Modeling Framework (EMF) and Net Beans Meta Data Repository (MDR)
Applicable Experience

• **Modeling Business Processes**
  – Exposed to business process modeling mid 1990’s
  – Applied consistently on all of our projects
  – Formalized on BPM over the past two years

• **Applying SOA Concepts**
  – Began developing projects using elements of SOA in 2002
  – Currently working on very large scale SOA applications

• **Applying “MDA” Concepts**
  – Have developed numerous MDA projects since 1998
Proof-of-Concept Project

- 2006, January – asked to use MDA to develop a pilot project
  - We recognized that there would be business process and workflow elements to this project
    - Good candidate for developing business process models
    - Need to leverage business processes
  - We recognized that there were existing legacy systems that we thought would be good candidates for services
    - There was a notion of atomic services that they wanted to share
    - It was a message oriented approach with loose coupling
    - We could wrap these into web services to create consistent access
  - We recognized new services that we would need to develop
    - Additional services to add that would offer value to other parts of the enterprise
    - Both non functional as well as business services
Our MDA Approach

• MDA Express provides a mechanism to rapidly generate an application from domain models
  – Capture requirements from subject matter experts (SME)
  – Build representative UML models of the requirements (PIM)
  – Rules engine (Rete algorithm) applies a series of transformation rules and templates (Archetype) to create the downstream PSM and code/artifacts
  – Validate the requirements and re-generate as necessary

• Additional Tools Employed
  – Together Architect 2006 for Eclipse
  – JBoss Application Server
  – JBPM (workflow Engine)
  – Eclipse
Goals of the Project

• Client’s Initial Goal
  – Prove that MDA is a viable approach to building software
    • Apply MDA to a proof-of-concept project to begin to generate metrics
    • Validate or refute claims about MDA
    • Prove that MDA can be applied to an SOA application

• Revised Goal
  – Figure out how to integrate Business Process Modeling with Service Oriented Architecture using MDA
Hurdles to Overcome

• Business Processes and SOA are complimentary, but a gap exists in how they map to each other
  – We felt that MDA would bridge this gap

• Using BPM in an MDA process
  – Needed to figure out how to take BPM and run MDA transformations
    • *We figured we could do it because our engine works with EMF*

• Needed a way to describe the message structure
  – Knew we could use a UML class diagram to capture the structure
Hurdles to Overcome (cont.)

• Needed to relate the BPM model to the UML model
  – Used the data object in BPM

• Capturing the contract for communication between services
  – Knew we could create a component diagram to define the contract
  – Use data object to mark the relationship between the BPM artifact and UML artifact

• How to add metadata that’s used in the transformation process
  – Knew how to use stereotypes in UML
  – How do we do the same thing with BPM models
    • Together Architect has a concept of categories
Lessons Learned

• MDA is the glue that can bring Business Process Modeling and SOA together

• A Belief exists that SOA and MDA are mutually exclusive
  – Need more education across the industry
  – Small proof-of-concept project will provide localized education

• Need a strong team lead who understands BPM, SOA, and MDA and how to integrate them
  – Provides direction to individual team members who may only know one of the technologies
  – Instills confidence across the team and client base
Lessons Learned

• An MDA approach allows for a well-architected solution under tight schedule and budget constraint

• MDA tools must support the development of applications that are built upon a variety of architectures, languages, and platforms

• An organization’s previous experiences with modeling may effect their willingness to adopt an MDA approach
  – Need to understand the benefits of modeling before they can discuss the benefits of MDA
Lessons Learned

• Management, Architects, and Developers may express various levels of resistance to MDA
  – Managers
    • *May not understand the significance if they do not fully understand their current processes or the software development lifecycle*
    • *Tend to delegate the evaluation of MDA to the technical people*
  – Architects/Developers
    • *May have had experiences with code generation in the past*
    • *Have worked hard to establish their value to their organization*
    • *Tend to believe that it is not possible that a “program” can do a better job*
      – It’s the Monet syndrome
        » *Asking developers to use MDA is like asking Monet to paint by numbers*
Lessons Learned

• Readiness to use MDA does not have anything to do with a tool
  – Do they have a process for capturing requirements?
  – Do they manage the requirement lifecycle?
  – Are they experienced at modeling their requirements?
  – Do they understand UML?
  – Do they have an enterprise vision for software development?
  – Does their organization dictate one or more architectural frameworks in a reference implementation document?
PSM Creations

• Created a class diagram of the message structure using a UML profile
  – XSD profile based on “Modeling XML Applications with UML” – David Carlson

• Created the logical data model for information that needed to be persisted
  – UML Profile based on “Refactoring Databases” - Scott Ambler

• Created a class diagram that captured the WSDL structure using an SOA profile
  – Profile for software service based on several things we read in the open space

• We generated an activity diagram to capture the business process model
Metrics

• PIM
  – 8 business process models of average complexity
  – Identified 6 services
  – 15 UML diagrams (component and class diagrams)

• PIM to PSM
  – 265 transformation rules were utilized
  – Created
    • 36 Packages
    • 249 Classes
    • 790 Attributes
    • 230 Associations

• PSM to Code
  – Used 220 Rules
  – Used 78 Templates
  – Created
    • Files 250
    • Lines of Stuff (xml, JAVA, BPEL, WSDL, etc.) 32,480
Contact Information

Robert Mitchell
Principal, INHERIT, LLC
Mitchell@inherit.com
(703) 851-1557 (cell)

Robert Lario
Principal, INHERIT, LLC
Robert.lario@inherit.com
(404) 713-0612

www.inherit.com