Implementing Enterprise Architecture with MDA

Michael Rosen
Chief Technologist
Wilton Consulting Group
Mike.rosen@wiltonconsultinggroup.com
Mike Rosen

Consultant
- SOA, EA and MDA modeling, implementation, strategy and training
- Chief Architect for service-oriented systems
  - Finance, Insurance, Telecom
- IT Architecture and Strategy
- 25+ years experience in distributed systems, software and architecture

Affiliations
- Cutter Consortium – Director of Enterprise Architecture
- SOA Institute – Editorial Director, SOA Conference Co-chair

Author
- Cutter Consortium
  - “Designing Service Oriented Applications”
  - “EA – It’s not Just for IT Anymore”
  - “Agile Methods and Enterprise Architecture”
  - “Enterprise Architecture Roll-out and Training”
  - “Service Oriented Integration: Aligning SOA with Enterprise Integration”
  - “Implementing SOA on Common Technologies”
  - “An Application Centric Approach to Enterprise Architecture”
- Integrating CORBA and COM Applications, 1998, Wiley
Agenda

- What is EA?
- MDA Process
- Implementing EA with MDA
- Case Study
- Conclusion
What is Enterprise Architecture?

- The primary goal of Enterprise Architecture is to align present and future IT systems with business goals and strategy.

- Typical technology goals are:
  - Establish interoperability between systems
  - Provide consistently stable technology
  - Implement cost-effective standard infrastructure, leverage existing investments
  - Maximize benefit to the business through appropriate reuse of technology

- Business and architecture goals are:
  - Reduce IT expenditures
  - Support IT portfolio management
  - Support outsourcing
  - Provide framework for enterprise IT governance

- EA is about enabling change and managing complexity.
WebGo Telecom Portal

myTelecom Home Page

Services at a Glance

<table>
<thead>
<tr>
<th>Service</th>
<th>Retail Price</th>
<th>Volume Price</th>
<th>Service Tax</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Phone</td>
<td>$3.00</td>
<td>$2.00</td>
<td>$40.00</td>
</tr>
<tr>
<td>Internet</td>
<td>$5.00</td>
<td>$3.00</td>
<td>$30.00</td>
</tr>
<tr>
<td>Mobile</td>
<td>$5.00</td>
<td>$7.00</td>
<td>$70.00</td>
</tr>
<tr>
<td>Mobile Internet</td>
<td>$2.00</td>
<td>$5.00</td>
<td>$5.00</td>
</tr>
<tr>
<td>Television</td>
<td>$1.00</td>
<td>$90.00</td>
<td>$90.00</td>
</tr>
</tbody>
</table>

Advice

- John, we have noticed that you contact Australia many times around 5pm. You might want to note that calls after 6pm to that destination are 30% cheaper due to the charging schedule in place.
- Also please note that you may experience difficulties this evening in your local area in making a call with your mobile phone, as the special concert that is being held at your local stadium is booked out, and we are expecting a surge in demand.

Ask Eric?

Please type your question here:
Let’s assume all of the external systems for WebGo exist. While building WebGo, we would need certain information.

The first Architectural-Level needs for WebGo are:
- How the WebGo Portal will connect to those systems?
- How it will communicate (protocol)?
- How it will understand data (formats, meaning)?

The next Architectural-Level issues are understanding of the process WebGo implements:
- What users can do?
- What level of security is there?
- What amount of automated process is there?

Then as we turn WebGo on:
- What scale of usage was introduced?
- What about reliability of the integrated systems?
- What about operating (supporting) this new portal?
The previous slide did not describe a very broad view. There are additional considerations:

- How does WebGo support a business strategy?
- How will we measure the success of that?
- What of WebGo was already implemented in the enterprise?
- Did we just add a redundancy?
- What technologies did we introduce, and what is their support cycle?
- What did WebGo tell us about the various data models in the back-end systems?
- When WebGo becomes “WebGone”, are there components or services that can be re-assembled to make “WebGo++”?

The immediate needs described application architecture

- Application Architecture is part of EA, but it is not the same as EA
**Enterprise Architecture:**
Describes concerns and guidelines for integration of process and data across the entire enterprise. Applied to many application domains.

**Application Architecture:**
Describes abstract concepts, things and relationships within the application domain. Applies to many products or applications.

**Design:** Describes specific items and relationships, Applies to a single product or application.
WebGo Portal Architecture in Detail

Drives Service Provisioning

Driven by SLA & Assurance

Driven by Billing

Driven by Marketing

Driven by Fault Mgmt System

Driven by CRM System

Several Diverse Systems involved

Advice
- John, we have noticed that you contact Australia many times around 5pm.
- You might want to note that calls after 6pm to that destination are 30% cheaper due to the charging schedule in place.
- Also please note that you may experience difficulties this evening in your local area in making a call with your mobile phone, as the special concert that is being held at your local stadium is booked out, and we are expecting a surge in demand.

Ask Eric?
WebGo Portal “Conceptual Architecture”

Channels

Customers
Employees
Suppliers
Other Partners

Personalization
Security
Workflow

Content Management
Knowledge Management
Rules

Application Platform and Application Integration

Applications and resources

ERP
Legacy Systems
CRM
Packaged Apps
Corporate Data
WebGo Implementation Slice

Application Platform

Portal

Existing Functions

Customer

Existing Functions

legacy systems
WebGo Application Architecture

user
Presentation → View Controller → Work Coordinator → Session → User Profile

workspace
Business Composition → Business Process Component → Business Entity → Resource Adapter

enterprise
Application Adapter

resource
legacy System
packaged application

services

infrastructure
Application Services Platform
Application Architecture

- Describes how to build systems that meet specific application requirements

- Describes how to:
  - Create consistent applications across the enterprise
  - Use technical architecture
  - Configure and manage applications

- Includes
  - Architectural elements
  - Fundamental product elements
  - Common construction patterns
  - User model
  - Application frameworks
Web Services Architecture

Web Services Platform
Architectural Foundations

Tiers

- **User**
  - Presentation and device independence

- **Workspace**
  - User session and data manipulation

- **Enterprise**
  - Business processes and entities

- **Resource**
  - Shared enterprise resources

Layers

- **Application**
  - Application level business logic

- **Services**
  - Common utility functions applied across tiers

- **Infrastructure**
  - Underlying technical and communication capabilities
WebGo Deployment View

Enterprise Interwoven Environment
- Primary: Content Manager
- Redundant: Content Manager
- File Server
- CAM Process
- HTTPS
- HTTP

Primary WS: Web Server
- CAM Process
- LDAP
- HTTPS
- HTTP

Redundant WS: Web Server
- CAM Process
- LDAP
- HTTPS
- HTTP

Primary AS: Application Server
- ETL Process Server
- JDBC
- ODBC

Redundant AS: Application Server
- ETL Process Server
- JDBC
- ODBC

Primary RS: Reporting Server
- Database Server
- ODBC

Redundant RS: Reporting Server
- Database Server
- ODBC

Planned for a future release ... not in scope for release 1.
- Firewall
- HTTPS

External: Workstation
- HTTPS
- HTTP
- HTTPS
- HTTP
- HTTPS
- HTTPS
- HTTPS

Internal: Workstation
Technical Architecture

- Describes how to build systems that meet specific technical requirements
  - Non-functional requirements
  - Enterprise “abilities”
  - Architectural qualities

- Describes standards, products, protocols and technologies

- Includes
  - Architectural layers
  - Distribution tiers
  - Infrastructural services
Business Architecture — Value Chain

Primary Activities
- Process Management
- Core Business Process
- Quality Management
- Product Information

Supporting Activities
- Supporting Processes
- Support Asset Info

Supporting (Financial, HR, IT, ...) Assets

Goal
“… to provide a premium communications network and customer services”

Process Management
- Forecasting
- Network/Product Planning
- Capital Equipment Acquisition
- Construction
- Network Monitoring

Customer Order
- Service Provisioning
- Customer Service
- Customer Billing
- Real-time Operations

Network Research & Quality

Customer Usage & Quality

Network Infrastructure Assets
- Administration Management
- IT Management
- HR Management
- Financial Management
Business Architecture

- Understand and describe the underlying Business Value Chain within the enterprise
- Understand and describe fundamental business processes
- Link business strategies to business processes
- Leverage technology to align business process strategies with IT initiatives
- Understand the gap between business process “as is” and “to be” environments. Identify a transition strategy.
Enterprise Architecture Revisited

- A set of architectures, which taken together, provide a complete view of an organization.

- Conforms to architectural principles, especially Separation of Concerns:
  - Business
  - Information
  - Application
  - Technical
  - Operations
  - Implementation

- Architecture must achieve three primary goals:
  - Describe a solution to a specific set of problems and requirements.
  - Effectively communicate the solution to all stakeholders.
  - Enable the construction of systems that conform to the architecture.
Architecture-Driven Design

Enterprise Architecture

Business Architecture

Information Architecture

Application Architecture

Technical Architecture

Implementation Architecture

Operational Architecture

Application Analysis and Design

Application Requirements

Deployed Service

Business Requirements

Information Requirements

Enterprise Requirements

Program Requirements

Technical Requirements

Implementation Requirements

Operational Requirements
MDA Mappings

Computation Independent Model

Platform Independent Model

Platform Specific Model

Code

PIM → PIM Mapping

PIM → PSM Mapping

PSM → PSM Mapping

PSM → Code Mapping
MDA Under the Hood

- **Computation Independent Model**
- **Platform Independent Model**
- **Platform Specific Model**
- **Code**

**Architectural Standards and Guidelines Enforced in Model Profiles**

**Enterprise QoS and non-functional requirements implemented in transformations**
Metamodels

- Provide rules for how to build a correct model for a particular purpose, e.g. “business integration metamodel”

- UML Profile
  - Provides a targeted subset of UML
  - Standard mechanism for extending UML

- Refinement and Constraint
  - Metamodels refine the definition of modeling elements by placing constraints on their behavior through the use of stereotypes

- Stereotypes
  - Standard UML Stereotypes
    - <<boundary>>, <<control>>, <<entity>>
  - Extending the UML Stereotypes
    - Inheritance used to extend and refine the meaning of stereotypes
    - Tagged Values use to apply specific properties
WebGo Application Architecture

user
- Presentation
- View Controller
- Work Coordinator
- Session
- User Profile

workspace

enterprise
- Business Process Component
- Business Entity

resource
- Application Adapter
- Resource Adapter

services
- Authorization Service
- Profile Service
- BPM Service
- Persistence Service
- Configuration Service
- Logging Service

Application Services Platform

infrastructure

application

legacy System

packaged application
Metamodel Stereotypes

Standard UML Stereotypes

Usage

Location and purpose

Scope and visibility

Class Diagram: Metamodel / Meta class Interactions

© Michael Rosen 2007
MDA Profiles

- **Computational Independent Model**

- **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 Revisited

- Business Analyst
- Architect / Designer
- Developer / Tester
- MDA Architect

Models:
- Computation Independent Model
- Platform Independent Model
- Platform Specific Model
- Code
Architecture Profiles

Each Profile formally specifies a particular architecture

- Computation Independent Model
- Platform Independent Model
- Platform Specific Model
- Business Model
- Application Model
- Platform Model
- Code

Business Architecture
Application Architecture
Technical Architecture

<<UML Profile>>

© Michael Rosen 2007
Architecture-Driven Design

Enterprise Architecture

- Business Requirements
- Information Requirements
- Enterprise Requirements
- Program Requirements
- Technical Requirements
- Implementation Requirements
- Operational Requirements

Business Architecture

Information Architecture

Application Architecture

Technical Architecture

Implementation Architecture

Operational Architecture

Application Analysis and Design

Application Implementation

Deployed Service

MDA
EA and MDA Summary

- Enterprise Architecture involves separating concerns into viewpoints.
- Specific architectural viewpoints are formalized in profiles and metamodels.
- Profiles enforce application conformance to enterprise architecture.
- MDA provides a standards-based approach to defining profiles and using the profiles to help automate development.
Case Study

- Large US Insurance Company
- $25 Billion USD
- ~ 20 Lines of Business

- No consistency across design
- Little to no reuse
- Redundant functions and data
- IT costs too high
- IT time-to-market to slow

- Enterprise Architecture Group exists
- …with little to no effect
- …ARB Exceptions at 80%
What we did

- Institute an architecture based / MDA approach
- Create a set of UML Profiles that specify the enterprise architecture
- Integrate the UML Profiles into a modeling framework built with Rational Rose and REI
- Tie the Software Architecture Specification to the Framework
- Create education program: Architecture, Modeling, Framework
- Ramp up the number of application architects. Focus them on assisting projects
APSL Enterprise Process

- Define the approach
  - Integrate enterprise architecture into the development process.
  - Create meta-models and profiles

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

- Define the solution
  - Refine into PIMs and PSMs

- Leverage the results
  - Integrate assets into a reuse repository
  - Architecture and design accommodates: reuse, customization, enhancements, versioning…
Rose Framework for Application Architects

Define the Problem

Define the Solution

Define the Approach

Sports Club Management System

Problem Definition

Solution Definition

Project Support

© Michael Rosen 2007
Approach Definition Views

Enterprise Architecture

- Business Architecture
- Information Architecture
- Application Architecture
- Technical Architecture
- Implementation Architecture
- Operational Architecture

Architectural Development Services
- Architectural Styles
- Foundational Services
- Frameworks
- Patterns
- Technology Mapping
Problem Definition Views

<<Focus Package>>
<project name - problem space>

Business Analysis Model

Business Domain Model

Business View

Business Use Cases

Business Process Models

Enterprise Architecture

Business Architecture

Information Architecture

Application Architecture

Technical Architecture

Implementation Architecture

Operational Architecture
Solution Definition Views

Solution Definition

<<Focus Package>>
<project name - solution space>

Logical Architectural View

Application System Analysis
High Level Design Model
Structural View

Operational View

Process View
Deployment Views

Enterprise Architecture

Business Architecture
Information Architecture
Application Architecture
Technical Architecture
Implementation Architecture
Operational Architecture
Logical Architecture

- The Logical Architecture is a result of Application Analysis and the application of patterns

- Verifies that business requirements are addressed

- Confirms architectural alignment

- Identifies services
  - Consumed
  - Produced

- Reflected in the Model and the Software Architecture Document
The SAD is a communication tool that

- Ties together business needs, application development, domain / enterprise views within a project
- Keeps scope within application area, but with an enterprise architectural underpinning
- Helps integrate with other projects/areas
- Is an important physical deliverable

- Is a poor choice of acronym…
Basic Approach to SAD Modeling

- Use the SAD Rose framework to create your model and to document the elements in the model.
- Use the SAD SoDA template to create a project template for extracting information from the model and documenting sections of the SAD not placed in the model.
  - Don’t forget to update the File / Properties.
- Generate the SAD report.

Software Architecture Spec.
Challenges

- Adoption
  - Overcome skepticism about architecture
  - Change behavior of software organization
  - Win over business sponsors

- When is an architect done?
  - Don’t get sucked into implementation
  - Logical architecture level
  - Preliminary SAD – 4 weeks, SAD – 10 weeks

- Ramp-up
  - Where do you get all the architects?
  - Grow architects from within
  - Develop project criteria
New EA Organization

Central IT
- Services
  - Taxonomy
  - Repository
  - Maintenance
  - Project Mgmt
  - Enhancements
- Infrastructure / Operations

EA Program
- Review Board
- Enterprise Architects
- Consulting Architects
  - Business
  - Information
  - Application
  - Technology
  - Specifications
  - Models
  - Frameworks
  - Infrastructure

Consulting Architects

Business Units
- LOB Architects

Projects
- Analysis
- Design

Assigned To
Results

- Consistency
  - SADs have same structure and format across projects
  - Project architectures have the same logical structure, roles and responsibilities
  - Allows for comparison between projects
  - Enables standard technology platforms

- Reuse
  - Opportunities for reuse more easily identified
  - Component ~reuse up 200%

- Usage
  - Used for all ‘App Architect’ projects
  - Increasing adoption of LOB architects and designers

- Demand
  - 5 architects first year
  - 25 architects 2nd year, 50 architects 3rd year
  - 500 projects get architects…1000 others with them could

- Exceptions
  - Compliance up 500% for applications with architects, review process streamlined
  - Exceptions down to 10% for projects with architects
Architecture Value Proposition

- Alignment of IT systems with business goals and strategy
- Improved customer satisfaction
- Consistency across applications
- Reduced costs to implement, maintain, evolve, and retire applications
- Improved IT operations
  - Common semantics and information
  - Interoperability between applications
  - Integration of applications
  - Reuse of infrastructure, frameworks, utilities
### Improving Productivity and Consistency

#### Cost of a new application

<table>
<thead>
<tr>
<th>User Interfaces</th>
<th>Business services, processes and entities</th>
<th>Tools</th>
<th>Architecture</th>
<th>Custom Processes</th>
<th>Custom Infrastructure</th>
<th>Infrastructure and processes</th>
</tr>
</thead>
</table>

#### Key:

<table>
<thead>
<tr>
<th>New</th>
<th>Existing</th>
</tr>
</thead>
</table>

#### Typical application

*Today*

*Built mostly from scratch*

#### High productivity

*Tomorrow*

*Built on an existing foundation*

---

Each application needs the same parts. Don’t build them all each time.

Dramatically reduce costs by utilizing existing IT assets.
Parallel Paths of Technology and Application

Application Architect

Application Design to meet Business Functionality

Technology Architect

Infrastructure Design to meet QoS Requirements

Implementation

Deployment
The Importance of Early Knowledge

- Expenditures committed
- Domain/technology knowledge
- Project costs

Time vs. Expenditure

Courtesy: Cutter Fellow Ken Orr
Thank You!

“All complex problems have solutions that are clear, simple… and wrong”

— H.L. Mencken, 1949