Applying MDA Standards to bring the A to SOA

Pete Rivett, CTO Adaptive
pete.rivett@adaptive.com
Outline

- The SOA ‘platform’
- Why SOA needs MDA
- OMG technical standards supporting SOA
- OMG’s SOA Special Interest Group
SOA Needs Planning and Architecture

- Fallacy: A bottom-up approach to SOA is good enough. A Web Service here, a Web Service there, and pretty soon you have an SOA. Build enough Web Services, and make sure they are secure and managed, and that's all you need to do to build an SOA.

Counterpoint: The best approach to building an SOA is to combine a bottom-up approach of building Services that solve integration problems with a top-down, architectural approach that provides process-driven Services that enable business agility.

Jason Bloomberg, ZapThink, 08-05-2004
SOA ≠ Web Services

- SOA does not require use of web services
  - CORBA
  - REST/http

- Use of web services does not by itself give SOA
Web Services Platform Jungle

For full chart see innoQ.com
Platform Jungle contd

- Lots of standards
- Interdependencies between them
- Many still under development
- Usable stacks (combinations of proprietary and standards) forced by vendors
- In addition to all this is Enterprise Service Buses etc.

This is exactly the situation that MDA was designed for
Enterprise Service Bus

- Just one platform type for deploying SOA
- In reality, most large corporations will have several
- An ESB is made up of the following parts:
  - Service Interface Points (SIP) where the service connects to the bus.
  - Mediation services that switch protocols, and transform, enrich, route, distribute, monitor and correlate messages.
  - A Registry that contains metadata definitions of all the services connected to the ESB to enable registration, discovery and deployment of the services.
- And has 4 deployment patterns that reflect different levels of autonomy and control:
  - global
  - directly connected
  - brokered
  - federated
MDA Archetype

- Focus the investment on the model
- Then generate/deploy to a platform
  - Whether completely automated or not
- And remember that ‘Platform’ is a relative term
MDA for SOA
- Beyond Platform Independence

- Business traceability and alignment
  - Services in the context of the business/enterprise architecture

- Change and complexity management

- Development tool independence

- Models are easier to communicate and manage than XML files
Pulling it all together
Not just OMG View

“SOA Metamodel” from ZapThink
SOA Layers
Relevant OMG Standards - MDA

- Meta Object Facility (MOF)
  - Metamodel foundation

- XML Metadata Interchange (XMI)
  - Model interchange

- MOF Versioning
  - Model management

- MOF Queries Views and Transformations (QVT)
  - Transformation of models

- MOF Model to Text Transformation (M2T)
  - Generation of artifacts (code/XML)
Relevant OMG Modeling Standards - Technical Content

- Unified Modeling Language (UML)
  - Architecture and design of services/internals
  - General purpose modeling language (via profiles)

- Reusable Asset Specification (RAS)
  - Packaging of reusable assets for retrieval, usage and deployment

- Common Warehouse Metamodel (CWM)
  - Service payloads (information content) and mapping to enterprise data storage

- Ontology Definition Metamodel (ODM)
  - Semantics of information for interoperability

- Knowledge Discovery Metamodel (KDM)
  - Models of the enterprise systems to which services typically map

- Software Process Engineering Metamodel (SPEM)
  - Modeling the SOA development process
- Packaging of artifacts for reuse
- Manifest for:
  - Classification (finding)
  - Usage for a context
  - Variations for a context
- Independent of what’s being reused
- Manifest and artifacts may be in a Zip file or the artifacts may be external (e.g. a CM system)
- Two flavors of interchange for manifest
  - XML Schema (legacy)
  - MOF/XMI
RAS Profiles

- Details for specific types of asset
- Spec includes a Component Profile and a Web Service Profile
- Options for a Profile:
  - New metamodel (or Schema) that extends the RAS metamodel
  - Reference to a UML Profile
  - Reference to a MOF Metamodel
Default Web Service RAS Profile
Information Management Metamodel (IMM) Outline of requirements

- **Leverage** latest modeling standards
  - MOF2 Metamodel
  - UML2 Profile where appropriate for visual modeling
- Modular structure
- Inclusive of CWM 1.x scope (with migration)
- XML Schema metamodel and profile
- Relational metamodel and profile
- Record metamodel and profile
- Data modeling notation mapping for IDEF1X and IE
- Reconciliation of persistent and programming language structures
- Address issues deferred from CWM 1.1
Knowledge “dimensions” in KDM

Based on Architecture Views

Conceptual architecture

Structural “things”

Data “things”

Domain model

Behavior “things”

Code “things”

UI “things”

Requirements,
Features,
Use Cases

Approximates modeling world from the ground up
Language-independent KDM repository

Java extension extends C++ extension extends Cobol extension extends C++ extension

Language-aware KDM Tool

Analyst

Language-independent KDM Tool

Cobol KDM Tool 1 uses Cobol KDM Tool 2 uses

Cobol extension

Java KDM Tool

Java extension uses C++ KDM Tool uses C++ KDM Tool

Constraints are defined in Core KDM package: All language-specific extensions are uniform

Transformation Tool
Where KDM adds value?

- **Architectural context for**
  - Understanding software
  - Modularization, untangling “hairballs”
  - Discovering reusable components
  - Inventory, estimations of effort
  - Modernization

- **Architecture management**
  - Before modernization
  - After modernization

- Application Portfolio Management

- Information exchange between vendors
## Modernization Scenarios Matrix

<table>
<thead>
<tr>
<th>Modernization Scenario</th>
<th>Check Box</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. Application Portfolio Management</td>
<td></td>
</tr>
<tr>
<td>II. Application Improvement</td>
<td></td>
</tr>
<tr>
<td>III. Language-to-Language Conversion</td>
<td></td>
</tr>
<tr>
<td>IV. Platform Migration</td>
<td></td>
</tr>
<tr>
<td>V. Non-Invasive Application Integration</td>
<td></td>
</tr>
<tr>
<td>VI. Services Oriented Architecture Transformation</td>
<td></td>
</tr>
<tr>
<td>VII. Data Architecture Migration</td>
<td></td>
</tr>
<tr>
<td>VIII. Application &amp; Data Architecture Consolidation</td>
<td></td>
</tr>
<tr>
<td>IX. Data Warehouse Deployment</td>
<td></td>
</tr>
<tr>
<td>X. Application Package Selection &amp; Deployment</td>
<td></td>
</tr>
<tr>
<td>XI. Reusable Software Assets / Component Reuse</td>
<td></td>
</tr>
<tr>
<td>XII. Model-Driven Architecture Transformation</td>
<td></td>
</tr>
</tbody>
</table>
Structure of KDM packages

<table>
<thead>
<tr>
<th>Environment</th>
<th>Conceptual</th>
<th>Storage</th>
<th>UI</th>
<th>Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>RunTime</td>
<td>Logical</td>
<td>Build</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data</td>
<td>Code</td>
<td>Actions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SourceRef</td>
<td>SourceBundle</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Core</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SPEM Example – Use Cases
SPEM Example - Activities
OMG SOA SIG

- Working Group meeting Dec 2005
- Chartered as SIG in Feb 2006
- Working to identify and prioritize gaps in existing OMG standards and build a roadmap
- Standards only adopted through existing OMG task forces
- Questionnaire available to get your input
  - on paper
The SOA SIG provides a forum for discussion of SOA definition, methodologies, models, and both business and technical implications.

Primary goals:

- To support an MDA approach to SOA that links architectural, business and technology views of services, including Business Process Management (BPM) and Event-Driven Architecture (EDA).

- Identify and foster development of OMG modeling standards for SOA that integrate with and complement standards developed by other organizations such as W3C, Open Group and OASIS.

- To improve awareness and understanding of SOA by OMG members.

- To coordinate SOA related efforts within OMG.

See www.omg.org/soa
## Draft Gap Analysis

<table>
<thead>
<tr>
<th>Gap</th>
<th>Opportunity for OMG SIG</th>
</tr>
</thead>
<tbody>
<tr>
<td>■ Modeling approach (Service Oriented Modeling)</td>
<td>■ White paper on SOA Framework mapped to OMG standards (MDA and others)</td>
</tr>
<tr>
<td>– SOA Framework mapped to MDA, other OMG stds.</td>
<td>■ RFP(s) for SOA Metamodel</td>
</tr>
<tr>
<td>– SOA Metamodel</td>
<td>– Use of MDA standards (UML, MOF, XMI, CWM/IMM)</td>
</tr>
<tr>
<td>– Service Modeling Language (SML – a flavour of UML and BPMN)?</td>
<td>■ RFP for UML Profile for Web Services (review of IBM’s UML profile for software services)</td>
</tr>
<tr>
<td>■ Focus on complete Life Cycle of a Service</td>
<td>■ SML?</td>
</tr>
<tr>
<td>– Model, develop, manage and monitor</td>
<td>■ Covered by SOA metamodel?</td>
</tr>
<tr>
<td>– Metrics (service availability, performance, maturity, SLA...)</td>
<td>■ SOA Metrics RFP</td>
</tr>
<tr>
<td>■ Architecture and topology of SOA middleware potentially including ESB</td>
<td>– Reuse/extend ADM Metrics Package</td>
</tr>
<tr>
<td>■ Mapping of Services to business functions/processes and components</td>
<td>■ ESB (Enterprise Service Bus) ?</td>
</tr>
<tr>
<td>■ SOA Governance</td>
<td>■ Use of Archimate</td>
</tr>
<tr>
<td>– Policy, Contract, Regulatory Compliance</td>
<td>■ Cross metamodel means to model traceability</td>
</tr>
<tr>
<td></td>
<td>■ BPDM and BMI as examples</td>
</tr>
<tr>
<td></td>
<td>■ Research RFPs being issued by RC DSIG</td>
</tr>
<tr>
<td></td>
<td>■ RFP for Service Policy metamodel?</td>
</tr>
<tr>
<td></td>
<td>■ RFP for Service ‘Contract’ metamodel?</td>
</tr>
<tr>
<td>Gap</td>
<td>Opportunity for OMG SIG</td>
</tr>
<tr>
<td>-------------------------------------------------------------------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>● Standard Service Registry/Repository model</td>
<td>● RFP for SOA Registry/Repository metamodel</td>
</tr>
<tr>
<td>● Gap between service development and service registration and deployment</td>
<td>● Use/extend UDDI</td>
</tr>
<tr>
<td></td>
<td>● RAS, EDOC</td>
</tr>
<tr>
<td>● Correlation/mapping to EDA</td>
<td>● Event Metamodel?</td>
</tr>
<tr>
<td>– Events that trigger Service execution</td>
<td>● Pub-Sub metamodel? (OMG EAI metamodel)</td>
</tr>
<tr>
<td>– Causality relation with Events</td>
<td>● Causality relationship metamodel</td>
</tr>
<tr>
<td>- Sense and respond</td>
<td></td>
</tr>
<tr>
<td>● Service relationship with Agents</td>
<td>● Agent SIG?</td>
</tr>
<tr>
<td>– Agent part of Sense and Respond?</td>
<td>● Other sources like W3C?</td>
</tr>
<tr>
<td>● Service Semantics (Service Ontologies)</td>
<td>● Work with Ontology SIG</td>
</tr>
<tr>
<td>– Semantic reasoner</td>
<td>● WSMO (Web Service Modeling Ontology Metamodel)</td>
</tr>
<tr>
<td></td>
<td>– Service reasoner metamodel?</td>
</tr>
<tr>
<td>● Way to model web service functionality and policy independent of WS* platform languages</td>
<td>● Mapping PIM &gt; PSM for web services</td>
</tr>
<tr>
<td>● SOA Relationship with SCA, SDO</td>
<td>● Mapping to SCA</td>
</tr>
<tr>
<td>● SOA Relationship with CBA (component-based Architecture)</td>
<td>● Mapping to CBA?</td>
</tr>
</tbody>
</table>
Is OMG too late to the party?

“Yes but no but…”

- There is a lot already there generically with MDA standards
  - at business and technical levels
- The SOA Platform is still very fragile and evolving
  - (ironically to become more like CORBA!)
  - initiatives such as SCA still emerging
- The SOA Reference Model (OASIS) is only just completing
- Consensus over best practice for modeling SOAs is still emerging
  - especially with respect to business alignment
- A lot more work is still needed with respect to service metadata
  - Contracts, policies, semantics
- So only now is sufficient maturity emerging with SOA to start developing
  SOA-specific modeling standards
  - OMG is not a research organization
Summary

- SOA is the best argument there is for MDA
- OMG already has a large number of standards addressing the needs of SOA
- It has established a SOA SIG to address SOA specifics