SoaML standard and example

Slides by Arne Jørgen Berre, Cory Casanave, Ståle Walderhaug, Erlend Stav, Babak Farshchian, OMG, et al.
(see references page at the end)
Service oriented architecture Modeling Language (SoaML) - Specification for the UML Profile and Metamodel for Services (UPMS)

OMG Adopted Specification
Finalisation Task Force Beta 2 document (FTF Beta 2)

OMG Document Number: ptc/2009-12-09
Associated File(s)*: http://www.omg.org/spec/SoaML/20091101
The SoaML submission team

- **Submitters**
  - 88Solutions
  - Adaptive
  - EDS
  - Model Driven Solutions
  - Capgemini
  - Fujitsu
  - Fundacion European Software Institute
  - Hewlett-Packard
  - International Business Machines
  - MEGA International
  - MID GmbH
  - Rhysome
  - Softeam
  - Telelogic AB

- **Supporters**
  - Everware-CBDI
  - General Services Administration
  - Inherit
  - Mega
  - BAE Systems
  - DERI – University of Innsbruck
  - DFKI
  - France Telecom R&D
  - NKUA – University of Athens
  - Oslo Software
  - SINTEF
  - THALES Group
  - University of Augsburg
  - Wilton Consulting Group
Context for Enterprise SOA

MDA Terms

- Business Concerns
  - Business Model
    - Enterprise Services (e-SOA)
    - Roles, Collaborations & Interactions
    - Process, Information & Rules
  - Logical System Model
    - Technology Services (t-SOA), Components, BPM
    - Interfaces, Messages & Data
  - Technology Specification
    - JMS, JEE, Web Services, .NET
    - WS*, BPEL, XML Schema

Copyright © 2009 Data Access Technologies, Inc.
Model Driven Solutions
May 2009
SoaML Example – The “Dealer Network”

The dealer network models an “industry community” of dealers, shippers and manufacturers. The community defines the SOA architecture by which they all work together.

Note – This example is presented top-down, integrating both the business and technology viewpoints. SoaML can be used top-down, bottom up or middle out. It can be focused on the business and/or the technology based on the users needs.
Marketplace Services

Mechanics Are Us
Dealer

Status
Physical Delivery

Order
Conformation
Shipped

Provider

Consumer

GetItThere Freight Shipper

Ship Req
Shipped
Delivered

Consumer

Acme Industries
Manufacturer
Focus on the Business Model

Business Concerns

Business Model
Business Services (e-SOA)
Roles, Collaborations & Interactions
Process & Information

Logical System Model
Technology Services (t-SOA), Components,
Compositions
Interfaces, Messages & Data

Technology Specification
JEE, JMS, Web Services
WSDL, BPEL, XML Schema...
ServiceCapabilities represent the services or capabilities of some system or architecture. These ServiceCapabilities may then be linked to the Participants that actually provide the capabilities through UML2 Realizations.
Services Architecture for the Dealer Network

A Services Architecture (or SOA) is a network of participant roles providing and consuming services to fulfill a purpose. The services architecture defines the requirements for the types of participants and services that fulfill those roles.
Drilling down - Inside a Manufacturer

Not every manufacturer is going to be the same inside – this shows some of the internals of “Acme”
SOA architectures are able to “drill down” in more detail – this shows the architecture inside of a particular manufacturer, Acme. Other manufactures may have different internal architectures and processes.
Specifying Services

• Specification of services includes
  – The roles each participant plays in the service, such as provider and consumer
  – The message types that go between the participants when the service is enacted
  – The interfaces provided and used by each participant for the service
  – The choreography of the interactions between the participants while enacting the service
  – Placeholders are provided for service policies and motivation

• Modeling services
  – Services are modeled using “Service Contracts” and “Service Interfaces” in SoaML. These use UML interfaces, classes and behaviors.
This view of a service only identifies the service name and the roles each participant plays in the service. This is a high-level summary view.
Service Choreography for “Place Order”

The role of the consumer (a participant that places orders) and the consumers interface

The role of the provider (an order taker) and their interface

The optional interaction to request a quote

The optional interaction to return the quote

The required interaction to place an order

The required interaction to accept or reject the order

A more detailed look at the same service. Note that this models a fully asynchronous SOA – like most business interactions, the document message types are detailed on the next page.
Message Detail for Place Order

This is the detail for the message types that correspond to the interactions for the place order service.

Note that at the technology level this can produce XML schema for the messages.
Example Information Model

CRR Information Model

- Managed Information
  - +identifier : String [0..1](readOnly,isID)
  - +date information created : date [1](readOnly)
  - +date information changed : date [0..1](readOnly)
  - +deactivated : Boolean = false(readOnly)

- Contactable

- Contact Information
  - +isPrimary : Boolean = false

- Electronic Address
  - +email address : String

- Physical Address
  - +address : String

- Email Address
  - +email address : String

- Phone Number
  - +country code : Integer = 1
  - +area code : Integer [0..1]
  - +exchange : Integer [0..1]
  - +number : Integer

- Principal
  - +legal name : String
  - +date principle created : date [0..1]

- Organization
  - +tax ID : String [0..1]
  - +DUNS ID : String [0..1]
  - +date founded : date(redefines date principle created)

- Person
  - +SSN : String [0..1]
  - +birth date : date(redefines date principle created)

- Role Type
  - +description : String

- Role
  - -role type : 1

- Contact information

Technologies, Inc.
Model Driven Solutions
May 2009
Linking messages to business information

SOA Messages can reference and include parts of the logical information model – forming a connection between SOA and enterprise data.
Producing the logical systems model

**Business Concerns**

**Business Model**
- Business Services (e-SOA)
- Roles, Collaborations & Interactions
- Process, Information & Rules

**Logical System Model**
- Technology Services (t-SOA), Components & Compositions
- Interfaces, Messages & Data

**Technology Specification**
- Web Services, JEE, .NET
- WSDL, BPEL, XML Schema...

---

Copyright © 2009 Data Access Technologies, Inc.
Model Driven Solutions

May 2009
Interfaces for Participants

Each role in the service that receives interactions has an interface, this is the interface for a logical technology component and is implemented by components providing or using this service. This service is bi-directional - messages flow in both directions.

Interfaces will correspond with parts of WSDL in a web services mapping of SoaML.
Logical System Components

Components implement the service interfaces providing the link to systems. Participants and services may be used in multiple architectures.

“Ports” on the participating components provide and require the service interfaces for each service provided or used.
Components can be assembled from other components by linking their services. This corresponds to the architecture for Acme.
Adapting Enterprise Systems

This is the inside of the SAP AR component – also a composition, it uses the existing SAP interfaces and adapts them to the enterprise contract.

This separates the concerns of a particular enterprise system from the enterprise SOA. Sometimes the system interfaces are used directly or adapted by an ESB.
Provisioning Technology Artifacts

**Business Concerns**

- Business Model
- Business Services (b-SOA)
- Roles, Collaborations & Interactions
- Process, Rules & Information

**Logical System Model**

- Technology Services (t-SOA), Components
- Interfaces, Messages & Data

**Technology Specification**

- JEE, JMS, Web Services, .NET
- WSDL, BPEL, XML Schema
Realizing the Model

• How do we use I.T. to realize our processes and services?
  – Direct execution frameworks
    • The “no code” approach where the process and services execute directly from the model
    • May use other standards, such as BPEL
  – Wrapping and adapting existing capabilities
    • Automatic or manual creation of “adapter components” that use legacy systems, information or services to create the architected enterprise services
  – Creation of new application components and services
    • Build new capabilities by creating new components and creating composite applications
    • May be visual and declarative or code oriented
• Under the SoaML framework, all of these options can co-exist as a system of systems linked by services
The SoaML Profile

- SoaML is defined as a small set of UML stereotypes.
- These specialize a UML tool for use with SoaML.
- Standard UML can be used as well, as part of a SoaML model.
- Some tools provide enhanced SoaML support.
What you need beyond SoaML

- SoaML is a profile of UML, it needs a UML tool in which to be used – some tools may offer additional support to make creating a SOA easier
- MDA provisioning requires additional tooling which goes beyond the standard. E.G. ModelPro (ModelDriven.org) and RSA (IBM)
- Some SOA execution framework and/or ESB (I.E. “The Platform”)
- Optional, but highly desirable
  - Support for policies
  - Business process execution
  - Information modeling and implementation
  - Business rules
  - Requirements & motivation modeling
  - Security modeling and infrastructure
  - An IDE for elements that are not model driven
  - Application server and/or ESB
Current SoaML Support

- OMG Web site
  - SoaML Wiki: http://www.SoaML.org

- Known SoaML Tooling
  - Cameo SOA+ (NoMagic) UML with SoaML Modeling and Provisioning
  - ModelPro (ModelDriven.org) Open Source MDA provisioning for SoaML
  - Enterprise Architect (Sparx) SoaML Profile for UML tool
  - Objecteering (Softeam) SoaML Profile for UML Tool
  - RSA (IBM) UML tool with SoaML & code generation [Not yet released]