INF5120
”Modelbased System development”

Lecture 1: 14.01.2013
Arne-Jørgen Berre

arneb@ifi.uio.no and Arne.J.Berre@sintef.no
Welcome to INF5120 “Model based System development”

- Model based System Development
  - [http://www.uio.no/studier/emner/matnat/ifi/INF5120/v13/](http://www.uio.no/studier/emner/matnat/ifi/INF5120/v13/)

- Lecturers:
  - Arne-Jørgen Berre
  - Guest lecturers
  - Email: inf5120-forelesere@ifi.uio.no

- Responsible for Obligatory exercises:
  - Yannick Lew, Tore Vatnan
  - Email: inf5120-oppgaver@ifi.uio.no
3 parts of the course

- MDI I (Model Driven Enterprise and business architecture, with service innovation and design)
- MDE II (Model Driven Engineering) – Design of domain specific languages and editors
- MDI II (Model Driven system architecture and realisation)
INF5120 - Lecture plan - 2013

- 1 (14/1): Introduction – overview Enterprise Architecture with UML and BPMN and DSLs
- 2 (21/1): Service Innovation and Design, AT ONE method/workshop – myServiceFellow (Marika Lüders)
- 3: (28/1): Value Networks/VDML BPMN, vs. UML Activity diagrams - Oryx
- 5 (11/2): UML and Req. Modeling – Agile User stories versus Use cases
- 6 (18/2): UML 2.0 and Service Modeling – SoaML and System architecture
- 7 (25/2): Model driven engineering – Metamodels, DSL, UML Profiles etc.
- 8 (4/3): Model driven engineering EMF, Eclipse, GMF
- 9 (11/3): Model driven engineering, transformation technologies (Franck Fleurey)
- 10(18/3): UML Service Modeling – Service composition, USDL, ISO 19119, etc.
- 12(15/4): UML and Entity and Information modeling, UML, ISO 19103
- 13(22/4): UML and Semantic models, Facts, SBVR, Ontologies, Rules
- 14(29/4): UML and Platform models, realisation, migration, Java, Apps, CloudML
- 16(13/5): Conclusion and Summary for INF5120 - Preparation for Exam

Exam: Monday June 3rd, 2013, (4 hours)
Obligs

- Partially individual, partially group - in 3 parts

- Will be presented in more detail on January 28th
Course literature – available on web

- Material from all lectures and OBLIG 1 and 2 a/b

- Some selected articles and documents, and subset of standard documents from OMG - will be updated

- Web site for practices:

  - Inf5120.modelbased.net

- Practical use of tools (OBLIG, part 2)

- Oryx - Cloud based UML, BPMN (and ServiceML) editor


- GMF - http://www.eclipse.org/gmf/


- See also: http://live.eclipse.org/node/575

- BPMN – http://www.eclipse.org/stp/bpmn

- SoaML - www.soaml.org
INF5120 Methodology (release 1.0)  

http://INF5120.modelbased.net

About

The INF5120 Methodology provides practices that contain guidelines for service innovation, business model innovation, business motivation modelling, business process networked enterprises. The methodology adopts an OMG Model Driven Architecture (MDA) approach to software development and prescribes a set of model artefacts (BPMN) and Service oriented architecture Modelling Language (SoaML) that are used in the development of service-oriented solutions. In particular, the methodology service-oriented solutions from both a business and an IT perspective. The methodology also adopts the Scrum practice for project management and includes some other practices.

Navigation

You can navigate the methodology website from different perspectives by using the links below or by using the tree browser on the left:

- Getting Started
- Delivery Processes
- Practices
- Role sets
- Work Products
- Tasks

Practices

The INF5120 Methodology is a collection of practices. The table below shows the practices categorized according to disciplines:

<table>
<thead>
<tr>
<th>Disciplines</th>
<th>Practices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Networked business value analysis and innovation</td>
<td>• Service innovation</td>
</tr>
<tr>
<td></td>
<td>• Business model innovation</td>
</tr>
<tr>
<td>Networked process and service modelling</td>
<td>• Business motivation modelling</td>
</tr>
<tr>
<td></td>
<td>• Business process modelling</td>
</tr>
<tr>
<td>Requirements engineering</td>
<td>• User story</td>
</tr>
<tr>
<td></td>
<td>• Use case modelling</td>
</tr>
<tr>
<td>Software architecture</td>
<td>• Services architecture and contract modelling</td>
</tr>
<tr>
<td></td>
<td>• Service interface and component modelling</td>
</tr>
<tr>
<td>Testing</td>
<td>• Concurrent testing</td>
</tr>
<tr>
<td></td>
<td>• Test Driven Development</td>
</tr>
<tr>
<td>Project management</td>
<td>• Scrum</td>
</tr>
</tbody>
</table>
UML 2.0

- UML 2.0 and SysML Background and Reference material
- See www.uml-forum.com/specs.htm

- Også hos OMG:
  - http://www.omg.org/uml/ (UML)
  - http://www.omg.org/mda/ (MDA)
  - http://www.omg.org/cwm/ (MOF, XMI, CWM)
UML 2.0 recommend books:

UML 2.0 in a Nutshell
by Dan Pilone (Author), Neil Pitman (Author)

The Unified Modeling Language User Guide
(G, Booch, J. Rumbaugh, Jacobsson)
Agile Service Development (1/3)

We will use a publication preprint initially

Table of Contents

1 Introducing Agile Service Development .................................................. 1
   1.1 Introduction .................................................................................. 1
   1.2 Services and Service Thinking ....................................................... 3
   1.3 Agile Enterprise Engineering ......................................................... 10
   1.4 Towards an Engineering Approach to Agile Service Development .... 13

2 Agility .................................................................................................... 17
   2.1 Introduction .................................................................................. 17
   2.2 Common Agile Methods ................................................................. 19
   2.3 Operationalizing Agility ................................................................. 21
   2.4 Business Drivers for Agility ......................................................... 29

3 Agile Architecture .................................................................................. 41
   3.1 Introduction .................................................................................. 41
   3.2 Architecture to Manage Agility ..................................................... 44
   3.3 Architecture Processes in an Agile Context ................................. 53
## Agile Service Development (2/3)

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td><strong>Service Modelling</strong></td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>4.1 Introduction</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>4.2 The Role of Models in Agile Service Development</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>4.3 Adoption Levels of Modelling</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>4.4 The ASD Framework</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>4.5 The ASD Conceptual Model</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td>4.6 Model Integration</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>4.7 Requirements for Tool Support</td>
<td>93</td>
</tr>
<tr>
<td>5</td>
<td><strong>Patterns for Agility</strong></td>
<td>95</td>
</tr>
<tr>
<td></td>
<td>5.1 Introduction</td>
<td>95</td>
</tr>
<tr>
<td></td>
<td>5.2 Conceptual Model</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>5.3 Pattern Classification</td>
<td>98</td>
</tr>
<tr>
<td></td>
<td>5.4 Pattern Catalogue</td>
<td>101</td>
</tr>
<tr>
<td></td>
<td>5.5 Example: Multichannel Management Patterns</td>
<td>103</td>
</tr>
<tr>
<td></td>
<td>5.6 Patterns at Work</td>
<td>108</td>
</tr>
<tr>
<td>6</td>
<td><strong>An Agile Way of Working</strong></td>
<td>111</td>
</tr>
<tr>
<td></td>
<td>6.1 Introduction</td>
<td>111</td>
</tr>
<tr>
<td></td>
<td>6.2 A Situational Approach to an Agile Way of Working</td>
<td>113</td>
</tr>
<tr>
<td></td>
<td>6.3 Practices, Goals and Stakeholders</td>
<td>114</td>
</tr>
<tr>
<td></td>
<td>6.4 Constructing A Situational Way of Working</td>
<td>122</td>
</tr>
</tbody>
</table>
Agile Service Development (1/3)

6.5 Step 1: Identify Situational Factors, Goals and Practices ........................................ 124
6.6 Step 2: Select Agile Practices .................................................................................. 129
6.7 Step 3: Combine Practices ..................................................................................... 130
6.8 Step 4: Execute the Way of Working ...................................................................... 139
6.9 Step 5: Reflect on the Way of Working .................................................................... 139

7 Stakeholder Communication .......................................................................................... 141
7.1 Introduction ................................................................................................................ 141
7.2 Communication Situations ....................................................................................... 146
7.3 Communication Setups ............................................................................................ 152
7.4 Communication Needs and Capabilities of Stakeholders ........................................ 158
7.5 Model Visualization Guidelines .............................................................................. 165
7.6 Communication Practices ......................................................................................... 170

8 Adopting Agile Service Development .......................................................................... 177
8.1 Barriers to Agility ..................................................................................................... 177
8.2 Scaling Up Agile Processes ..................................................................................... 180
8.3 A Capability Model for Agile Service Development .............................................. 181
8.4 Concluding Remarks ............................................................................................... 188
Open Services Innovation

HENRY CHESBROUGH
Author of the best-selling book Open Innovation

OPEN Services INNOVATION
RETHINKING YOUR BUSINESS TO GROW AND COMPETE IN A NEW ERA

Book: January 2011
Supporting literature

- Book: *Model-Driven Software Development: Technology, Engineering, Management (Paperback)*
  by Thomas Stahl, Markus Voelter, Krzysztof Czarnecki

- *Engineering Service Oriented Systems: A Model Driven Approach*, Karakostas, Bill; Zorgios, Yannis
  April 2008
Supporting literature – EMF and GMF

  - Dave Steinberg (Author), Frank Budinsky (Author), Marcelo Paternostro (Author), Ed Merks (Author)

- Book: Eclipse Modeling Project: A Domain-Specific Language (DSL) Toolkit (Paperback)
  - Richard C. Gronback
Value Network Analysis

http://www.valuenetworksandcollaboration.com
Kermeta – www.kermeta.org

Kermeta - Breathe life into your metamodels

Kermeta is developed by the Triskell team, all external contributors are welcome.

Quick links
- Documentation folder
- Download
- Development area
- Development control board
- Community

Kermeta is also part of Topcased
Software engineering practices and methods

- modelbased.net
- practices.modelbased.net

- A practices framework, SEMAT, [www.semat.org](http://www.semat.org)
Book is available now – SafariBooksonline/Addison Wesley
The Kernel is a stripped-down, lightweight set of definitions that captures the essence of effective, scalable software engineering in a practice independent way.

The Kernel is described using a small subset of the Language.
Alphas: Example

Requirements

Description
What the software system must do to address the opportunity and satisfy the stakeholders.

It is important to discover what is needed from the software system, share this understanding among the stakeholders and the team members, and use it to drive the development and testing of the new system.

Associations

scopes and constrains: Work

- Conceived
- Bounded
- Coherent
- Acceptable
- Addressed
- Fulfilled

- The need for a new system has been agreed.
- The purpose and theme of the new system are clear.
- The requirements provide a coherent description of the essential characteristics of the new system.
- The requirements describe a system that is acceptable to the stakeholders.
- Enough of the requirements have been addressed to satisfy the need for a new system in a way that is acceptable to the stakeholders.
- The requirements that have been addressed fully satisfy the need for a new system.
Activity Spaces: The Essential Things to Do

Explore Possibilities
Understand Stakeholder Needs
Ensure Stakeholder Satisfaction
Use the System

Understand the Requirements
Shape the System
Implement the System
Test the System
Deploy the System
Operate the System

Prepare to do the Work
Coordinate Activity
Support the Team
Track Progress
Stop the Work
Update to the course in 2013

- We will start with a focus on enterprise architecture and business architecture and service innovation and business process modeling, illustrated with various modeling tools - compared with UML

- We will make the second part on Model Driven Engineering smaller and practical – i.e. make an editor

- In part three we will focus on practical model driven development with UML and UML profiles and with the transformations to platform environments for service oriented architecture (SOA) and cloud computing.
New content started in 2012, continued in 2013

- Continued focus on “Service Science” – with Service Innovation (with BI/NHH), Service Design (OAH) and Service Engineering (UIO)

- VNA – Value Network Analysis, Verna Allee
- New standard: OMG MDA standard: VDML, FACESEM
- Service Design: AT ONE
- Business Model Innovation: Osterwalder/Lindgren
Which OMG modeling standards will you learn?

- UML 2.0 – what is new in version 2
- VDML – Value Delivery Modeling Language – with VNA
- SoaML – SOA Modeling Language
- MDA – Model Driven Architecture
- BPMN 2.0 – Business Process Modeling Notation
- BMM _ Business Motivation Model
- SysML – Systems Engineering Modeling Language
- SPEM – Software Process Engineering Metamodel
- QVT, MOF2Text – Query, View, Transformation

See www.omg.org
Which tools/environments will you learn?

- UML modeling tools
- AT ONE – Service Design
- VNA – Value Network Analysis
- BMI – Business Model Innovation/Generation
- Balsamiq – UI Mockups – for further UI modeling
- MagicDraw with UML and BPMN
- Eclipse EMF and XMI, Principles of GMF
- EuGENia – for GMF
- EPF/SPEM Software Process Modeler
- Overview of ATL, KerMeta, OpenArchitectureWare-OAW, …
Requirements for the course

- Student at UIO

- Only assumption is basic knowledge of UML and Java (but not necessarily UML 2.0)
Exam

- Case-based (ref. earlier exams)
- All written material can be used

- 4 hours

- Monday June 3rd, 2013, 4 hours
OMG Model-Driven Architecture (MDA)

www.omg.org/mda
Automation in Software Development

- **Requirements**: Manually implement
  - **Source in a general-purpose language, e.g., Java or C++**: Compile
    - Implementation
  - **Source in domain-specific language (DSL)**: Compile
    - Implementation

- **High-level spec (functional and nonfunctional)**: Implement with Interactive, automated support
  - **Source in domain-specific language (DSL)**: Compile
    - Implementation

---

**Source in domain-specific language (DSL)**: Compile
- Implementation

**Source in a general-purpose language, e.g., Java or C++**: Compile
- Implementation

**Source in domain-specific language (DSL)**: Compile
- Implementation
MDA CIM, PIM and PSM/Code

Computational Independent Model

CIM

ATL

BPMN, POP*, ARIS, ArchiMate, GERAM, GRAI, Zachman, UEML, B.Rules

Platform Independent Model

PIM

MOFScript

BPDM, SBVR, EDOC, UPMS, PIM4SOA, ODM

Platform Specific Model/Code

PSM

ADM

UML profiles and metamodels for Java JEE, BPEL, WSDL, XML, XPDL, OWL-S, WSML, WSDL-S

ADM

BPEL, WSDL, XML, XPDL, OWL-S, WSML, WSDL-S

Code, Java JEE, ....
What is Enterprise Modelling?

**Enterprise Modelling (EM)** is a capability for externalising, making and sharing enterprise knowledge.

EM tools can either be:
- used stand-alone to produce various kinds of model views,
- integrated as front-ends to other systems,
- part of an environment providing a contextual user-environment.
Why Enterprise Architecture?

- How can I involve my people in improving the performance of the business?
- How can I use best practices to ensure the success of the business?
- How can I ensure that the IS technology helps the work of my people?
Representations of Architecture

ARIS

ZACHMAN

GERAM

EN/ISO 19439

Athena OEA

NIST
Three Views in DOD Architecture Framework and C4ISR-AF

Operational View
- Identifies What Needs to be Accomplished and Who Does It
  - Operational Requirements
  - Basic Technology Capabilities
  - Technical Standards

Systems View
- Relates Systems and Characteristics to Operational Needs
  - Specific System Capabilities Required to Satisfy Information Exchanges

Technical Standards View
- Prescribes Standards and Conventions
  - Technical Standards Criteria Governing Interoperable Implementation/Procurement of the Selected System Capabilities
Zachman Framework – for Enterprise Architecture (IBM, 1987)

<table>
<thead>
<tr>
<th>VA Enterprise Architecture</th>
<th>DATA What</th>
<th>FUNCTION How</th>
<th>NETWORK Where</th>
<th>PEOPLE Who</th>
<th>TIME When</th>
<th>MOTIVATION Why</th>
<th>Based on work by John A. Zachman</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCOPE (CONTEXTUAL) Planner</td>
<td>Things Important to the Business</td>
<td>Processes Performed</td>
<td>Business locations</td>
<td>Important Organizations</td>
<td>Events Significant to the Business</td>
<td>Business Goals and Strategy</td>
<td></td>
</tr>
<tr>
<td>SYSTEM MODEL (LOGICAL) Designer</td>
<td>Logical Data Model</td>
<td>Application Architecture</td>
<td>Distributed System Architecture</td>
<td>Human Interface Architecture</td>
<td>Processing Structure</td>
<td>Business Rule Model</td>
<td></td>
</tr>
<tr>
<td>TECHNOLOGY MODEL (PHYSICAL) Builder</td>
<td>Physical Data Model</td>
<td>System Design</td>
<td>Technology Architecture</td>
<td>Presentation Structure</td>
<td>Control Structure</td>
<td>Rule Design</td>
<td></td>
</tr>
<tr>
<td>DETAILED REPRESENTATIONS (OUT-OF-CONTEXT) Sub-Contractor</td>
<td>Data Definition</td>
<td>Program</td>
<td>Naknek Architecture</td>
<td>Security Architecture</td>
<td>Timing Definition</td>
<td>Rule Design</td>
<td></td>
</tr>
<tr>
<td>FUNCTIONING ENTERPRISE Sub-Contractor</td>
<td>Data</td>
<td>Function</td>
<td>Naknek</td>
<td>Organization</td>
<td>Schedule</td>
<td>Strategy</td>
<td></td>
</tr>
</tbody>
</table>

Telecom and Informatics
IT/Business architecture transformation

As-Is Business Architecture

- Motivations
- Organization Unit
- Capabilities
- Information/Intelligence Value
- Business Processes
- Business Architecture Ecosystem

As-Is IT Architecture

- Application Architecture
- Data Architecture
- Technical Architecture
- Business Synergy
- Application Architecture

To-Be Business Architecture

- Motivations
- Organization Unit
- Capabilities
- Information/Intelligence Value
- Business Processes
- Business Architecture Ecosystem

To-Be IT Architecture

- Application Architecture
- Data Architecture
- Technical Architecture
- Business Synergy
- Application Architecture

Business Transformation

Synchronization of Business / IT Transformation

Synchronization of To-Be Business Architecture & To-Be IT Architecture

Business / IT As Is Architecture Mapping

- As-Is Business Architecture
- As-Is IT Architecture

IT Transformation

Telecom and Informatics

SINTEF
UPDM RFC - Domain Meta Model Summary

Products

Legend

| Individual | Type | Tuple | Thing |

TV

TV-18383

AV

RV-1

AO-2

OV

OV-1

OV-2

OV-3

OV-4

OV-4a

OV-4b

OV-5

OV-6

OV-7

StV

SV-1

SV-2

SV-3

SV-4

SV-5

SV-6

SV-7

SV-10b

SOV

SOV-1

SOV-2

SOV-3

SV-9

SV-11

SV-12
### Zachman with OMG standards

<table>
<thead>
<tr>
<th><strong>Scope</strong> (Contexts)</th>
<th><strong>Business</strong> (Concepts)</th>
<th><strong>System</strong> (Logic)</th>
<th><strong>Technology</strong> (Physics)</th>
<th><strong>Component</strong> (Assemblies)</th>
<th><strong>Operation</strong> (Instances)</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of things important to business</td>
<td>Semantic Model ODM, IMM (CWM)</td>
<td>Logical Data Model ODM, IMM (CWM), UML</td>
<td>Physical Data Model IMM (CWM), UML</td>
<td>Data Definition IMM (CWM), UML</td>
<td>Data</td>
</tr>
<tr>
<td>List of processes that the business performs</td>
<td>Business Process Model BPMN, CMPM</td>
<td>Application Architecture SoaML, UML</td>
<td>System Design SoaML, UML</td>
<td>Program UML</td>
<td>Function</td>
</tr>
<tr>
<td>List of locations which the business operates</td>
<td>Business Logistics System BPMN, CMPM</td>
<td>Distributed System Architecture SoaML, UML</td>
<td>Technology Architecture SoaML, UML</td>
<td>Network Architecture UML</td>
<td>Network</td>
</tr>
<tr>
<td>List of organizations important to the business</td>
<td>Workflow Model OSM, BPMN, CMPM</td>
<td>Human Interface Architecture BPMN, CMPM</td>
<td>Presentation Architecture</td>
<td>Security Architecture</td>
<td>Organization</td>
</tr>
<tr>
<td>List of events/cycles important to the business</td>
<td>Master Schedule BPMN, CMPM, DTFV</td>
<td>Process Structure BPMN, CMPM, DTFV</td>
<td>Control Structure BPMN, CMPM, DTFV</td>
<td>Timing Definition DTFV</td>
<td>Schedule</td>
</tr>
<tr>
<td>List of business goals/strategies</td>
<td>Business Plan SBVR</td>
<td>Business Rule Model SBVR</td>
<td>Rule Design SBVR</td>
<td>Rule Definition SBVR</td>
<td>Strategy</td>
</tr>
</tbody>
</table>
OMG standards coverage

Scope (Contexts)

Business (Concepts)

System (Logic)

Technology (Physics)

Component (Assemblies)

Operation (Instances)

Data (What)

Function (How)

Network (Where)

People (Who)

Time (When)

Motivation (Why)

List of things important to business

List of processes that the business performs

List of locations which the business operates

List of organizations important to the business

List of events/cycles important to the business

List of business goals/strategies

BMM

VDM

OSM

SBVR

ODM

BPMN

CMPM

IMM (CWM)

SoaML

UML

DTFV

BMM

SBVR

Telecom and Informatics
Models on Different Abstraction Levels

Fig. 17. The framework’s abstraction levels.
Agile Service Development Framework
CSI Norway – Center for Service Innovation
led by Norwegian Business School, Bergen, SINTEF, AOH, Telenor,..
(National Center for Research Driven Innovation (SFI) – 20 Meuro budget, 8 years 2011 – 2018)
CSI planned Activities

<table>
<thead>
<tr>
<th>Long term</th>
<th>Research themes</th>
<th>Short and intermediate term</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foresight/Insight</td>
<td>Innovations in customer and brand experiences</td>
<td>Innovation projects in WPs</td>
</tr>
<tr>
<td>Theme Governance</td>
<td>Co-creation and open innovation processes</td>
<td>Innovation projects in WPs</td>
</tr>
<tr>
<td></td>
<td>Business model innovations</td>
<td>Innovation projects in WPs</td>
</tr>
<tr>
<td></td>
<td>Infrastructure- and structural innovations</td>
<td>Innovation projects in WPs</td>
</tr>
<tr>
<td></td>
<td>New themes ...</td>
<td>Innovation projects in WPs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Innovation projects in WPs</td>
</tr>
</tbody>
</table>

Synthesis
Business Model Innovation

The Alexander Osterwalder canvas

Telecom and Informatics
Strategyzer (Osterwalder)
BMI – Canvases/Models
Actors

Services are often delivered by complex collaborations of actors in the form of a value network. There is considerable opportunity to be gained from innovating services based upon new actor constellations.

Touch-points

Services are delivered across multiple touch-points over time. Often, touch points are not exploited well, or are poorly coordinated. Focus upon touch-points and how new touch-points can be integrated allows a new view of service provision.

Offering

Services are usually based upon a core offering, although not all companies understand what their core offering actually is. By describing a companies projected offering and how this is perceived by the market, new services can be developed.

Need

Services should be based upon customer needs, wants and desires. This part of the method uses this as an innovation start-point.

Experience

Services are experiential in nature and experiences can be designed and staged. By defining desired experiences and developing a vocabulary for this, we hope that services can be developed from experience-pull rather than the traditional technology-push.
VDML – Value Modeling
Standard proposal, Nov. 2012

The Value Delivery Metamodel (VDM) RFP [OMG Document bml/2009-03-09] solicits proposals for a metamodel specification for modeling customer value delivery, based on the concept of a value networks and value chains.


NEFICS, represented by Cordys, Aalborg University and SINTEF, has been involved in the development of the joint revised proposal for Value Delivery Modeling Language (VDML) version 1.0 [OMG Document bml/2011-11-06].

- Official OMG document: http://www.omg.org/cgi-bin/doc?bml/12-11-06 (Restricted to OMG members)

This revised proposal is a joint submission by submitters: Cordys Corporation B.V., CSC and supporters: Aalborg University, Adaptive, Agile Enterprise Design, AT&T, BizAgi, Ltd., Fujitsu, Mega International, Ministry of Defense, Netherlands, Oei, Works, Vlastuin Group and XIBIX.
ServiceML Editor

- Web-based modelling editor
  - [http://tomcat.thingml.org/backend/poem/repository](http://tomcat.thingml.org/backend/poem/repository)
- User guide
- Currently being extended to support AT ONE Method (i.e., the Service Innovation practice)
A – Actors

Value Network (VDML)

Services Architecture (SoaML)

Hybrid notation
• Participants (from Value Network)
• Conversation (from BPMN 2.0)
• Groups a set of Flows
T – Touchpoints

Service Journey Map
- **Service Journey**: Chronological mapping (from the customer point of view) of a service encounter.
- Model as "Stages"
- Attach touchpoints to the different stages
- Library of different types of touchpoints to select from.
O – Offerings

Service Contract
- Detailing of the conversation.
- Conversation is the grouping of flows (messages).
- Service Contract defines the interfaces on both side (structure) and the protocol (behaviour) for how to use these interfaces.
N – Needs

Goals and Objectives
• We have not yet implemented modelling support for needs.
• One idea is to use a very small subset of BMM (Business Motivation Model) standard.
• **Goal:** a statement about a state or condition of the enterprise to be brought about or sustained through appropriate Means (i.e., Offerings expressed as Service Contracts).
• **Objective:** An Objective is a statement of an attainable, time-targeted, and measurable target that the enterprise seeks to meet in order to achieve its Goals.

Example of Goals and Objectives diagram
E – Experiences

• The idea is to extend the Service Journey Map.
• Each user/customer capture experiences (emotional icons) related to each touchpoints and deviations in the ideal/expected journey (seen from a Service Provider side).

Experiences

Ref. also

myServiceFellow smart phone app
Business Motivation Model (BMM) with MeansRealizations
What is BPMN (Business Process Modeling Notation)?

- BPMN is flow-chart based notation for defining Business Processes

- BPMN is an agreement between multiple modeling tools vendors, who had their own notations, to use a single notation for the benefit of end-user understand and training

- BPMN provides a mechanism to generate an executable Business Process (BPEL) from the business level notation
  - A Business Process developed by a business analyst can be directly applied to a BPM engine instead of going through human interpretations and translations into other languages
BPMN example

Patient
- Illness occurs
  - Doctor request
  - Send doctor request
  - Handle appointment
  - Handle symptoms
  - Handle prescription
  - Handle medicine
  - Send medicine
  - Receive prescription
  - Receive prescription pickup
  - Receive medicine

Receptionist
- Receive doctor request
- Send availability request
- Receive doctor availability
- Send booking
- Receive booking
- Send appointment
- Receive appointment
- Prepare prescription
- Receive prescription preparation
- Receive medicine request
- Send medicine

Doctor
- Receive availability request
- Send doctor availability
- Receive booking
- Receive symptoms
- Receive prescription preparation
- Receive prescription pickup
- Send prescription pickup
Extending COMET for SOA (1)
EPF Composer

EPF Composer is a tool platform for process engineers, project leads, project and program managers who are responsible for maintaining and implementing processes for development organizations or individual projects.

Aims to:

- provide for development practitioners a knowledge base of intellectual capital that allows them to browse, manage and deploy content.
- provide process engineering capabilities by supporting process engineers and project managers in selecting, tailoring, and rapidly assembling processes for their concrete development process.
**System and objects**

A *system* is a part of the real world which we choose to regard as a whole, separated from the rest of the world during some period of consideration.

A whole that we choose to consider as a collection of objects, each *object* being characterized by *attributes* and by *actions* which may involve itself and other objects.
Next Lecture, January 21, 2013
Service Innovation and Service Design

- Guest lecture: Marika Lüders
- AT ONE
- Service Innovation and Service Design