Specifying Services using the Service oriented architecture Modeling Language (SoaML): A baseline for Specification of Cloud-based Services

Brian Elvesæter\textsuperscript{1}, Arne-Jørgen Berre\textsuperscript{1} and Andrey Sadovykh\textsuperscript{2}

\textsuperscript{1} SINTEF ICT, P. O. Box 124 Blindern, N-0314 Oslo, Norway \texttt{brian.elvesater@sintef.no, arne.j.berre@sintef.no}

\textsuperscript{2} SOFTEAM, 21 avenue Victor Hugo, 75016 Paris, France \texttt{andrey.sadovykh@softeam.fr}
Outline

• Short presentation of the main part of the paper
  – What is SoaML?
  – Experiences and issues
  – Illustrative example
  – Two approaches

• Ongoing and future work
  – A baseline for Specification of Cloud-based Services
  – Towards a Cloud Modeling Language (CloudML)
What is SoaML?

- Service oriented architecture Modeling Language (SoaML)
  - Defines language constructs and extensions to UML2 to support service concepts (metamodel and UML profile)
  - Focuses on basic service modelling concepts and structure.
  - A foundation for further extensions and integration with BPMN, BMM and other metamodels.

- Key language constructs
  - Consumer
  - MessageType
  - Participant
  - Provider
  - ServiceContract
  - ServiceInterface
  - ServicesArchitecture
SoaML experiences, identified issues and purpose of our paper

• Our experiences with SoaML
  – Tooling
  – Methods and practices
  – Application in industry projects

• Identified issues
  – Inconsistencies in the specification.
  – Two (three) main approaches to service modelling.
  – Examples illustrating the two approaches are not consistent.
  – No clear separation, the two approaches are somewhat intertwined.
  – Tool support lacking or worse wrongly implemented.

• Purpose of our paper
  – Clarify the differences and similarities between the different approaches.
  – Describe how to align the approaches.
  – Position SoaML as a baseline for specification of cloud-based services.
  – SoaML can be extended with new modelling constructs and integrated with other modelling languages.
Consolidated and extended example

Services architecture:
- High level description of how participants work together for a purpose by providing and using services expressed as service contracts.

Service contract:
- Service specifications that define the roles each participant plays in the service and the interfaces they implement to play that role.
Two approaches

**ServiceContract (Collaboration)**

**Refinements or Views**

**Consumer (Interface)**
- quote()
- order()
- quoteRequest()
- orderConfirmation()

**Provider (Interface)**
- consumer:OrderPlacer
- provider:OrderTaker

**ServiceInterface (Class)**

**Type**

**Parameter**

**MessageType**
- QuoteRequest
- Order
- OrderConfirmation

**Enumeration**
- CurrencyType
  - EUR
  - USD
- ConfirmationType
  - Confirmed
  - Shipped
  - Cancelled
  - OutOfStock
A baseline for Specification of Cloud-based Services in REMICS
Model-Driven Migration of Legacy Applications to Service Cloud

- **Source Architecture**
  - Knowledge: REMICS KDM
    - Business Process and Rules Components: SoaML
    - Implementation: UML, U2TP

- **Recover**
  - Knowledge Discovery, Reverse Engineering
  - Source code, binaries, documentation, users knowledge, configuration files, execution logs and traces.

- **Model Driven Interoperability**
  - Service mediation for adaptation
  - SOA and Cloud Computing Patterns applied, Legacy Components Replacement and Wrapping, Design by Service Composition

- **Migrate**
  - Target Architecture for Service Cloud platform
  - SoaML with REMICS extensions for Service Clouds, Links to Business Models

- **Validate, Control and Supervise**
  - Forward MDA through PIM4Cloud
  - Model Transformation, Code Generation, Traceability

- **Legacy Artifacts**
  - RESERVOIR, Joyant, Amazon, Google, Microsoft

- **Models@Runtime for application management, Model Checking, Model-based Testing for validation**

**SINTEF ICT**
Cloud computing layers

• Cloud computing layers:
  – **Software as a Service** delivers software as a service over the Internet.
  – **Platform as a Service** delivers a computing platform and solution stack as a service.
  – **Infrastructure as a Service** delivers a platform virtualization environment as a service.

• Different issues for each layer:
  – Security
  – Quality of Service
  – Data Storage Interface
  – Client Application Interface
  – Provisioning
  – Development Platform
  – Virtual machine interface

• SoaML4Cloud (PIM4Cloud) can’t address all these issues
Cloud models and languages

- We are currently looking at some interesting models and languages:
  - Amazon Cloudformation
    - a textual description language for cloud resources
    - http://aws.amazon.com/cloudformation/
  - CA 3Tera AppLogic
    - a graphical language for Cloud configuration
  - Elastra – with DSLs for Cloud configuration
    - Elastra Cloud Modeling Language (ECML) is used to describe an application (software, requirements, and policies)
    - Elastra Deployment Modeling Language (EDML) is used to describe the resources (virtual machines, storage, and network) available in a data center.
    - www.elastra.com
# Cloud computing standardisation

<table>
<thead>
<tr>
<th>Standardisation organisation</th>
<th>Cloud standardisation</th>
<th>Members</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open Grid Forum</td>
<td>Open Cloud Computing Interface</td>
<td>Microsoft, Sun, Intel, HP, AT&amp;T, eBay, etc</td>
</tr>
<tr>
<td>Cloud Computing Interoperability Forum</td>
<td>Enable a global cloud computing ecosystem</td>
<td>Cisco, Intel, Thomson Reuters, Orange, Sun, IBM, RSA, etc</td>
</tr>
<tr>
<td>Distributed Management Task Force</td>
<td>Open Virtualisation Format Standard &amp; Open Cloud Standards Incubator</td>
<td>IBM, Microsoft, Novell, Oracle, Sun, Vmware, EMC, etc</td>
</tr>
<tr>
<td>Open Cloud Consortium</td>
<td>Standards and Interoperability for Large Data &amp; Open Cloud Testbed</td>
<td>Cisco, MIT Lincoln Labs, Yahoo, various colleges</td>
</tr>
<tr>
<td>Cloud Security Alliance</td>
<td>Best practices for providing security assurance</td>
<td>eBay, ING, Qualys, PGP, zScaler, etc</td>
</tr>
<tr>
<td>Storage Networking Industry Association</td>
<td>Storage Networking</td>
<td>Dell, EMC, Oracle, Juniper Networks, Qlogic, HP, Vmware, Hitachi, NetApp</td>
</tr>
<tr>
<td>Object Management Group</td>
<td>Modelling languages for Cloud computing</td>
<td>OMG members (industry consortium)</td>
</tr>
</tbody>
</table>
Future work: OMG standardisation goals

- Specify the REMICS SoaML4Cloud (PIM4Cloud) metamodel and profile
  - extension of the SoaML metamodel and profile
  - platform independent model
  - deployment modelling
  - support for code-generation aimed at cloud computing platforms

- Issue an OMG Request for Proposal (RFP) for CloudML
  - Focus on modelling deployment of applications & services on cloud for portability, interoperability and reuse
  - Address deployment to Cloud Platforms at the Infrastructure and Service level
  - Deployment model to specify infrastructure and QoS and SLA properties for analysis
Thanks for your attention!

- Questions?
- SoaML website:
  - http://www.soaml.org/
- SHAPE website:
  - http://www.shape-project.eu/
- NEFFICS website:
  - http://www.neffics.eu/
- REMICS website:
  - http://www.remics.eu/
- SiSaS website:
  - http://sisas.modelbased.net/