

Enterprise architecture frameworks with semantic models as a foundation for complex networked operations

SoaML – Service modeling

Semantic Days 2009, May 18th-20th, Stavanger, Norway

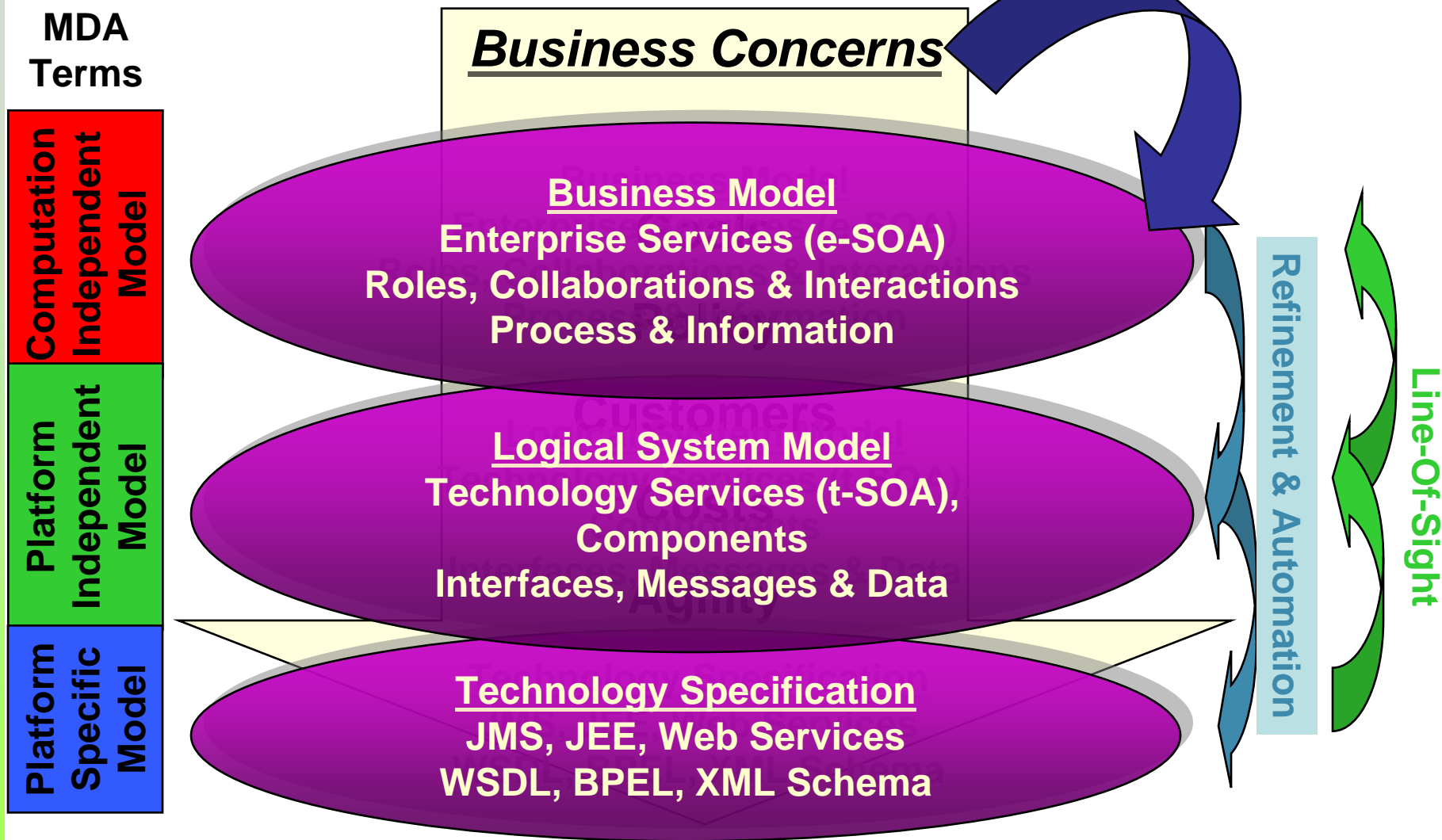
18.05.2009

Arne Jørgen Berre, SINTEF, Norway
Ulf Larsson, LFV, Sweden
Dima Panfilenko, DFKI IWi, Germany

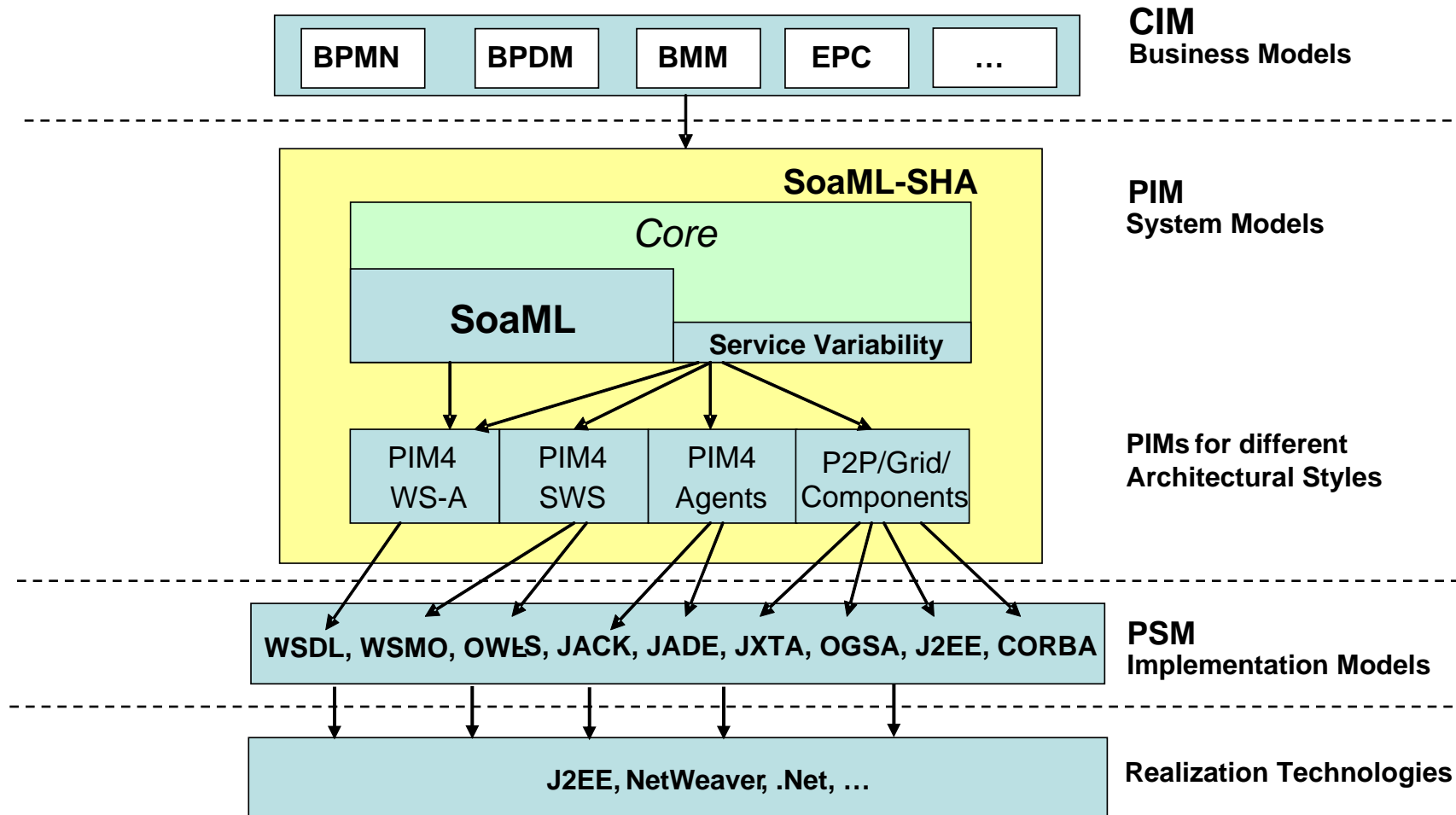
Agenda

- **(I) Enterprise Architecture, TOGAF, UPDM (Arne, Ulf, Dima)**
 - Zachman, TOGAF, MODAF/DODAF/NAF, MDA, UPDM - Arne
 - Saarstahl SHAPE - Dima
 - European ATM/SESAR - Ulf
- **(II) INFORMATION and ONTOLOGY MODELING (UML/ER, ODM/OWL with examples/tools) Arne (Ulf, Dima)**
 - Conceptual Modeling, Information Modeling, Ontologies - Ulf and Arne
 - ODM with OWL for semantic modeling (WSMT) - Dima
- **(III) PROCESS MODELING (EPC/BPMN with examples/tools) (Dima)**
 - ARIS/EPC (Event-Driven Process Chains) Dima
 - BPMN (Business Process Modeling Notation) Dima
- **(IV) SERVICE MODELING and Interoperability (SoaML with examples) (Arne)**
 - SoaML (Service oriented architecture Modeling Language) Arne
 - Semantic annotations, SAWSDL, from existing system specifications to an ontology can support semantic interoperability Arne

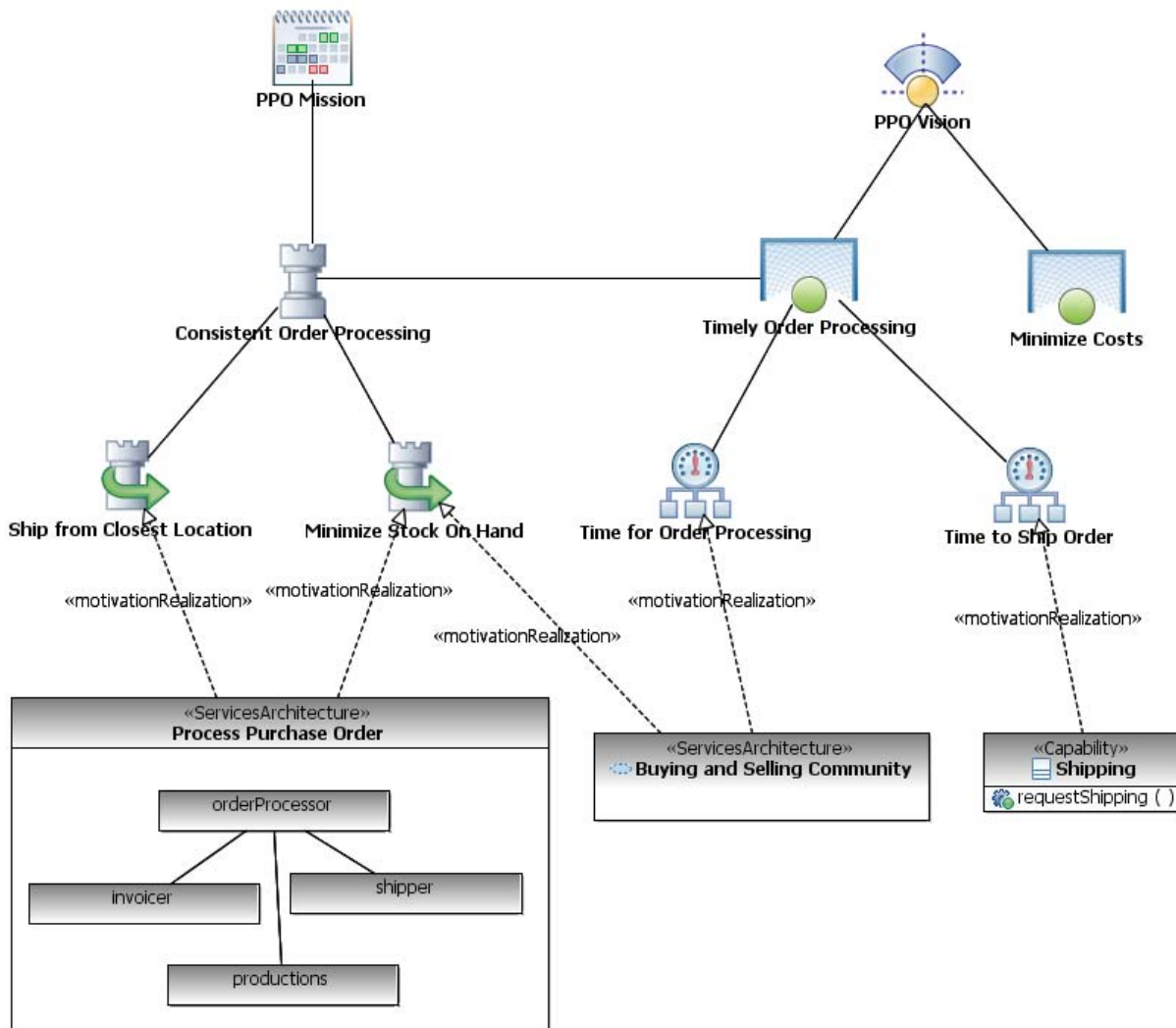
Business Focused SOA Using Model Driven Architecture



CIM – PIM - PSM



BMM with MeansRealizations



Service oriented architecture Modeling Language (SoaML) - Specification for the UML Profile and Metamodel for Services (UPMS)

Revised Submission

OMG document: ad/2008-08-04

Submitters

Adaptive
Capgemini
EDS
Fujitsu
Fundacion European Software Institute
Hewlett-Packard
International Business Machines
MEGA International
Model Driven Solutions
Rhysome
Softeam

Supporters

BAE Systems
STI/University of Innsbruck
DFKI
Everware-CBDi
France Telecom R&D
General Services Administration
Visumpoint
MID GmbH
NKUA - University of Athens
Oslo Software
SINTEF
THALES Group
University of Augsburg
Wilton Consulting Group

Primary Contact:

Arne J. Berre, SINTEF
email: Arne.J.Berre@sintef.no

Find the document here:

<http://www.omg.org/cgi-bin/doc?ad/08-11-01.pdf>

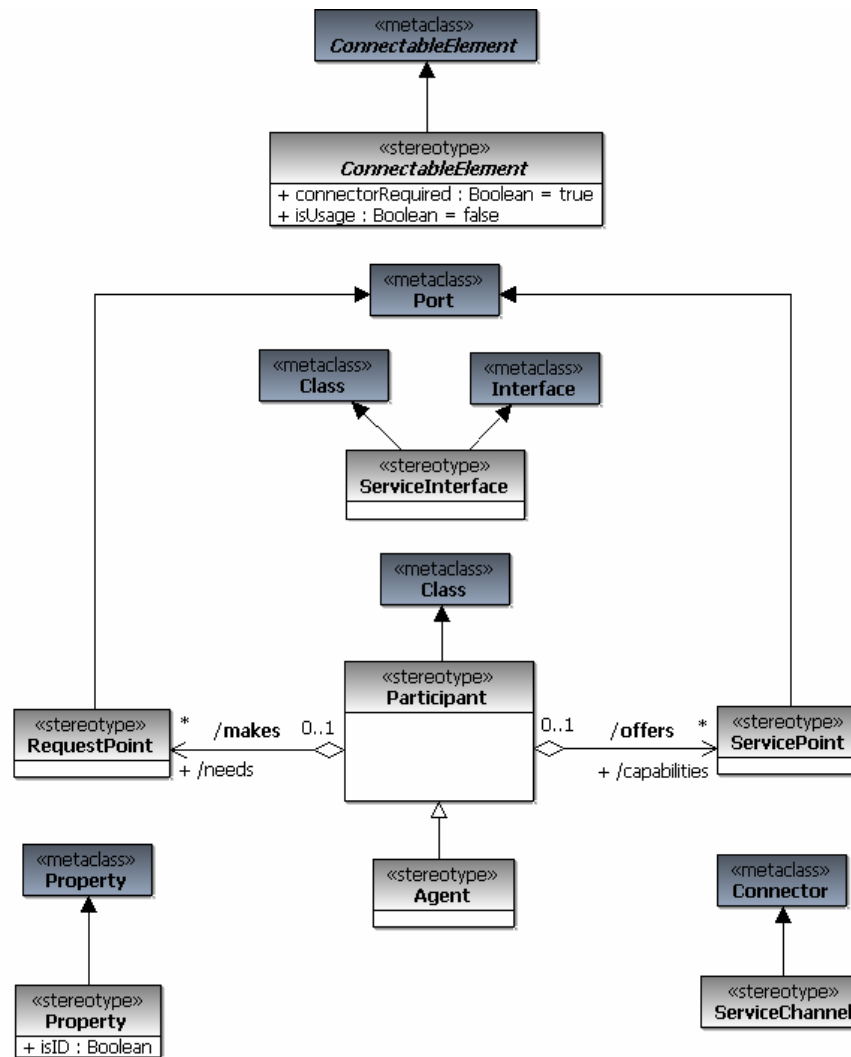
Revised submission per November 10th, 2008



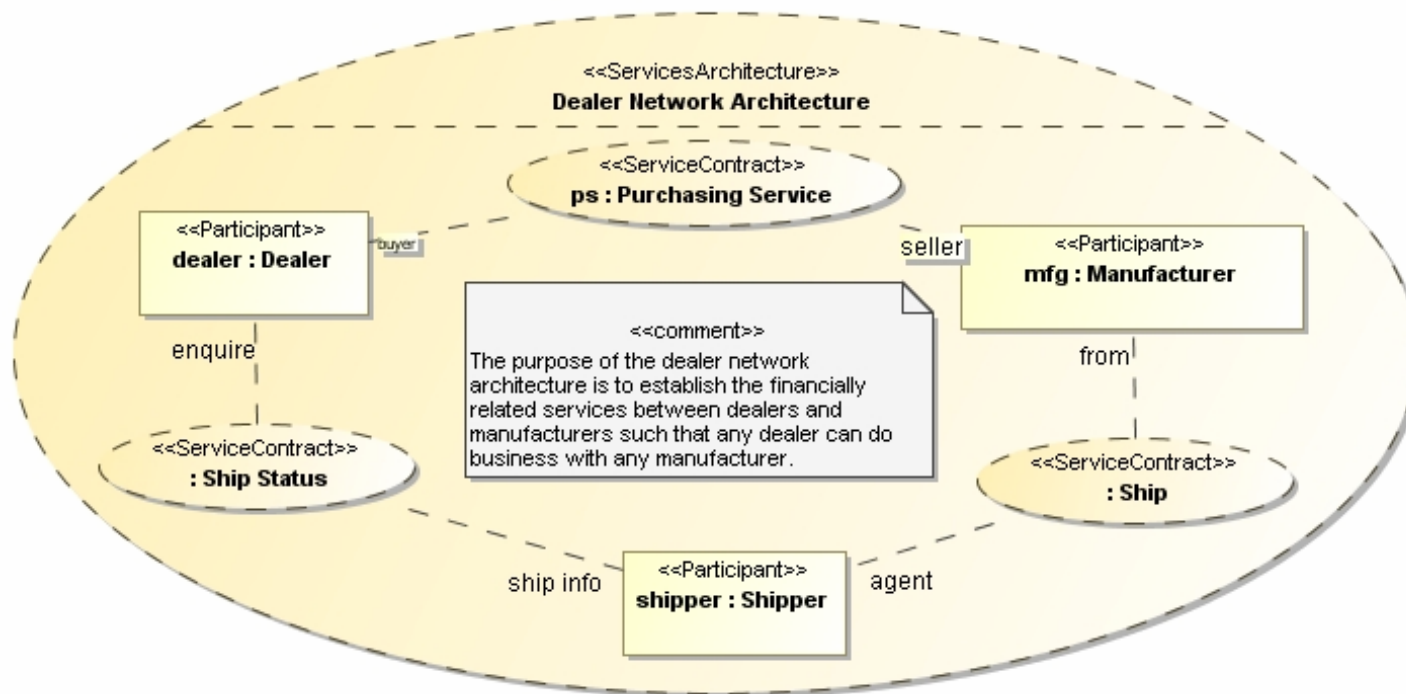
See also: www.soaml.org



SoaML UML Profile for ServiceInterface and Participants

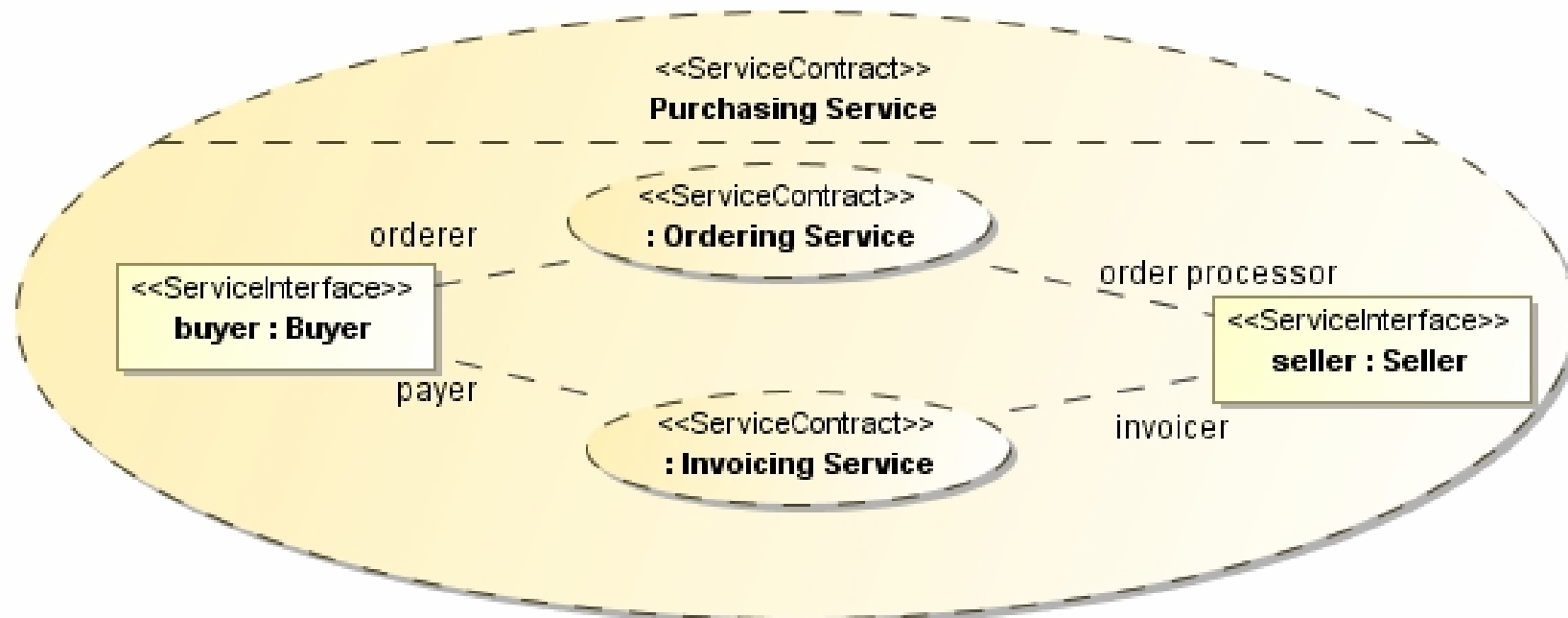


Services Architecture

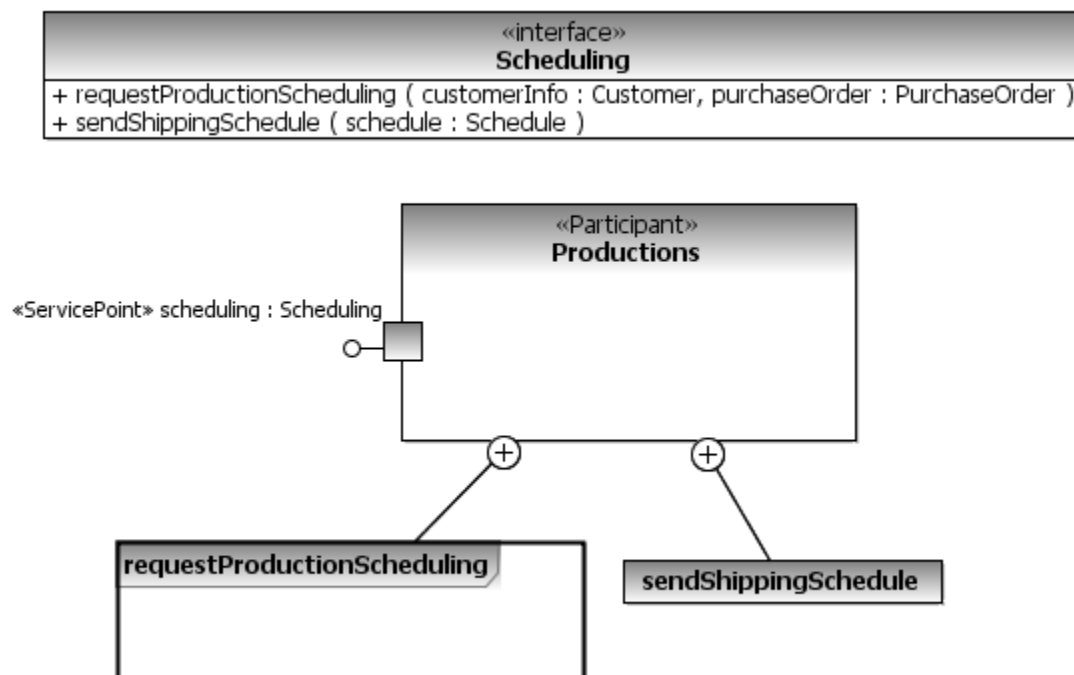


A ServicesArchitecture (or SOA) is a network of participant roles *providing* and *consuming services* to fulfill a purpose. The services architecture defines the requirements for the types of participants and service realizations that fulfill those roles.

Compound services



ServicePoints and Service Participants

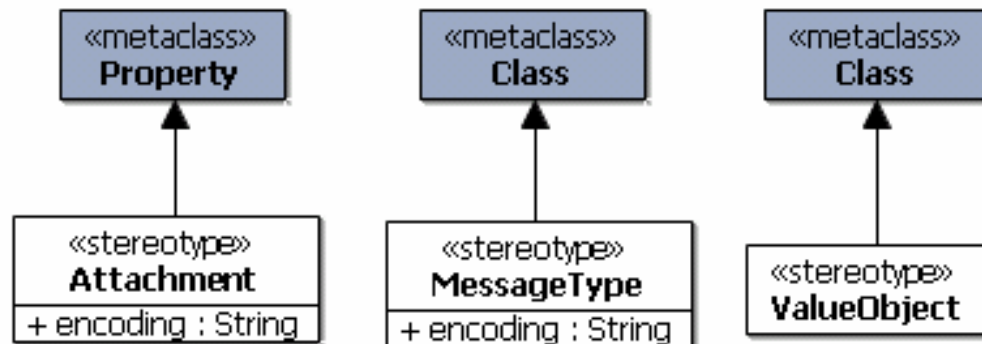


A ServicePoint is the offer of a service by one participant to others using well defined terms, conditions and interfaces. A ServicePoint defines the connection point through which a Participant offers its capabilities and provides a service to clients.

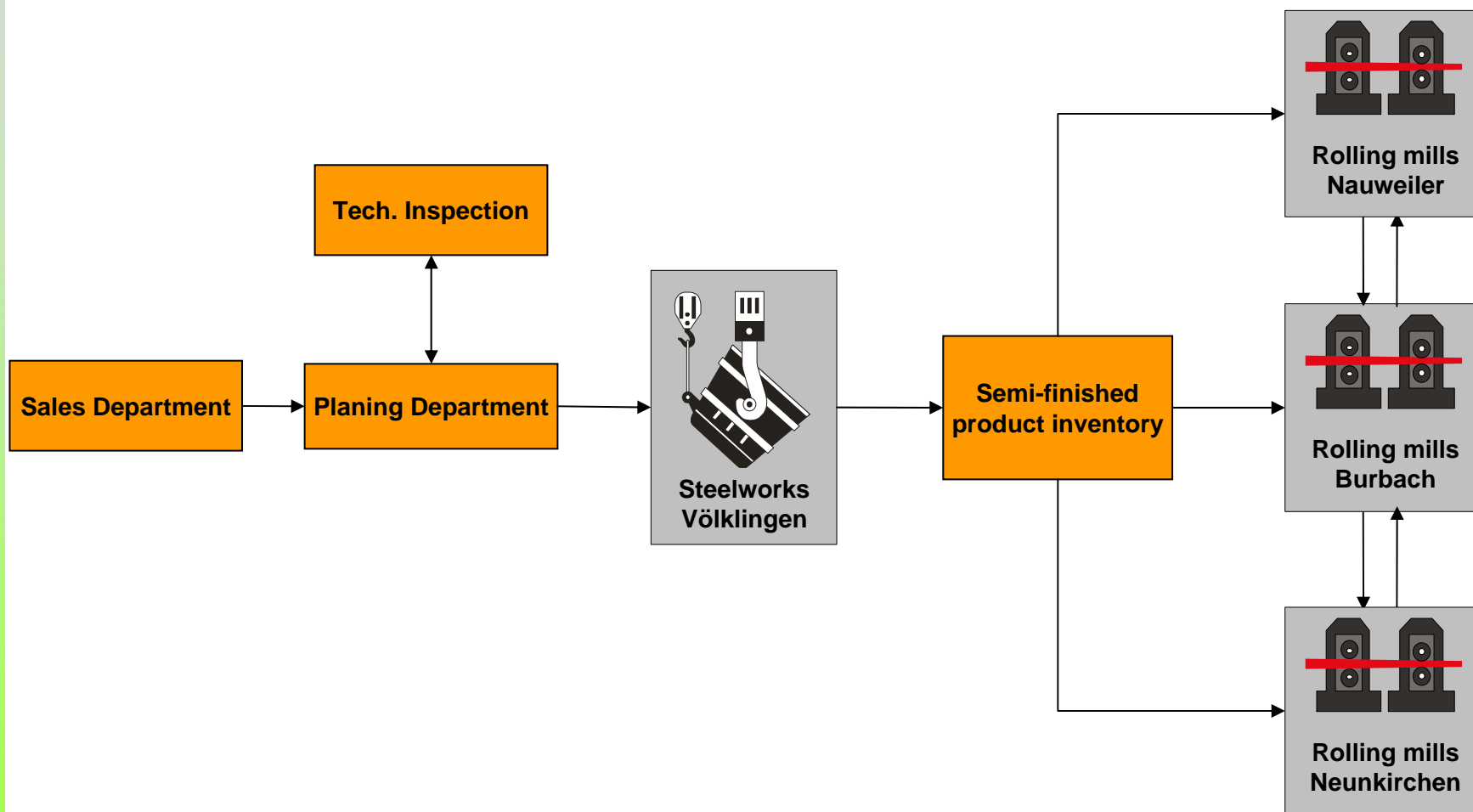
A ServicePoint is a mechanism by which a provider Participant makes available services that meet the needs of consumer requests as defined by ServiceInterfaces, Interfaces and ServiceContracts. A ServicePoint is represented by a UML Port on a Participant stereotyped as a «ServicePoint»,

Service Data

- Structural information exchanged between service consumers and service providers
- Attachments for opaque information
- Usage semantics make no assumptions with re control on,



Supply Chain of Saarstahl AG



18.05.2009

Arne Jørgen Berre, SINTEF, Norway
Ulf Larsson, LFV, Sweden
Dima Panfilenko, DFKI IWi, Germany

Participants

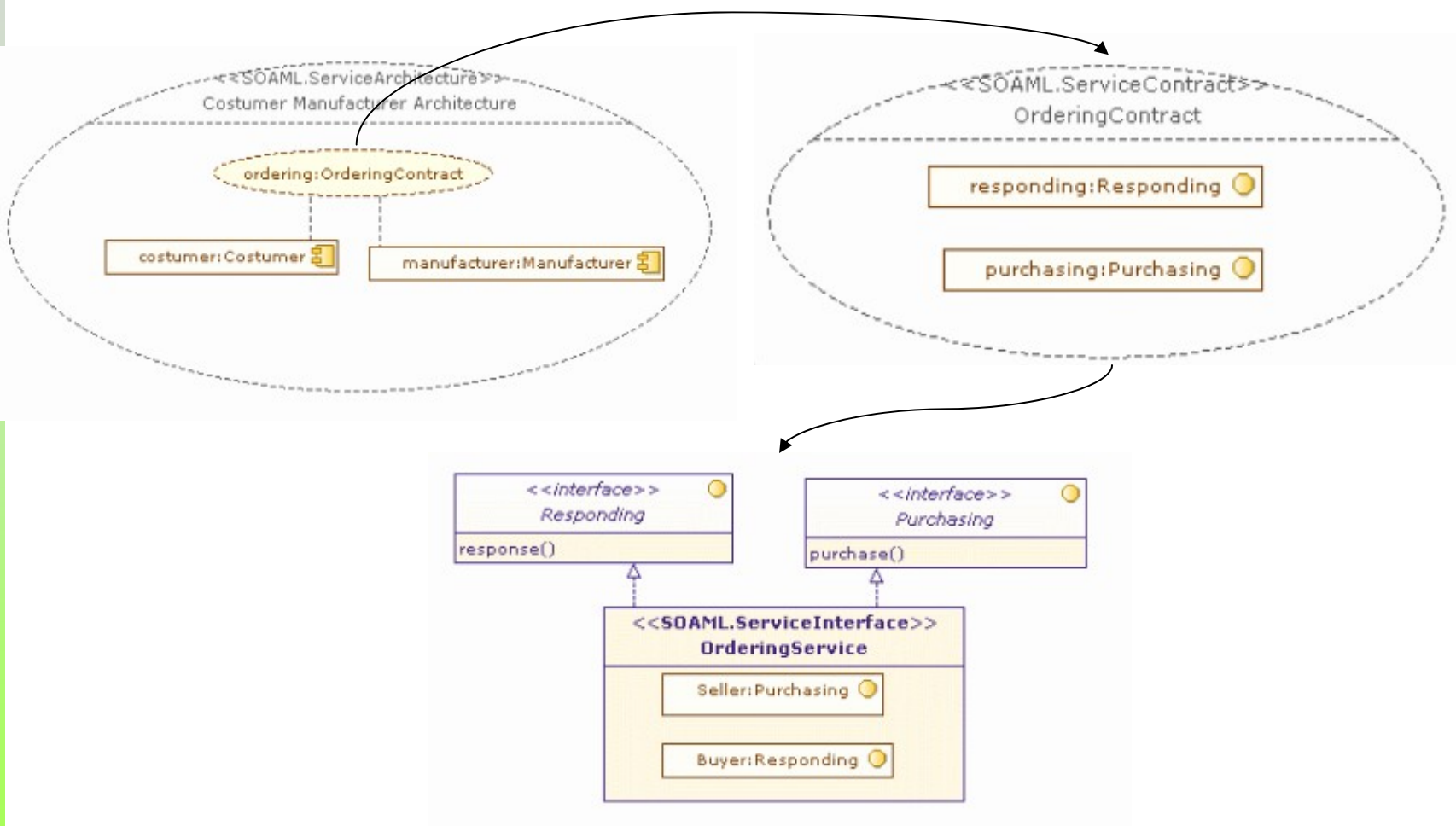
- Order Agent
 - Manages its working plan throughout production stages
 - Owns all necessary information
- Sales Department
 - Input of ordering service is forwarded to sales department
 - Creates instances of Order agents from this input
 - Can contact customer
- Technical Inspection
 - Offers service to check if there is a feasible working plan for some order
- Planning Department
 - Provides planning instructions that cannot be automated or are not yet automated inside the Order agent

Participants ctd.

- **Steelwork**
 - Offers service to request melting capacity
- **Rolling Mills**
 - Offer several information providing services (e.g. rolling date)
 - Service for detailed scheduling inside rolling group
- **Semi-finished Product Inventory**
 - Manages inventories and semi-finished material
 - Keeps track of bindings: Order to material

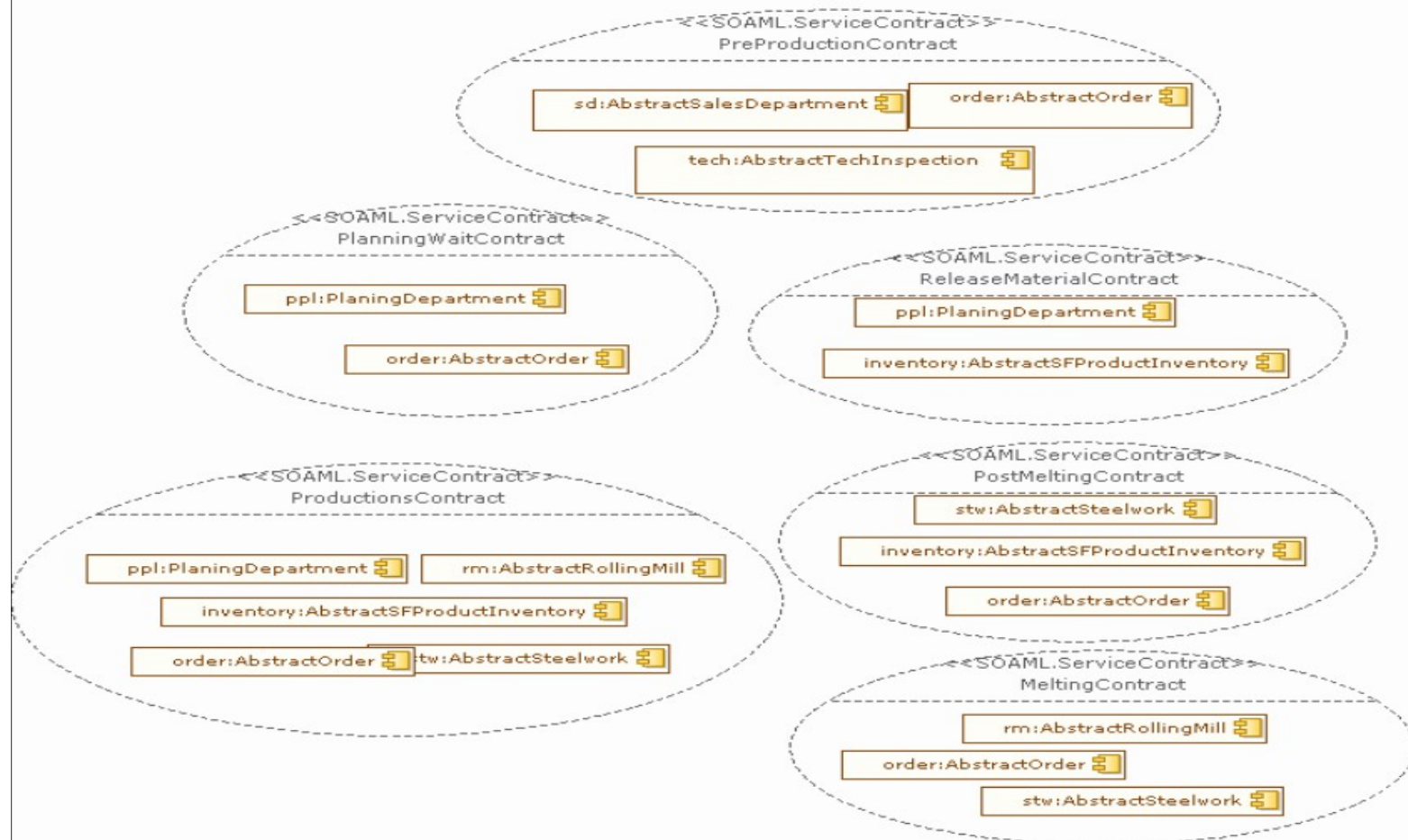
PIM Modelling

Service Architecture ~ Service Contract



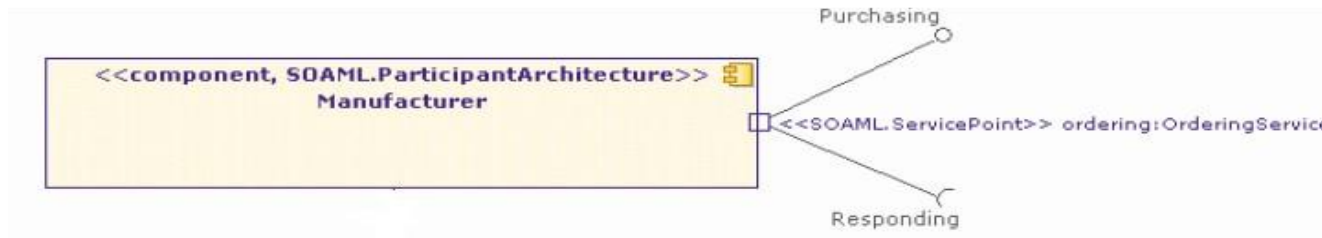
PIM Modelling

Set of Service Contracts



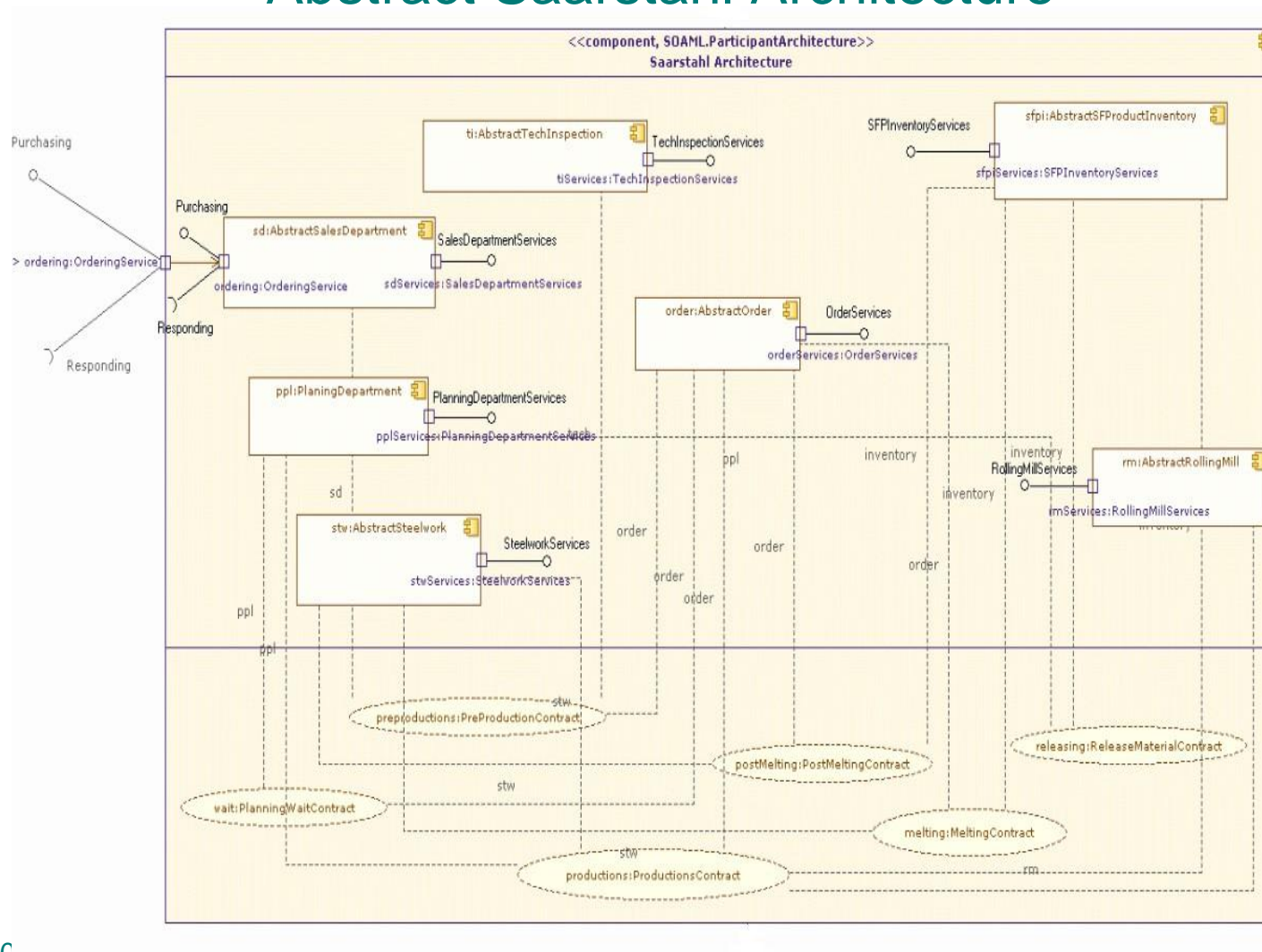
PIM Modelling

Participant Architecture



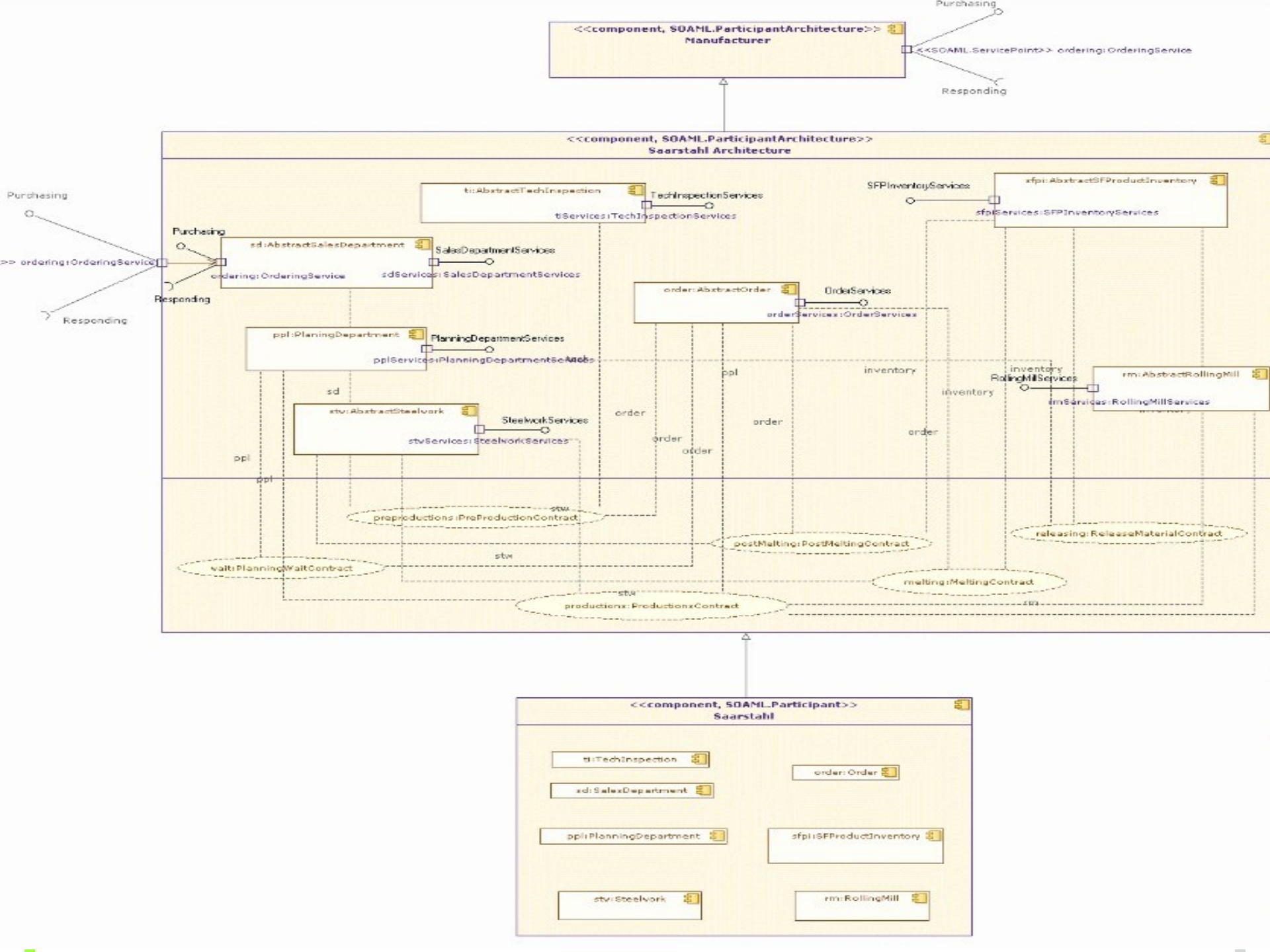
PIM Modelling

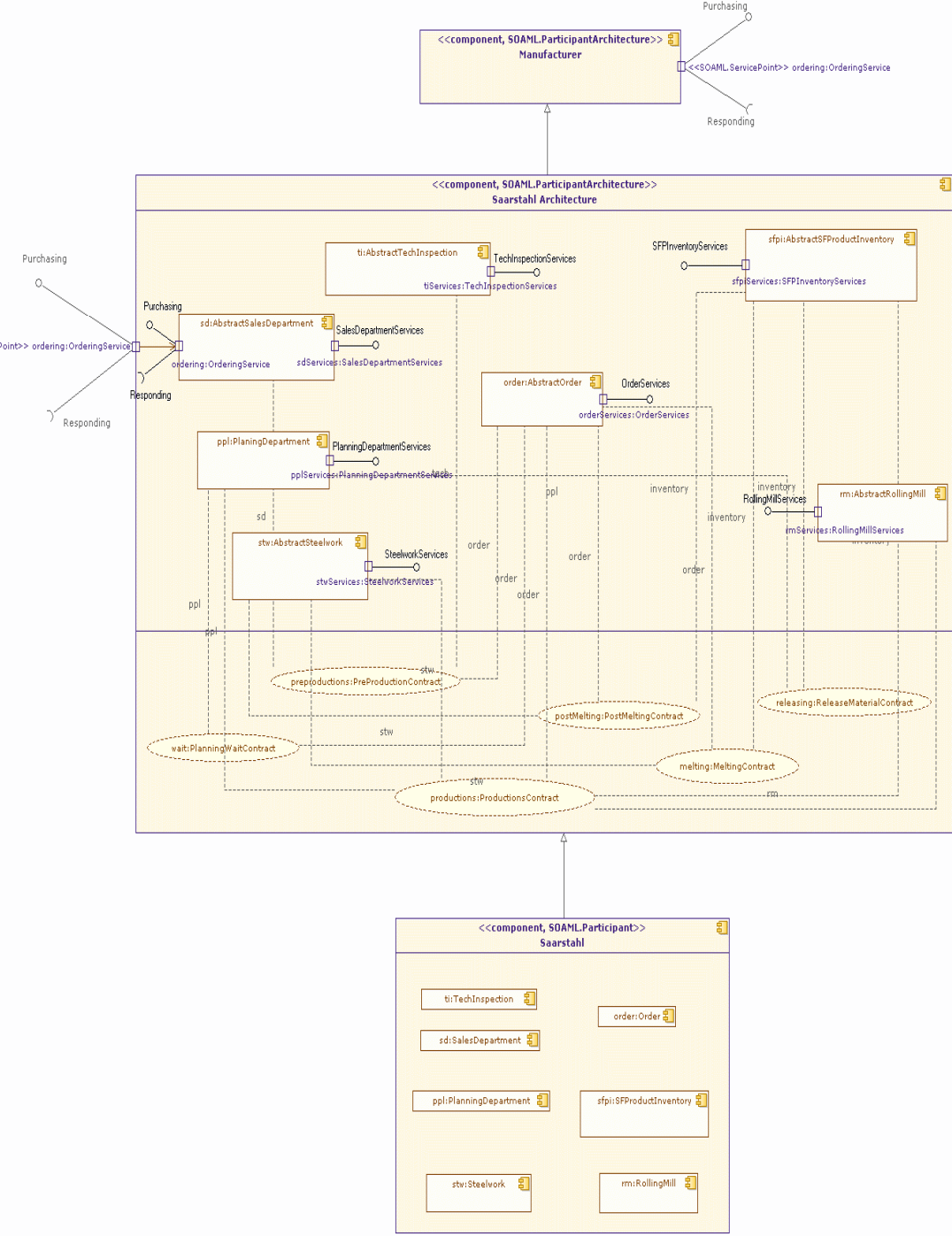
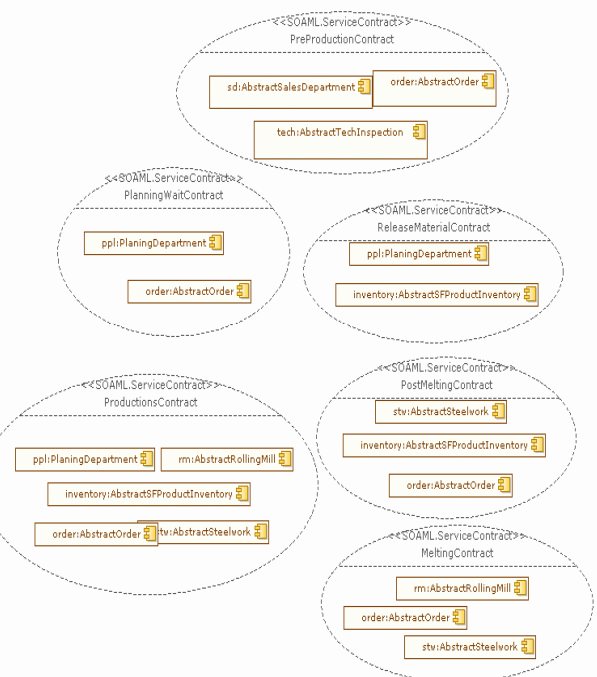
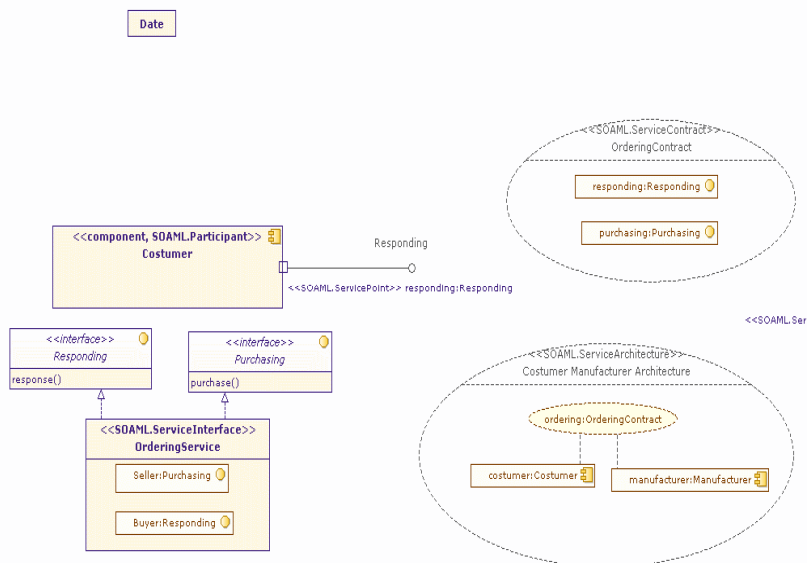
Abstract Saarstahl Architecture



18.05.2009

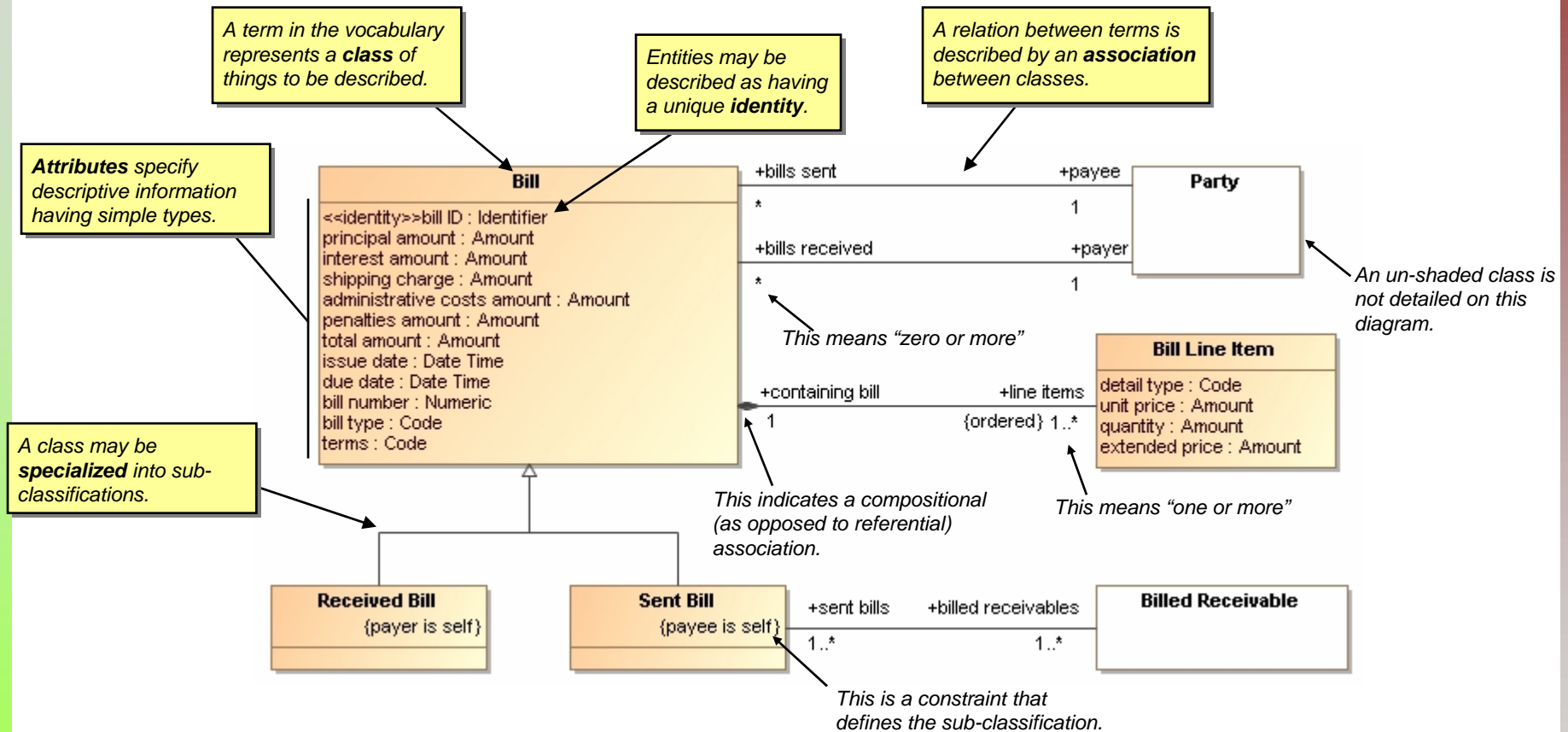
Ulf Larsson, LFV, Sweden
Dima Panfilenko, DFKI IWI, Germany





Information Modelling with UML/(SoaML) class models

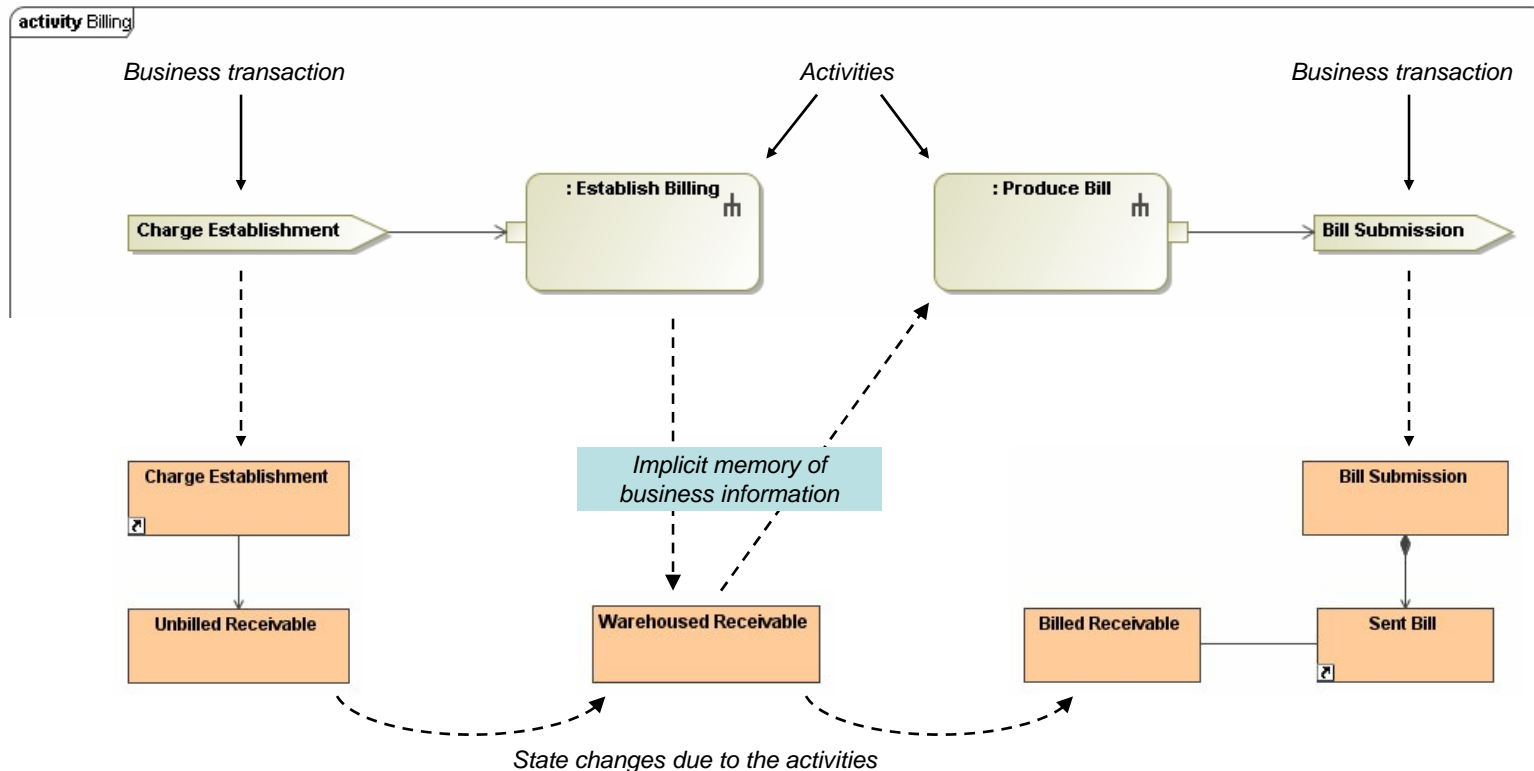
Information Model – For Messages and Entities



Integrating the Information Model with SOA

The **process model** describes how business activities are (or are to be) carried out.

Workflow

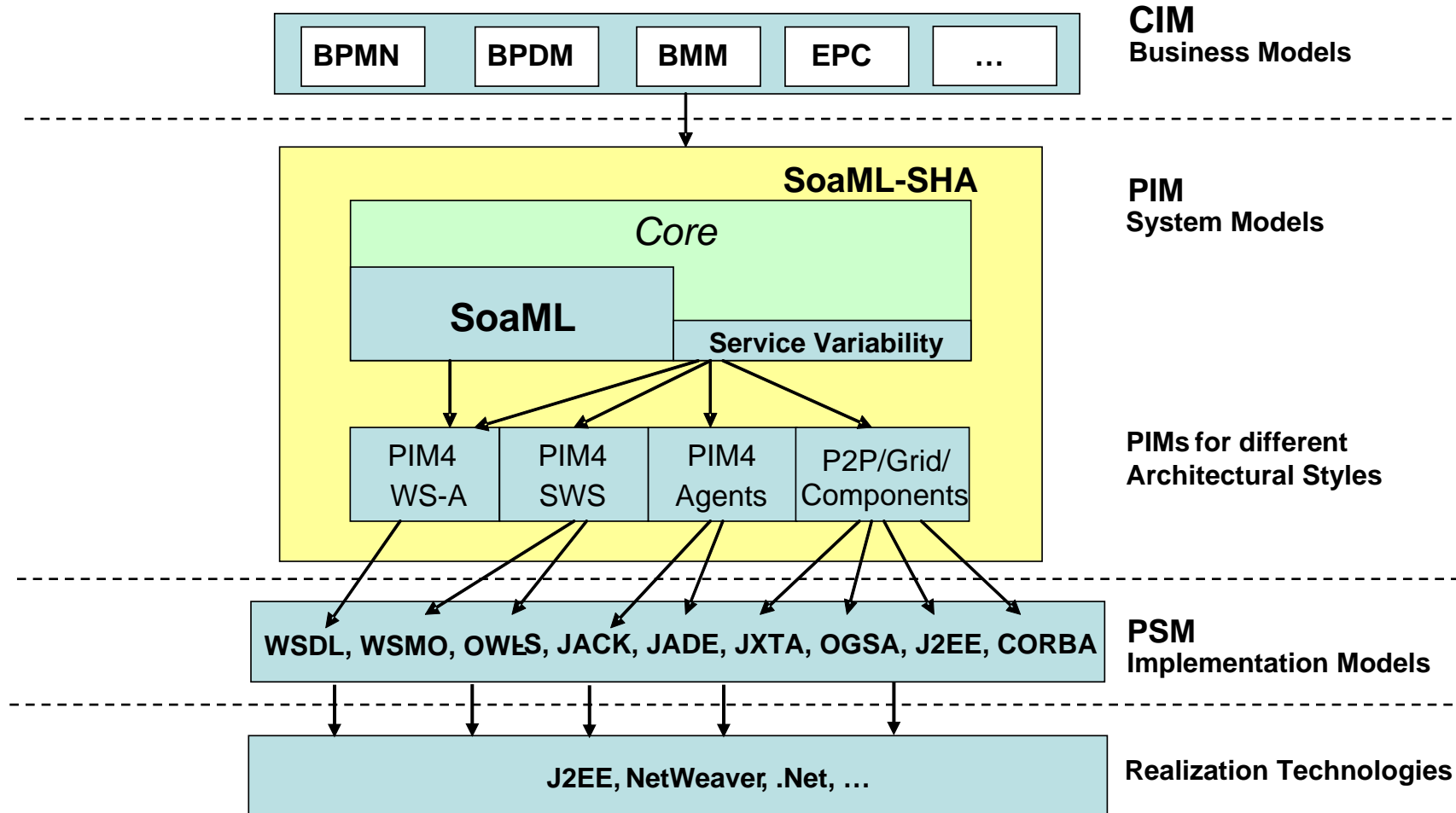


The **information model** details the vocabulary of the business entities and transactions used in the process model.

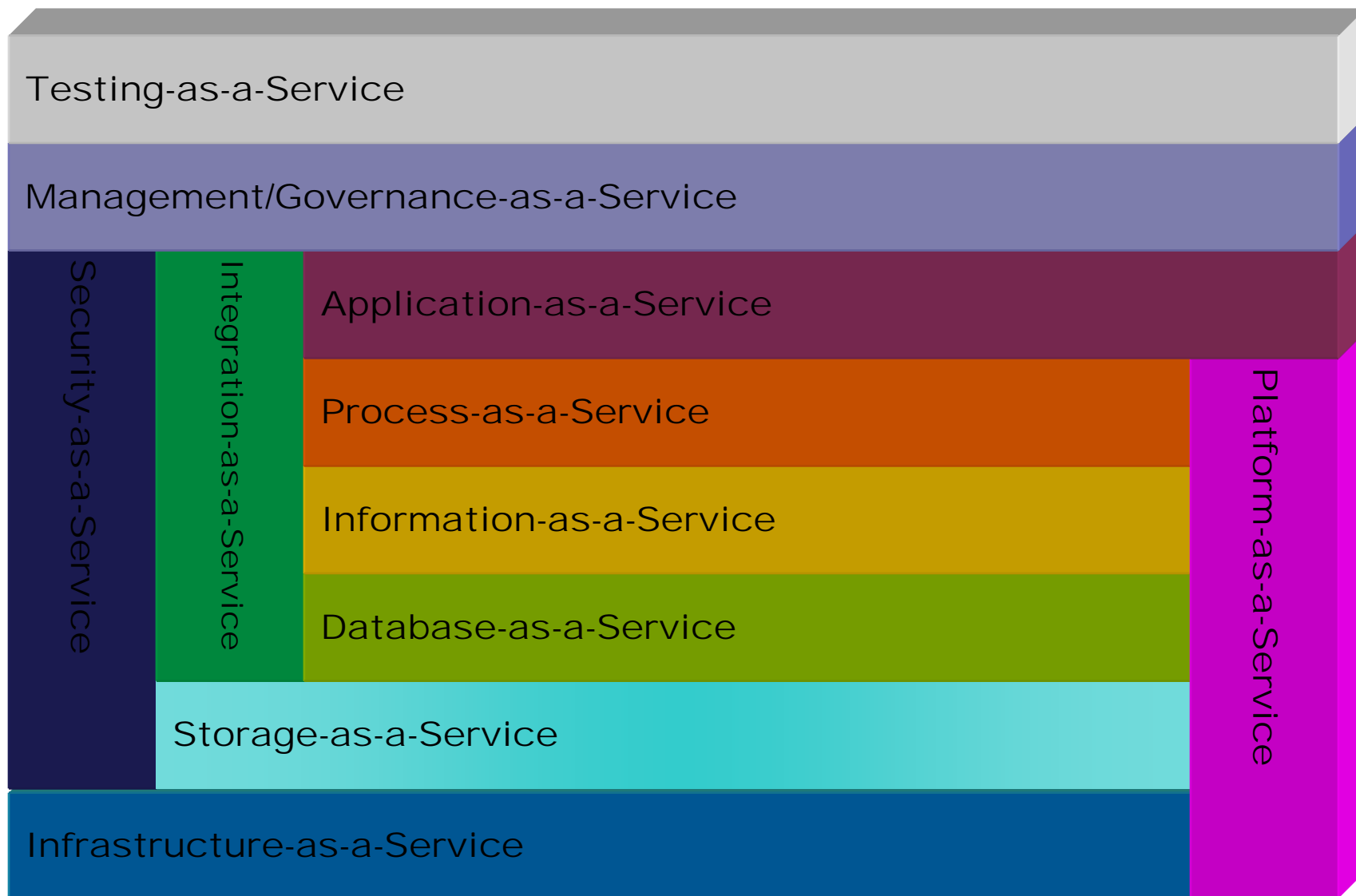
Everything as a service

Semantic Days 2009, May 18th-20th, Stavanger, Norway

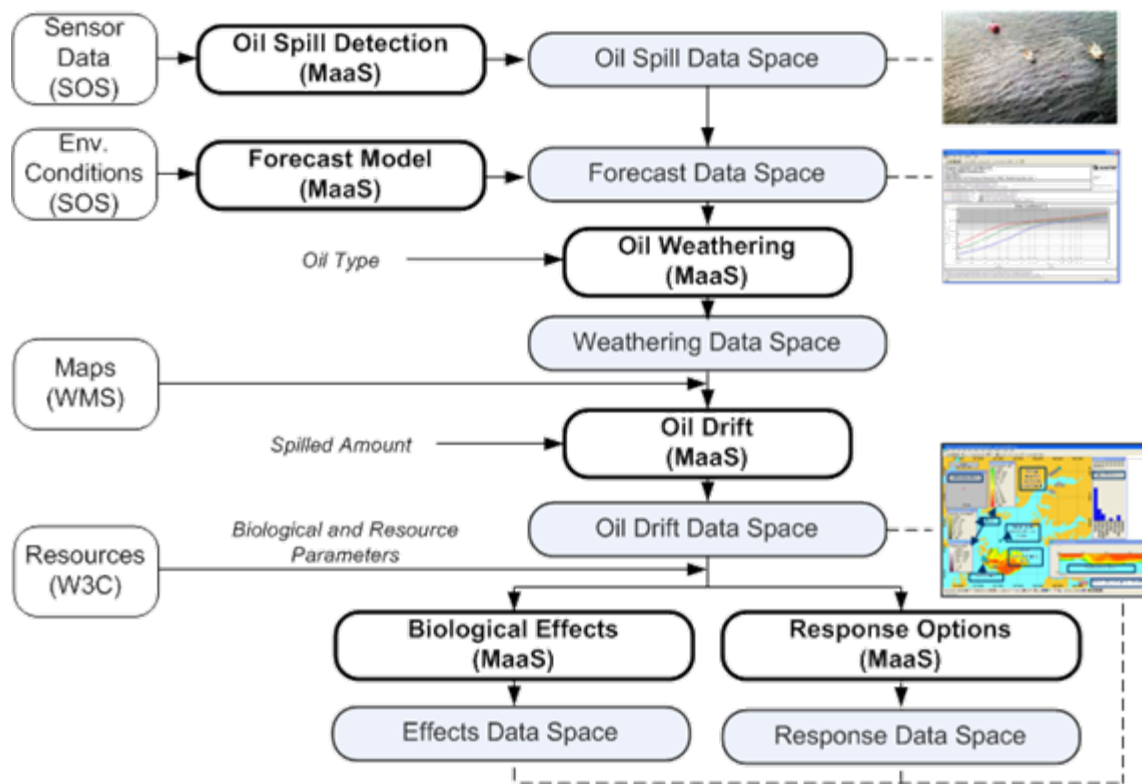
CIM – PIM - PSM



Clouds – everything as a service



ENVISION use case: Oil spill modelling as chained services



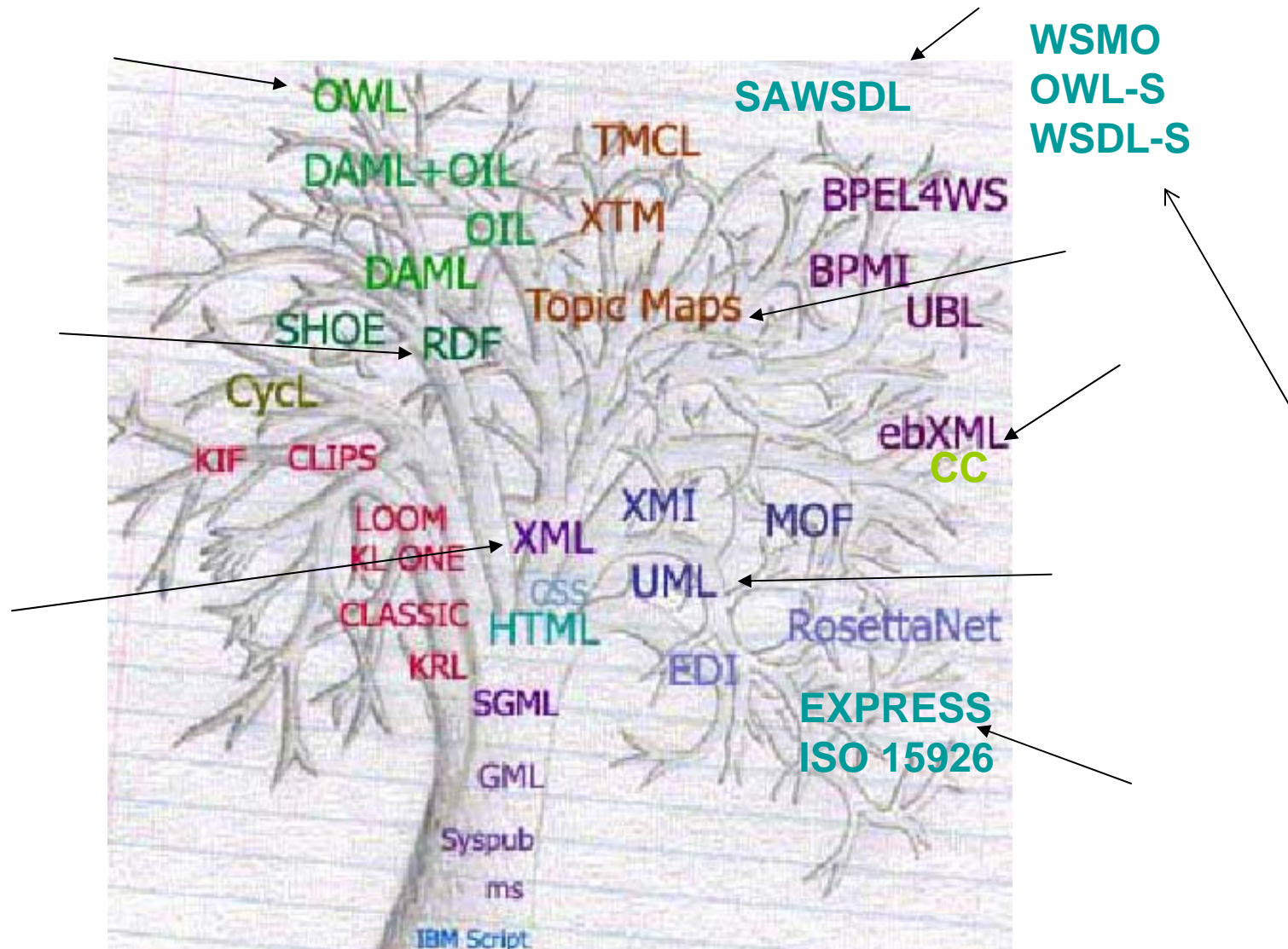
Semantic Web and Semantic Web Services

OWL-S
WSMO
SAWSDL

Semantic Days 2009, May 18th-20th, Stavanger, Norway

The Tree of Knowledge Technologies

(Extended from Top Quadrant)



Semantic web service technologies

- OWL-S (was DAML-S, US)
- WSMO (Europe, DERI, STI, OASIS)
- WSDL-S (basis for SAWSDL)
- SAWSDL (W3C standard)

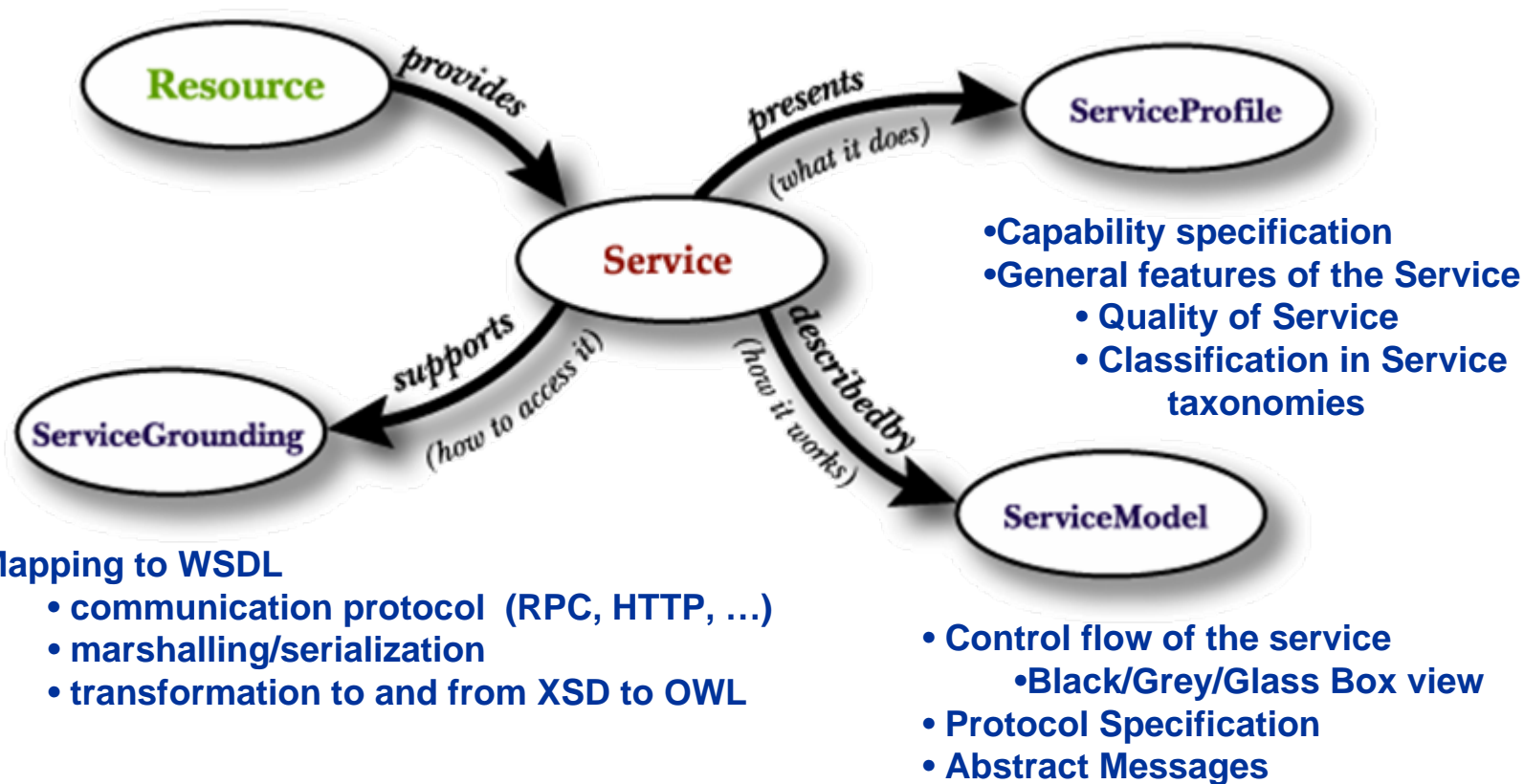
OWL-S Ontology

- OWL-S is an OWL ontology to describe Web services
- OWL-S leverages on OWL to
 - Support capability based discovery of Web services
 - Support automatic composition of Web Services
 - Support automatic invocation of Web services

"Complete do not compete"

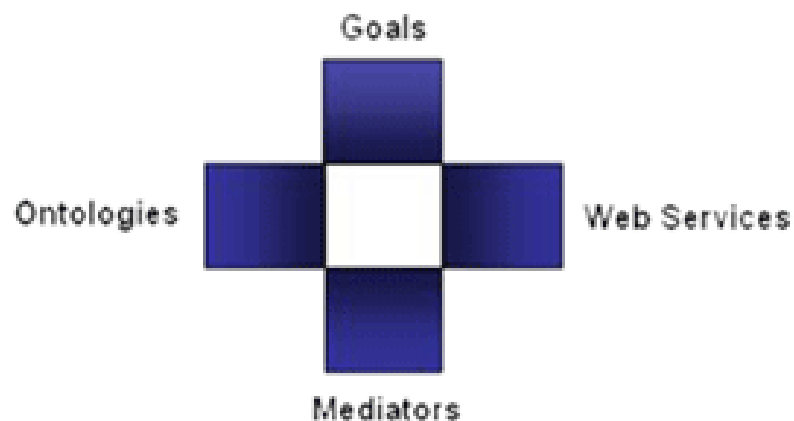
- OWL-S does not aim to replace the Web services standards

OWL-S Upper Ontology



The Web Service Modeling Ontology (WSMO)

Objectives that a client wants to achieve by using Web Services



Provide the formally specified terminology of the information used by all other components

Semantic description of Web Services:

- **Capability** (*functional*)
- **Interfaces** (*usage*)

Connectors between components with mediation facilities for handling heterogeneities

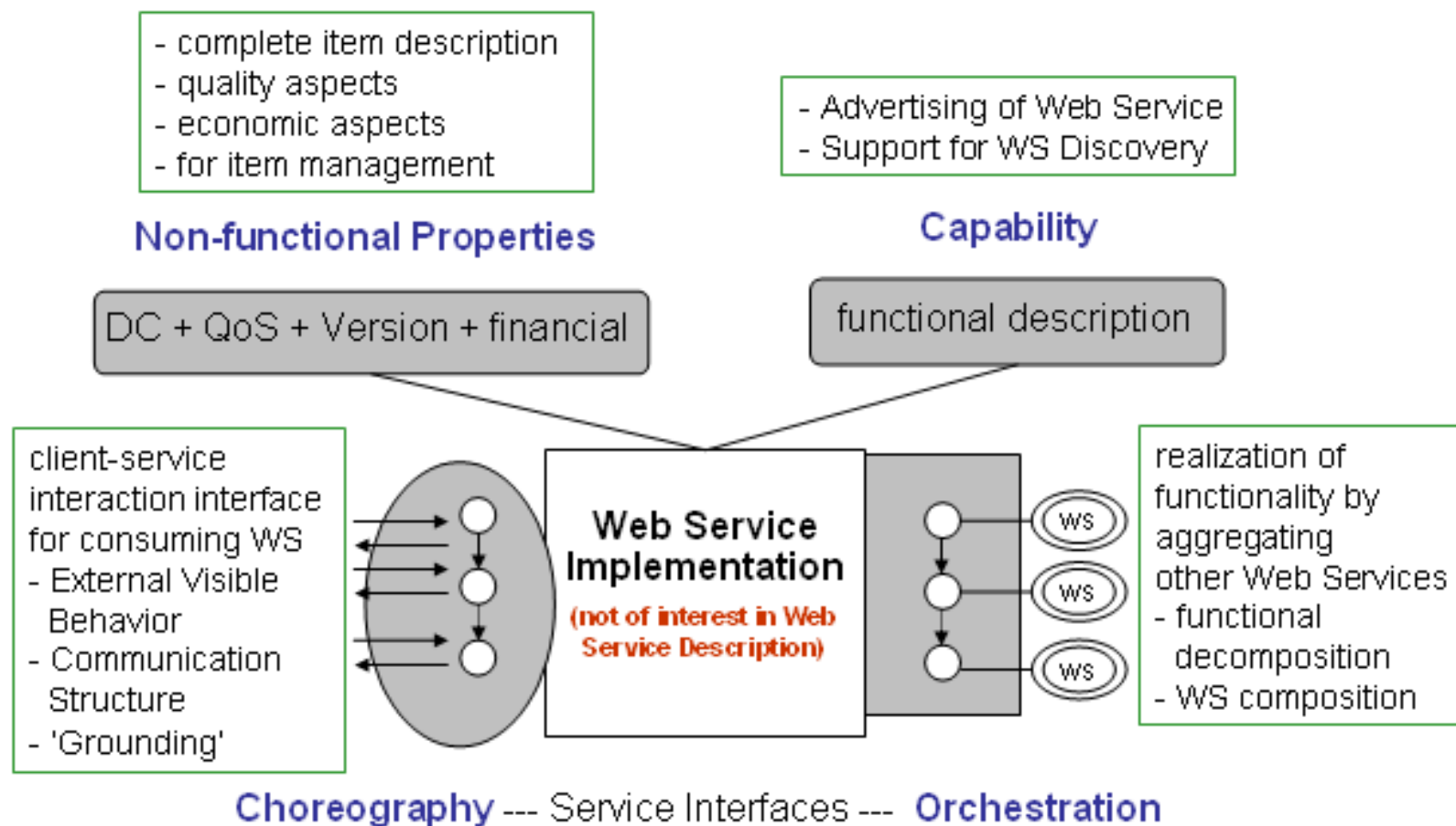
WSMO – Web Service Modeling Ontology

- WSMO working group includes the WSML working group, which aims at developing a language called Web Service Modeling Language (WSML) that formalizes the Web Service Modeling Ontology (WSMO).
- WSMO: an ontology called Web Service Modeling Ontology (WSMO) for describing various aspects related to Semantic Web Services. Taking the Web Service Modeling Framework (WSMF) as a starting point, we refine and extend this framework, and develop an ontology and a description language.
- WSML: aims developing a language called Web Service Modeling Language (WSML) that formalizes the Web Service Modeling Ontology (WSMO). Hereby, we have a two fold mission:
 - a) developing a proper formalization language for semantic web services and
 - b) providing a rule-based language for the semantic web

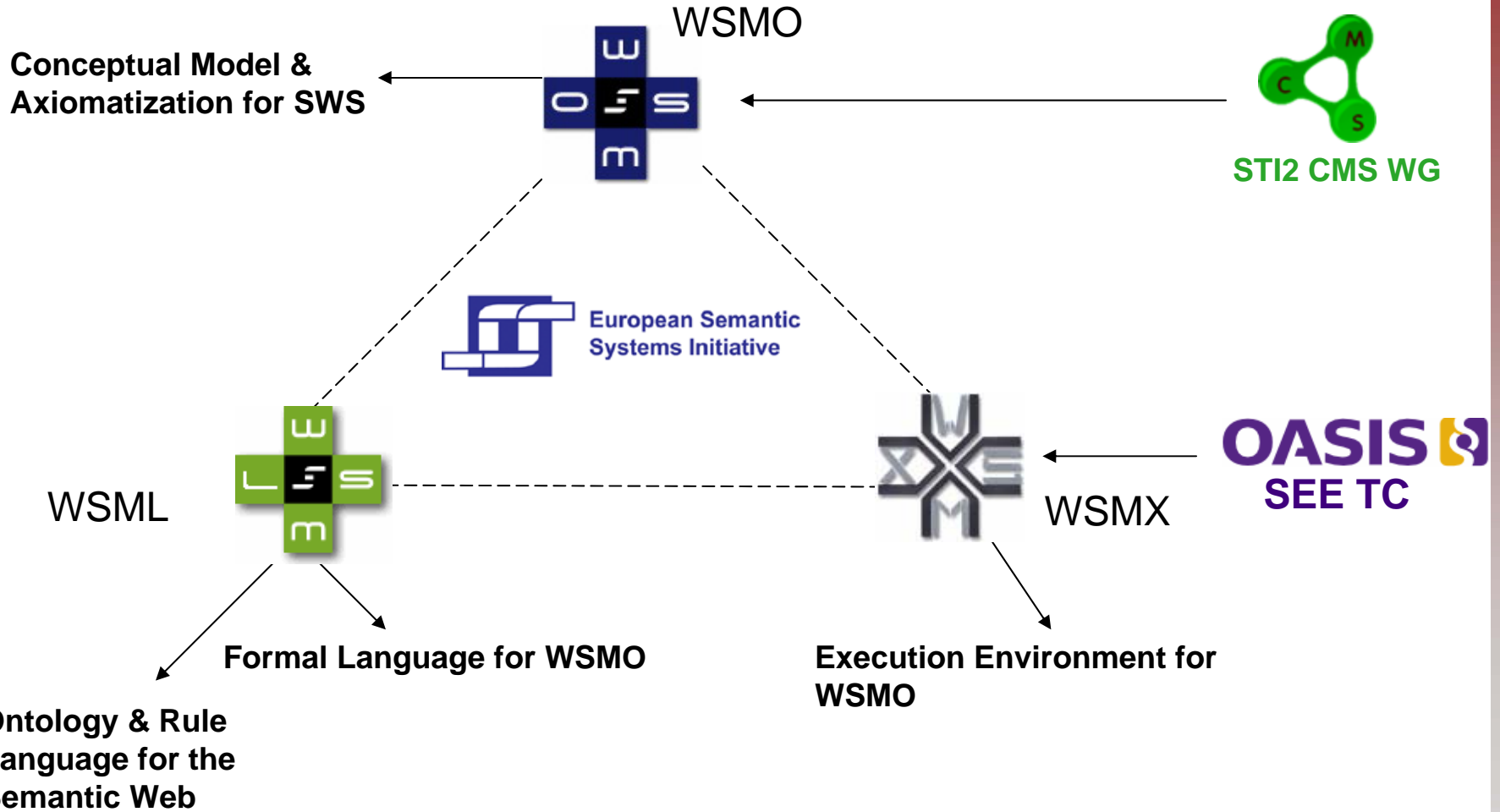
WSMF

- WSMF [consists of four different main elements for describing semantic Web Services:
 - (1) ontologies that provide the terminology used by other elements,
 - (2) goals that define the problems that should be solved by Web Services,
 - (3) Web Services descriptions that define various aspects of a Web Service, and
 - (4) mediators which bypass interpretability problems.

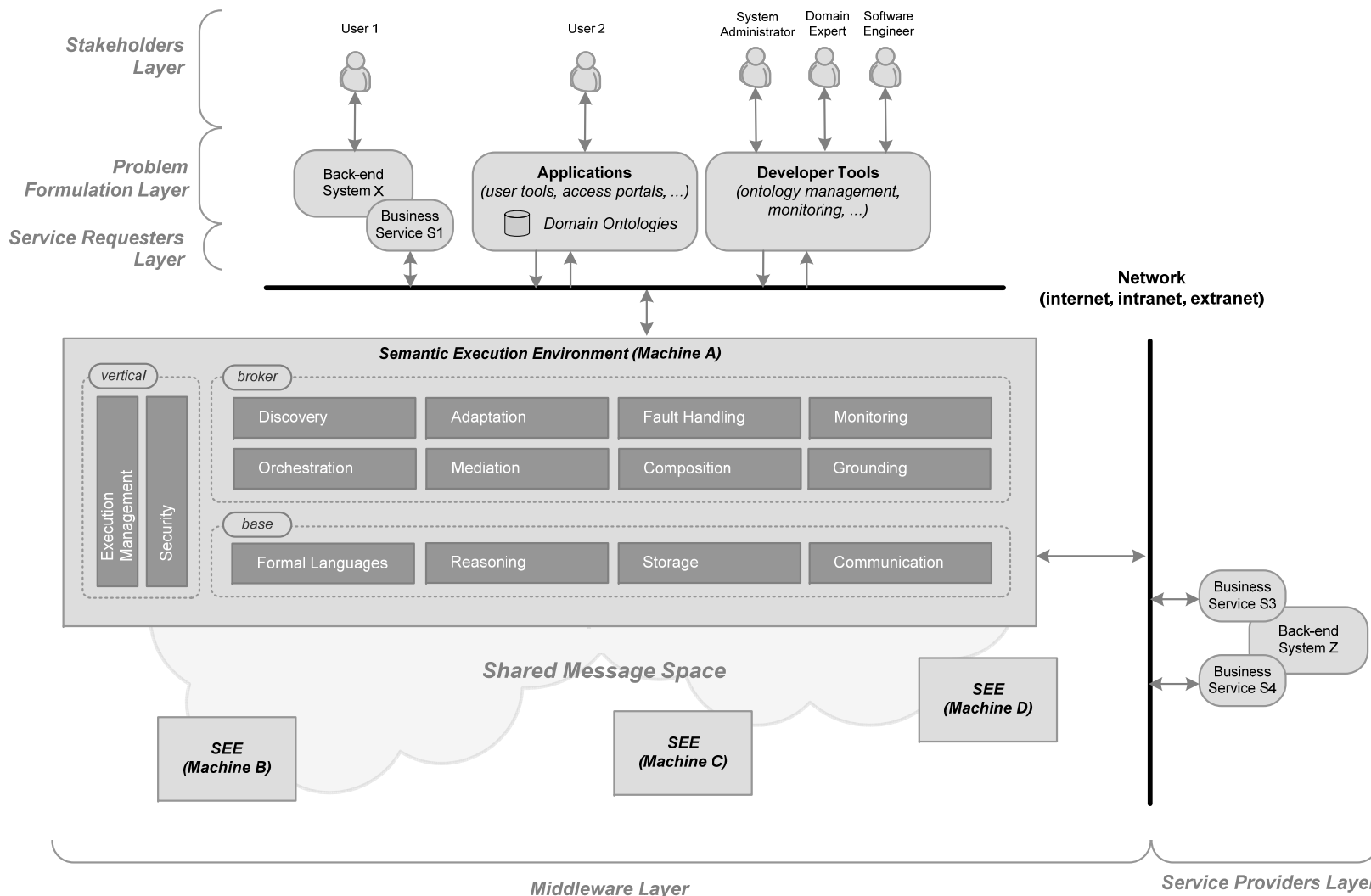
WSMO Web Service Description Model



WSMO Working Groups



Semantically-Enabled Service-oriented Architecture



SAWSDL - Semantic Annotations for WSDL and XML Schema

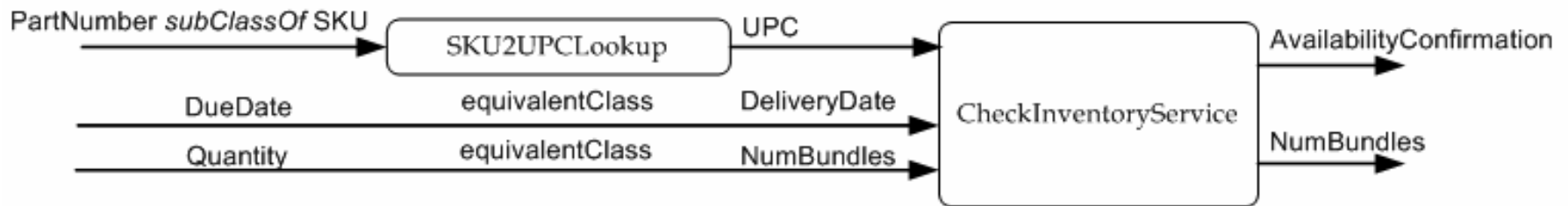
- **W3C Working Draft 10 April 2007**
- **This specification defines a set of extension attributes for the Web Services Description Language and XML Schema definition language that allows description of additional semantics of WSDL components. The specification defines how such semantic annotation is accomplished using references to semantic models, e.g. ontologies**
- **3 constructs: modelReference, liftingSchemaMapping, loweringSchemaMapping**

A Web Service Composition Scenario with Ontology Reasoning

A Web service request



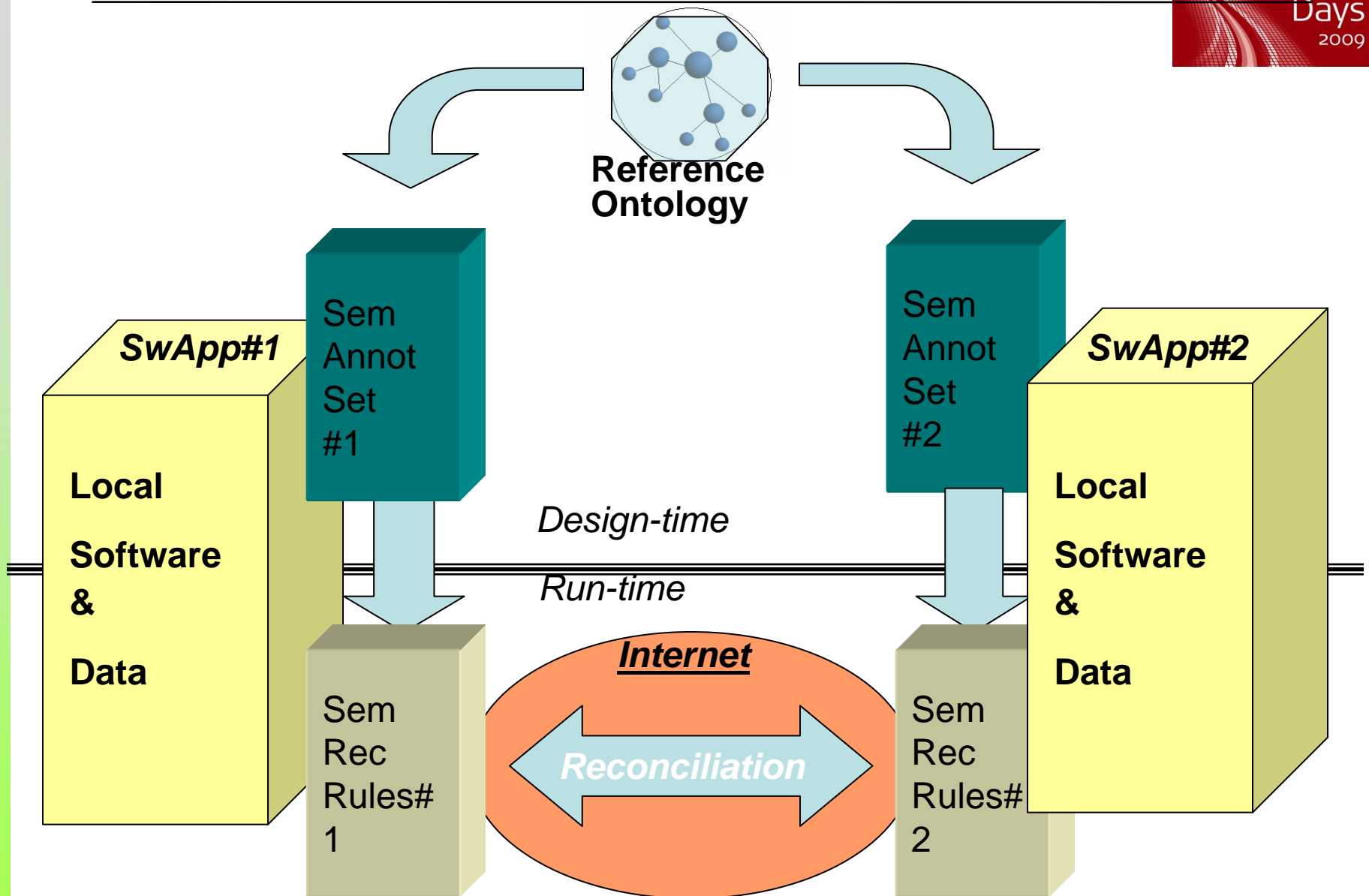
A Possible Web service composition that fulfills the above request



Model Driven Interoperability - for services

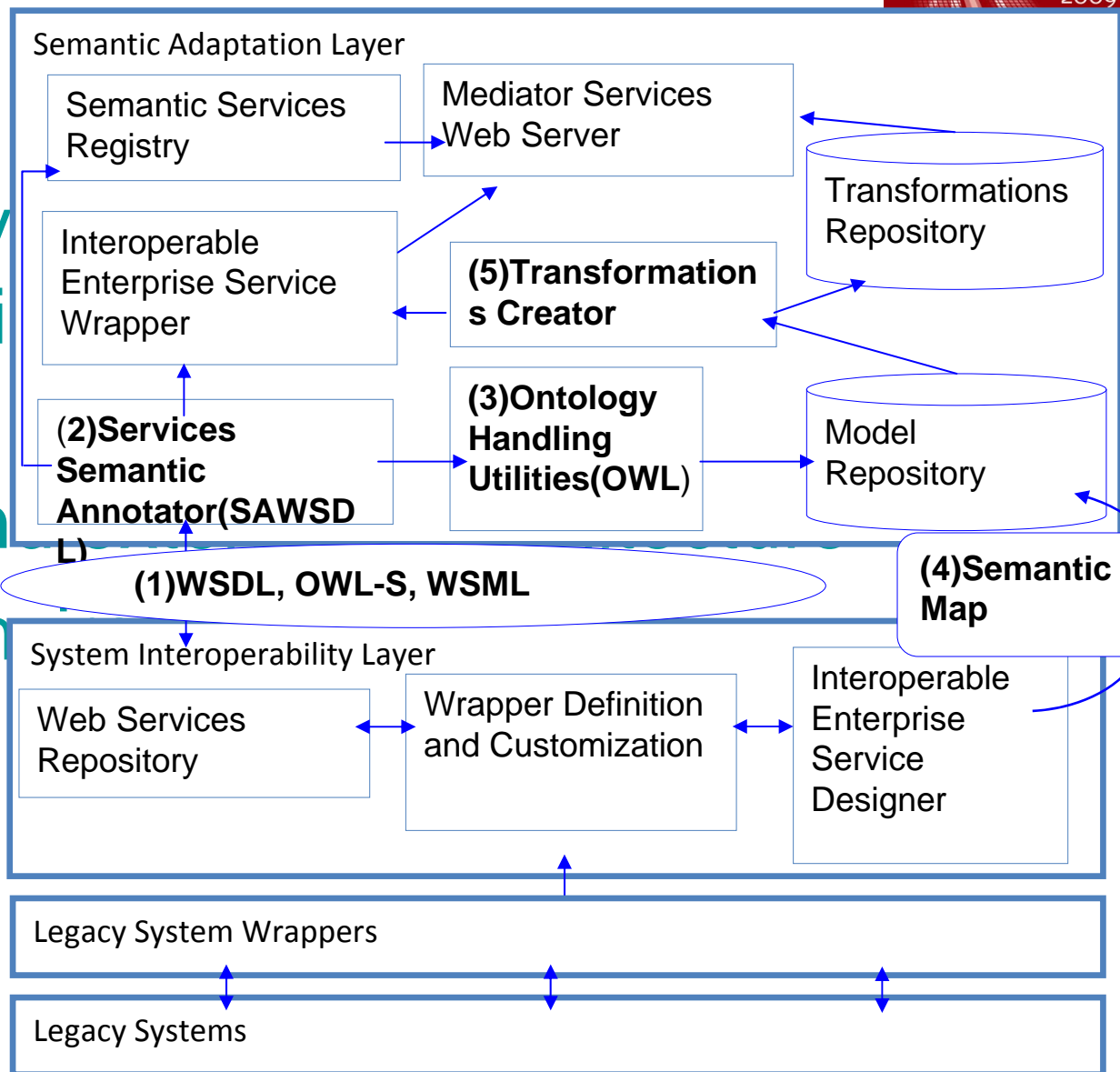
Semantic Days 2009, May 18th-20th, Stavanger, Norway

Architecture for semantic annotation and reconciliation



EMPOWER

- an innovative interoperability systems
- a flexible and
- a system en

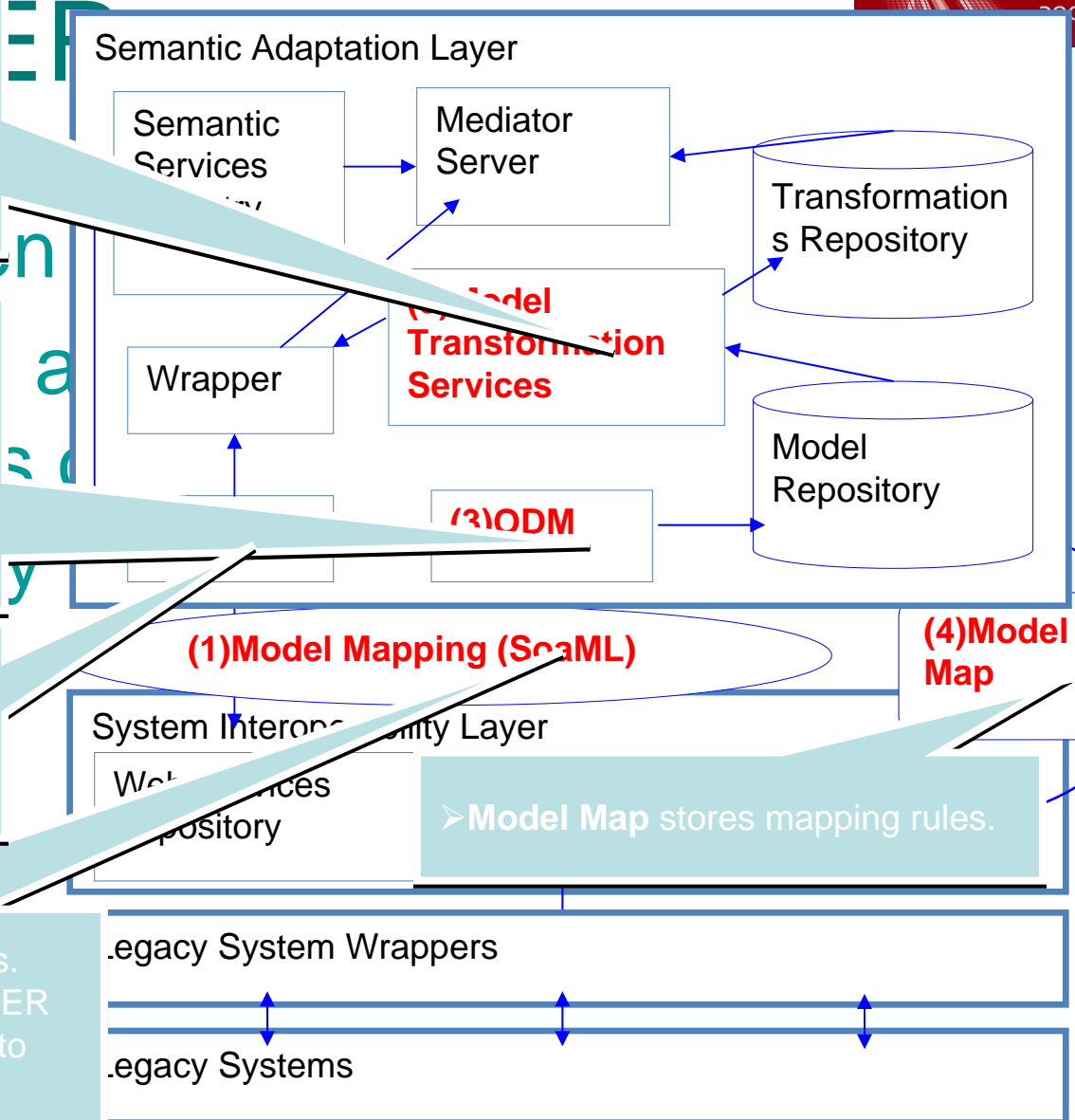


➤ **Model Transformation Services** support the runtime lifting and lowering transformations among messages and ontologies based on the Model Map.

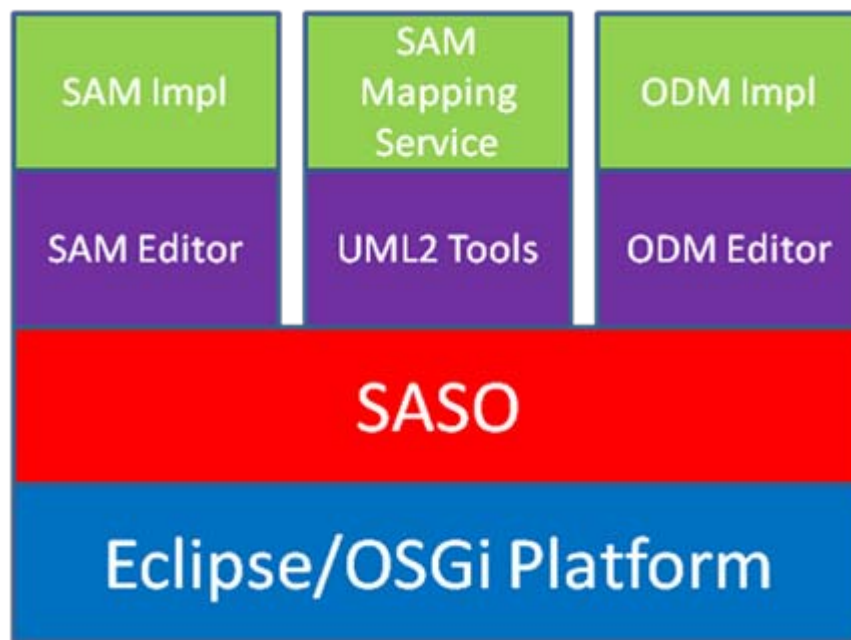
Ontology Definition Meta-model is a family of MOF meta-models, mappings between those meta-models, and a set of profiles that enable ontology modeling through the use of UML-based tools.

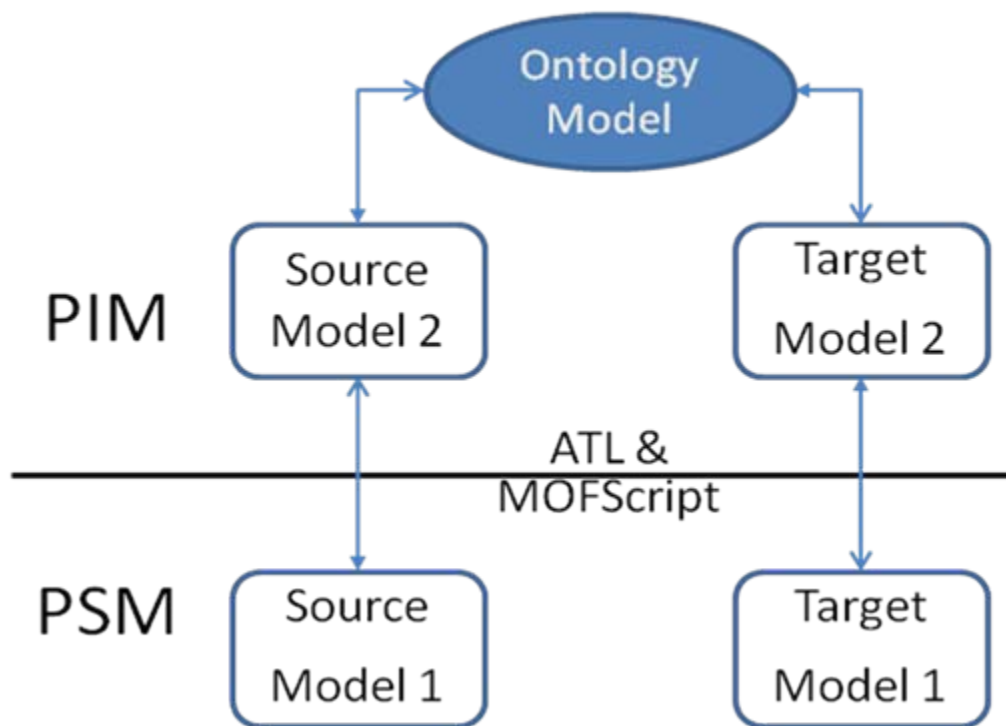
Semantic Annotation Model editor is used to relate different PIM models and ontology. It is used to annotate the SoaML model with Ontology.

SoaML describes the services models. The **Model Mapping** in the MEMPOWER includes transformations from models to ontology and ontology to models.

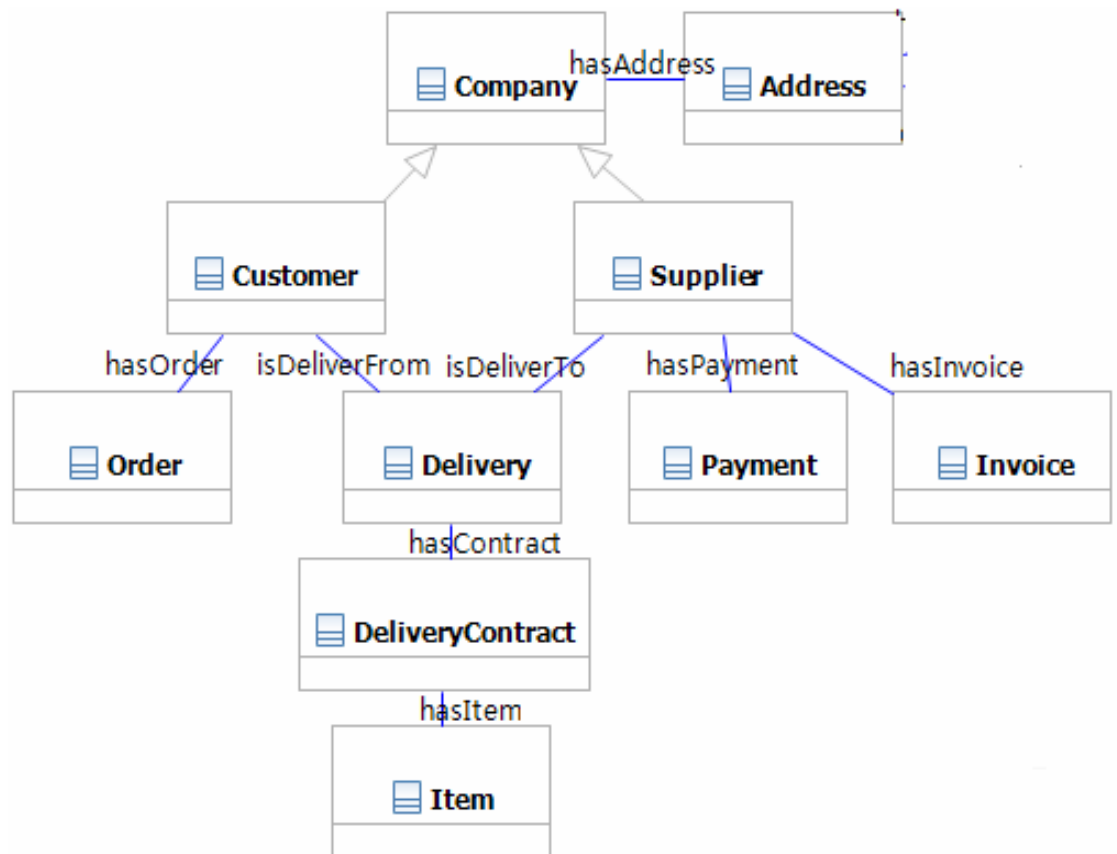


Overall SAS architecture

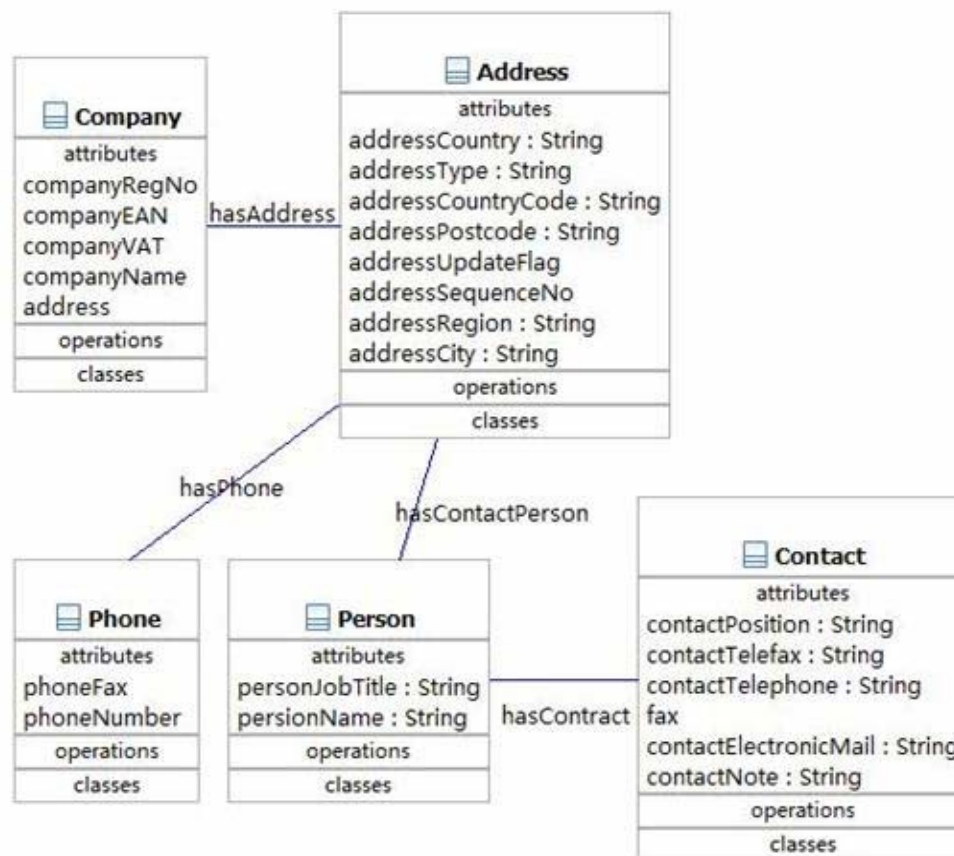




Ontology example



Address in the Ontology



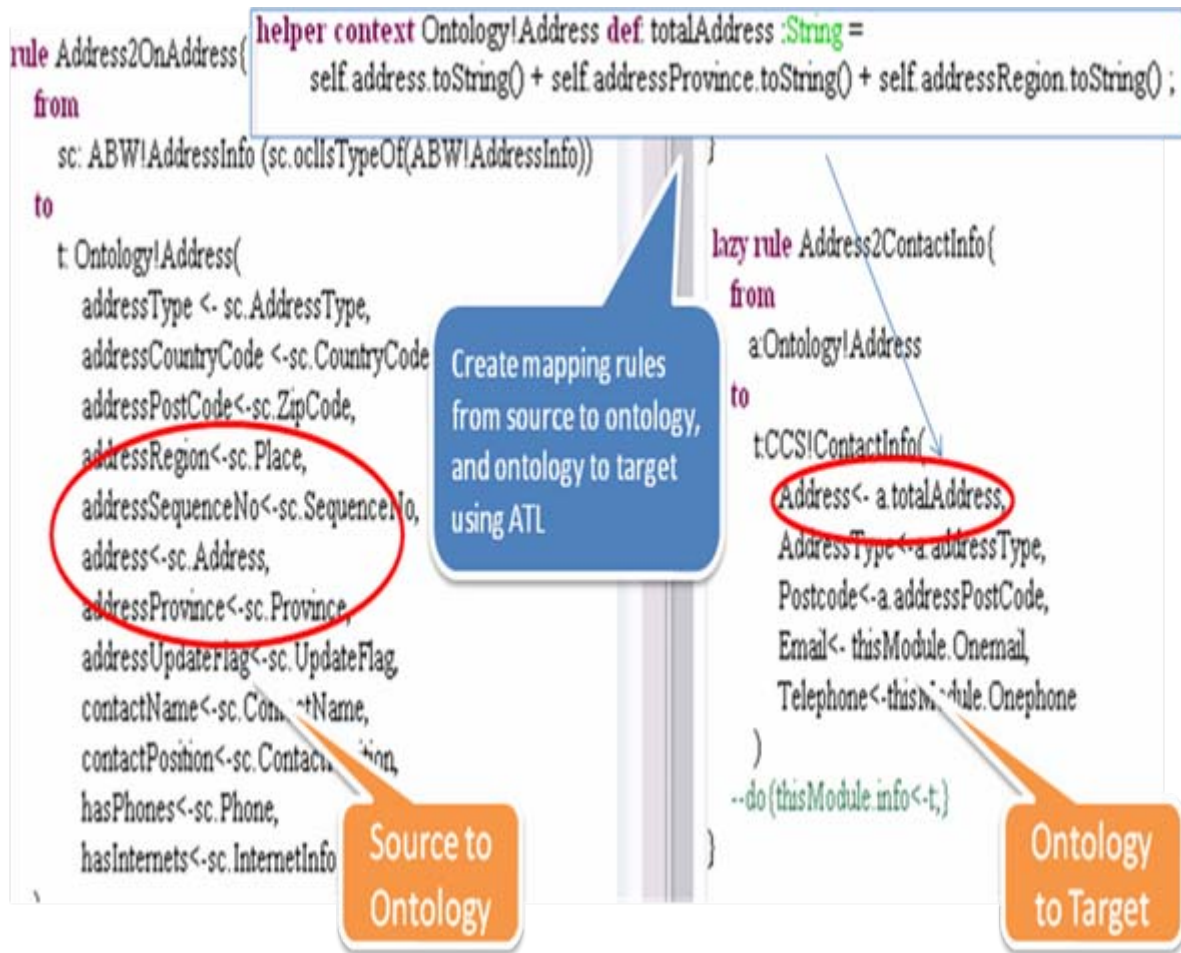
“Address” in the source schema and UML

The image shows two views of the 'Address' element in an IDE:

- UML Class Diagram (Left):** Shows the class `(AddressInfoType)` with attributes:
 - `UpdateFlag` (string)
 - `AddressType` (string)
 - `ContactName` ([0..1] string)
 - `ContactPosition` ([0..1] string)
 - `Address`** (string)
 - `Place` ([0..1] string)
 - `Province` ([0..1] string)
 - `ZipCode` ([0..1] string)
 - `CountryCode` (string)
 - `SequenceNo` ([0..1] int)
 - `InternetInfo` ([0..1] (InternetInfoType))
 - `Phone` ([0..1] (PhoneType))
- XSD Schema (Right):** Shows the `AddressInfo` element with children:
 - `GenModel`
 - `ExtendedMetaData`
 - `UpdateFlag` (String)
 - `AddressType` (String)
 - `ContactName` (String)
 - `ContactPosition` (String)
 - `Address`** (String)
 - `Place` (String)
 - `Province` (String)
 - `ZipCode` (String)
 - `CountryCode` (String)
 - `SequenceNo` (Int)
 - `InternetInfo` (InternetInfo)
 - `Phone` (Phone)

Two blue callout bubbles highlight the 'Address' element in both views, with text indicating its correspondence between the UML and XSD.

“Address” in the source and target transformation rules



Further details during Semantic Days

- Tue: 1100-1130 **Semantic interoperability for public administrations in Europe – Challenges and solutions (EIF)**
- Tue: 1530-1600 **Methods and tools for semi-automatic ontology engineering**
- Tue: 1600-1630 **QuOnto: Ontology-based data access and integration using relational technology**
- Tue: 1700-1730 **Semantic annotation for web services and their relevance to environmental models (ENVISION, SWING)**
- Tue 1730-1800 **Systems interoperability through use of semantic technologies (COIN)**

- Wed: 0830-0900 **Information management for interoperability in European air traffic control (SESAR)**
- Wed: 1030-1100 **Supporting intelligent and automated integrated operations with agent technologies in a services architecture (SHAPE)**
- Wed: 1130-1200 **IT architecture for supporting semantic interoperability through use of semantic annotations (EMPOWER)**
- Wed: 1330-1400 **Best practices in collaborative ontology engineering**
- Wed: 1430-1500 **Model-driven integration architecture for IO G2**
- Wed: 1500-1530 **Using semantic technology to represent standards for operations & maintenance**