Semantic Web Services for Business Processes Management

ICIW 2007, May 13, Mauritius

Semantics Utilised for Process Management
within and between Enterprises

Semantic BPM Tutorial

at ICIW 2007

Le Morne, Mauritius

May 13th, 2007

Presenters

- Carlos Pedrinaci (KMI)
- Sebastian Stein (IDS Scheer)
- Michael Stollberg (DERI Austria)

Acknowledgement & Copyrights

- **This material is based upon works supported by the EU under the SUPER project (FP6 - 026850)**

- **Material Preparation**
  - KMI: John Domingue, Carlos Pedrinaci, Barry Norton
  - Poznan University: Agata Filipowska
  - IAAS, University of Stuttgart: Dimka Karastoyanova, Jörg Nitzsche, Tammo van Lessen, Zhilei Ma, Frank Leymann
  - IDS Scheer: Sebastian Stein
  - DERI Austria: Dumitru Roman, Michael Stollberg
  - DERI Ireland: Maciej Zaremba

Agenda

1. Introduction: The Need of Semantics in BPM

2. Business Process Management
   - Introduction
   - BPEL

3. Semantic Web Services
   - Introduction
   - SWS Technologies

4. Integration: The SUPER Approach

Introduction

Querying the Process Space

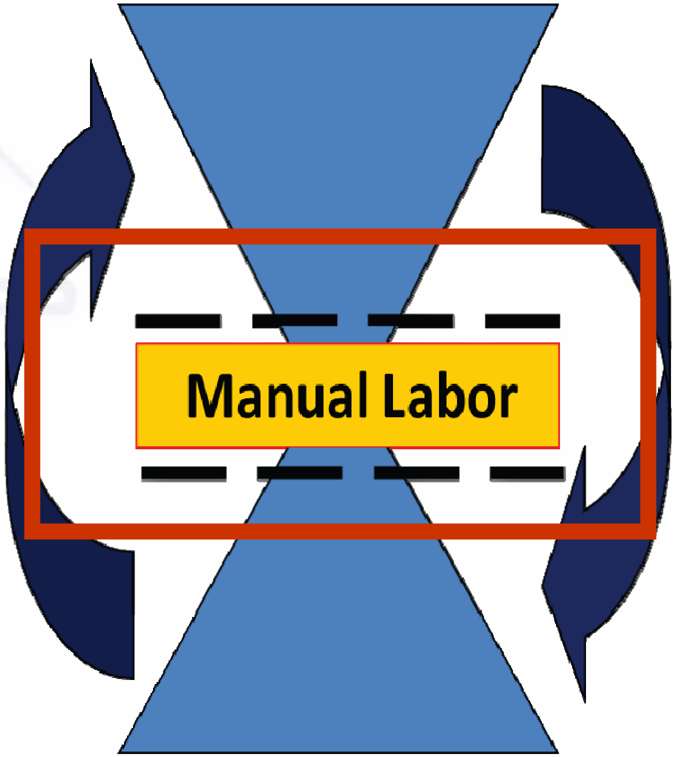
"In which of our food manufacturing machines are we processing meat or raw eggs?"

"Do we have a cost approval process for items below $ 200?"

"How many inventory management methods are currently in use?"

The Critical IT / Process Divide

Business Experts' Perspective: Processes

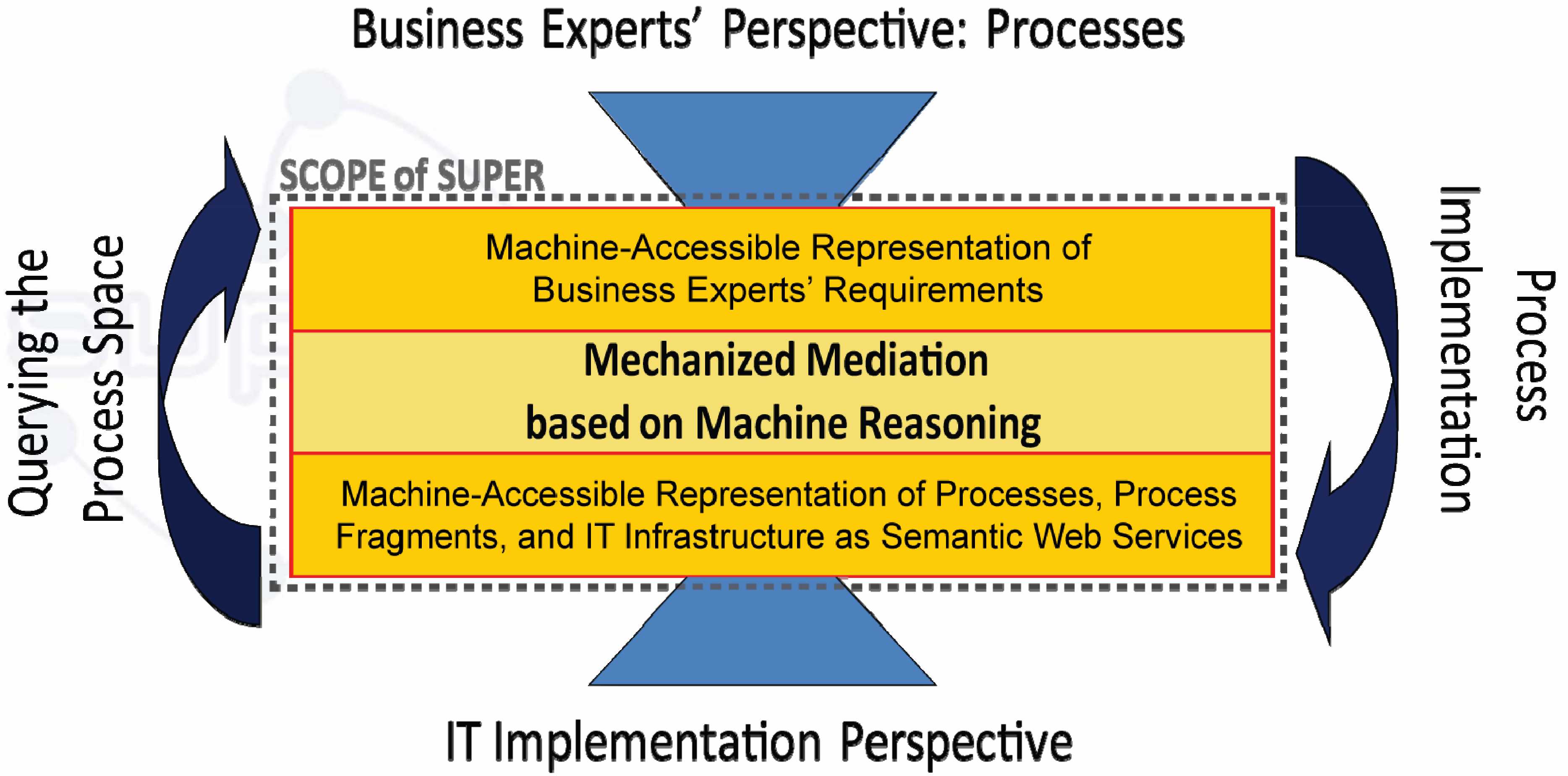


Querying the Process Space

Process Implementation

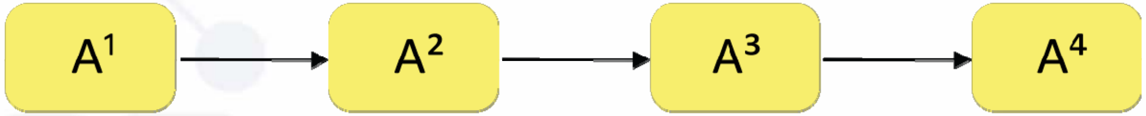
IT Implementation Perspective

The Critical IT / Process Divide


Business Experts' Perspective: Processes
SCOPE of SUPER
Machine-Accessible Representation of Business Experts' Requirements
Mechanized Mediation based on Machine Reasoning
Machine-Accessible Representation of Processes, Process Fragments, and IT Infrastructure as Semantic Web Services
IT Implementation Perspective
Querying the Process Space
Process Implementation

What Are My Services?

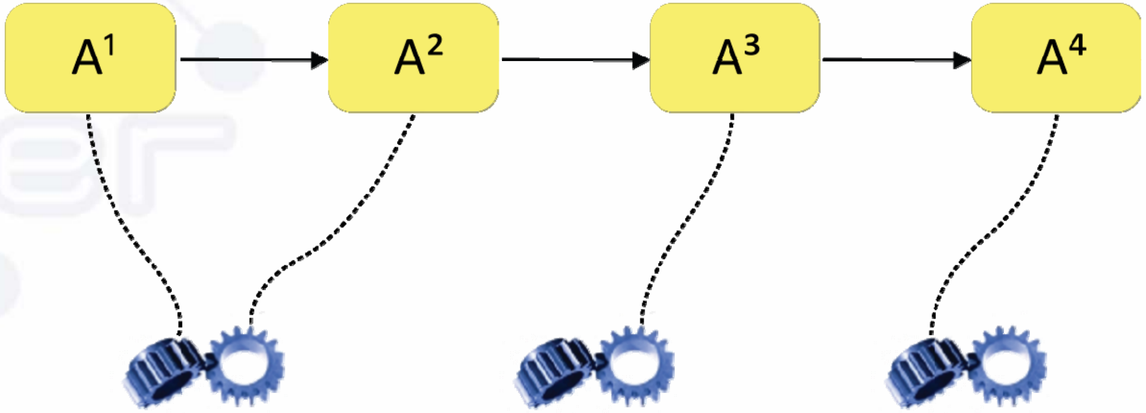
Here is my business process!
I think this solves my business problem nicely…



Nice try, but it won't run…
You need to specify the services that perform each step!

What Are My Services?

I don’t understand about these technical details!
This is my view on the process…



o.k. no problem, I will help you…

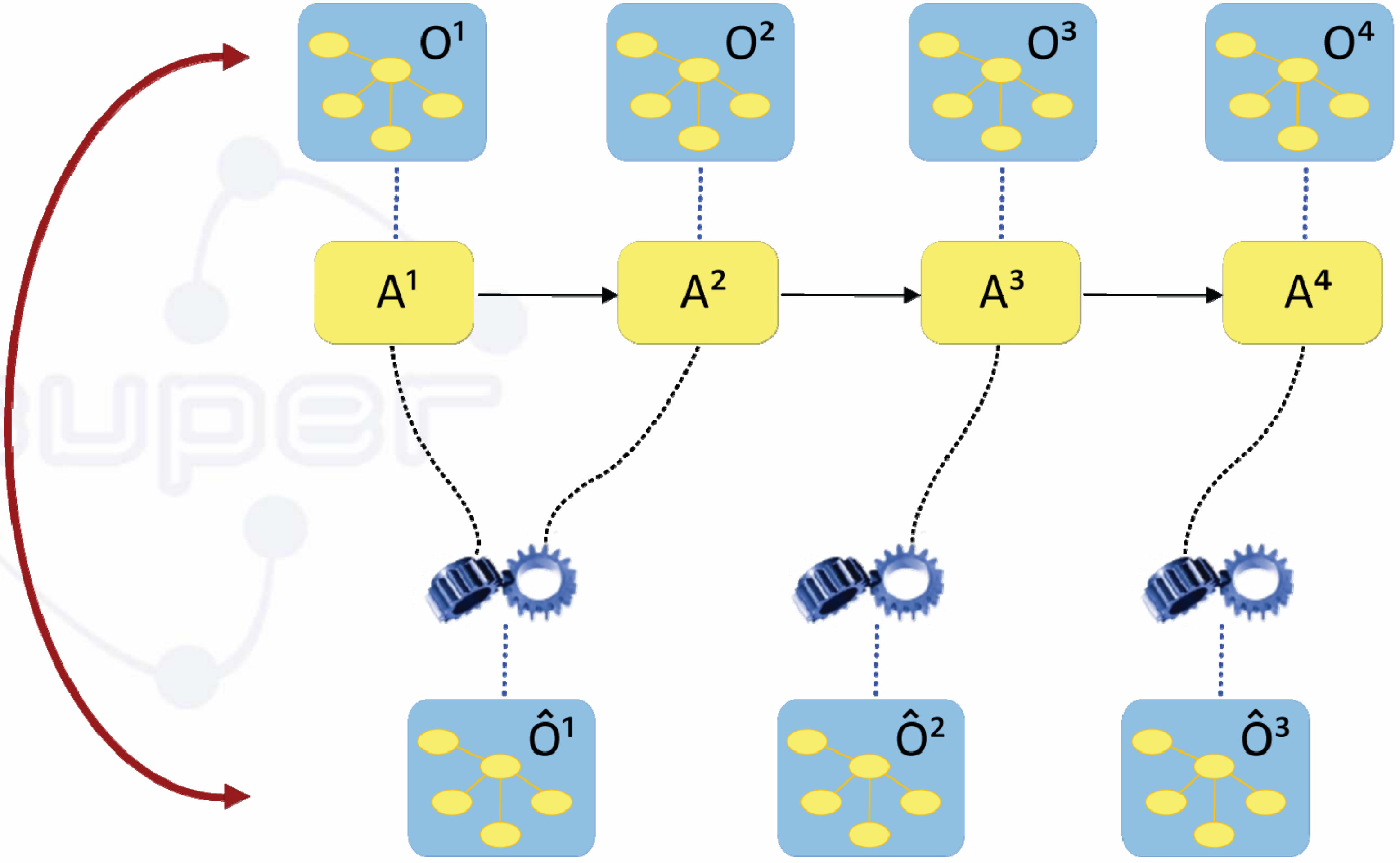
What are my services?

This is cumbersome!
Why do I always need IT staff
to solve my business problems?

It takes too long to get these folks,
they use different terminology than I do…

I am happy to describe what
the activities do in my terms.
Can the system be smart enough
and find the right services itself???

Matching Activities and Port Types Based on Semantics



Semantic Web Services

Supporting Business Users Better

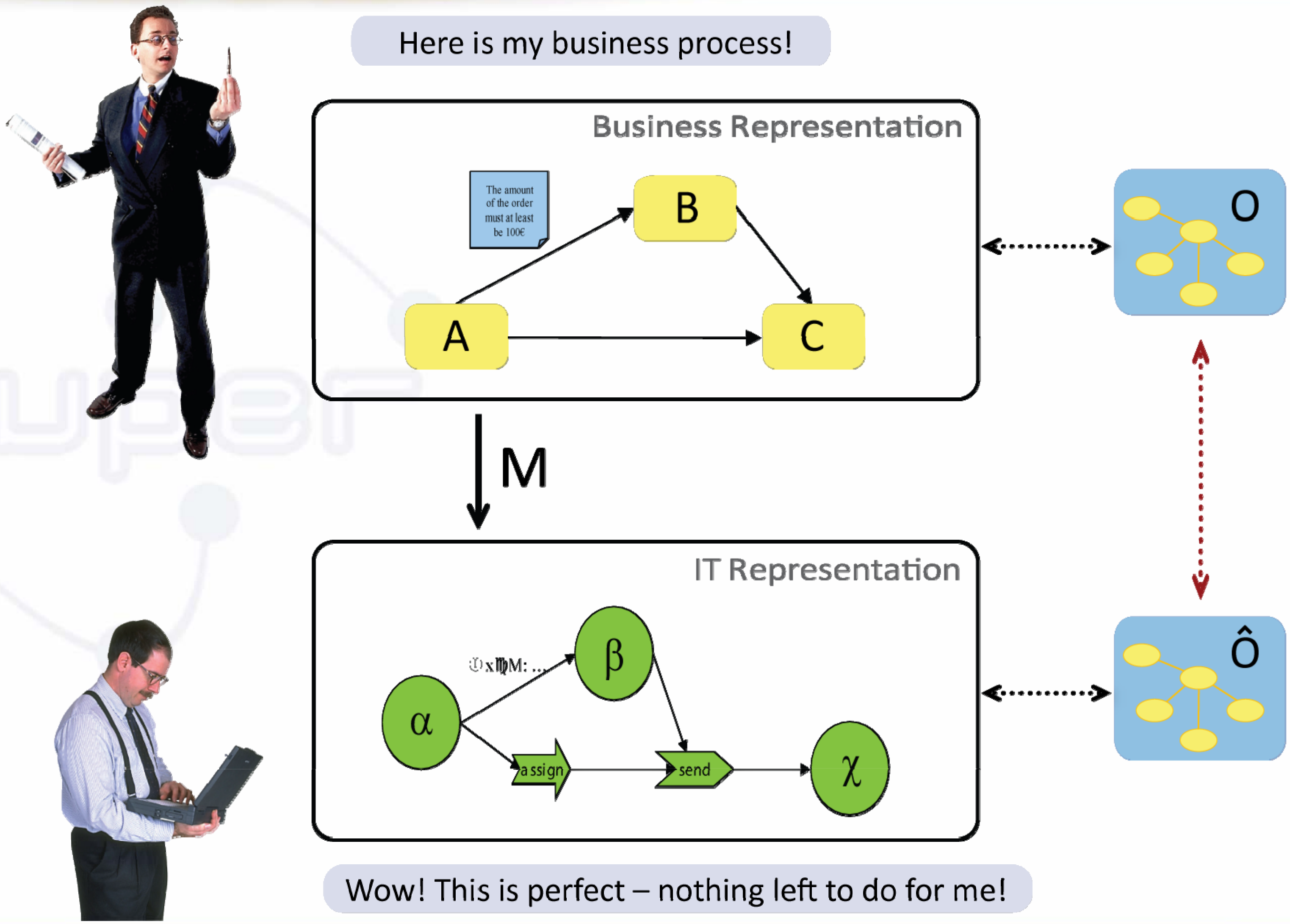
Why do I have to draw everything?

Why do I have to use “expressions” and that technical stuff at all?

Why isn’t my description sufficient?

Matching Model Representations & Semantics

Here is my business process!



Wow! This is perfect – nothing left to do for me!

Semantics Utilised for Process Management within and between Enterprises

Business Process Management

Introduction

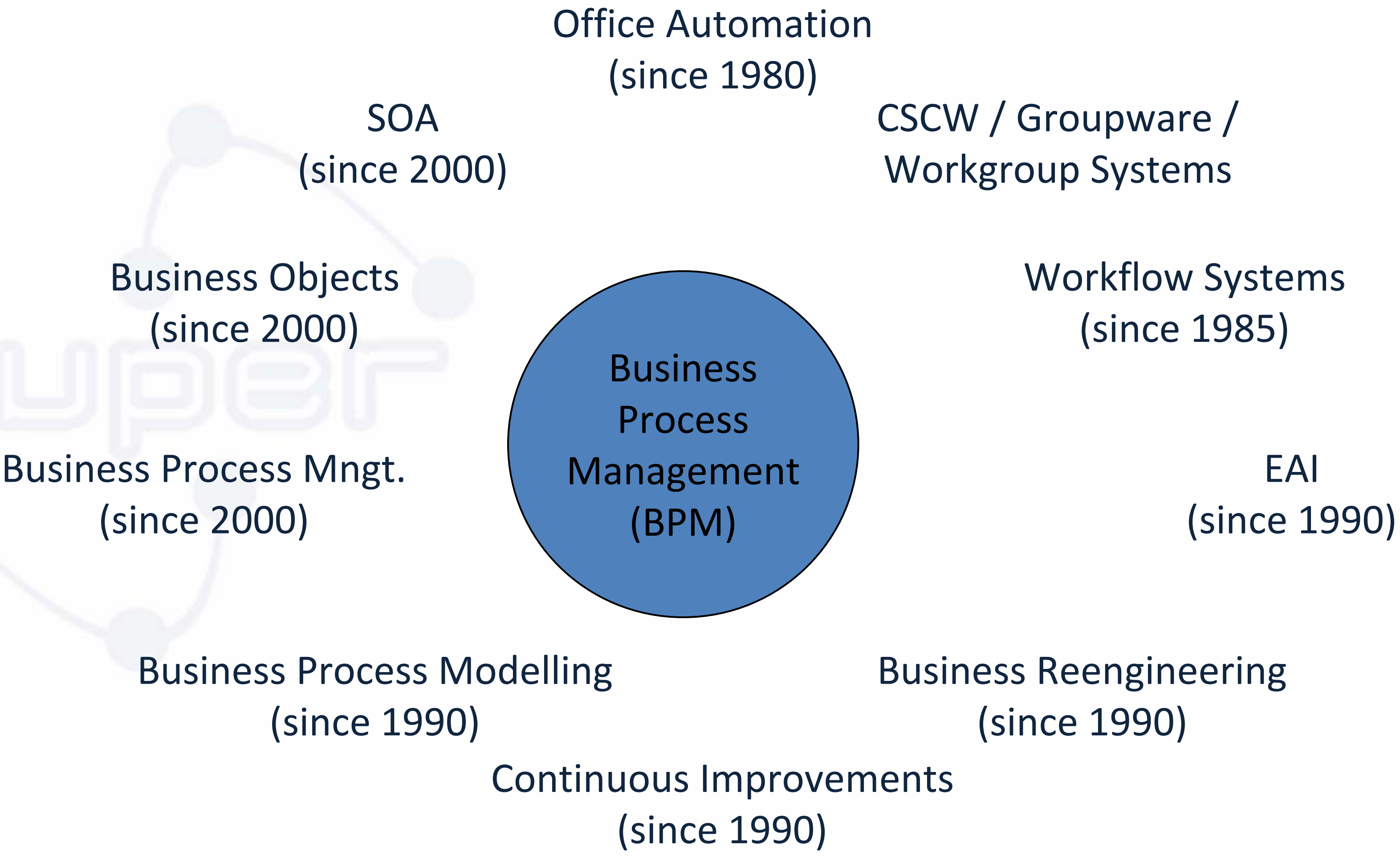
Sebastian Stein, IDS Scheer

BPM: Introduction

- BPM's Parents and Definition
- Enterprise Model
- Business Process Lifecycle
- BPM Applications
- Summary

© Sebastian Stein

BPM's Parents and Definition



Office Automation (since 1980)

SOA (since 2000)

CSCW / Groupware / Workgroup Systems

Business Objects (since 2000)

Workflow Systems (since 1985)

Business Process Management (BPM)

Business Process Mngt. (since 2000)

EAI (since 1990)

Business Process Modelling (since 1990)

Business Reengineering (since 1990)

Continuous Improvements (since 1990)

© Sebastian Stein

BPM’s Parents and Definition

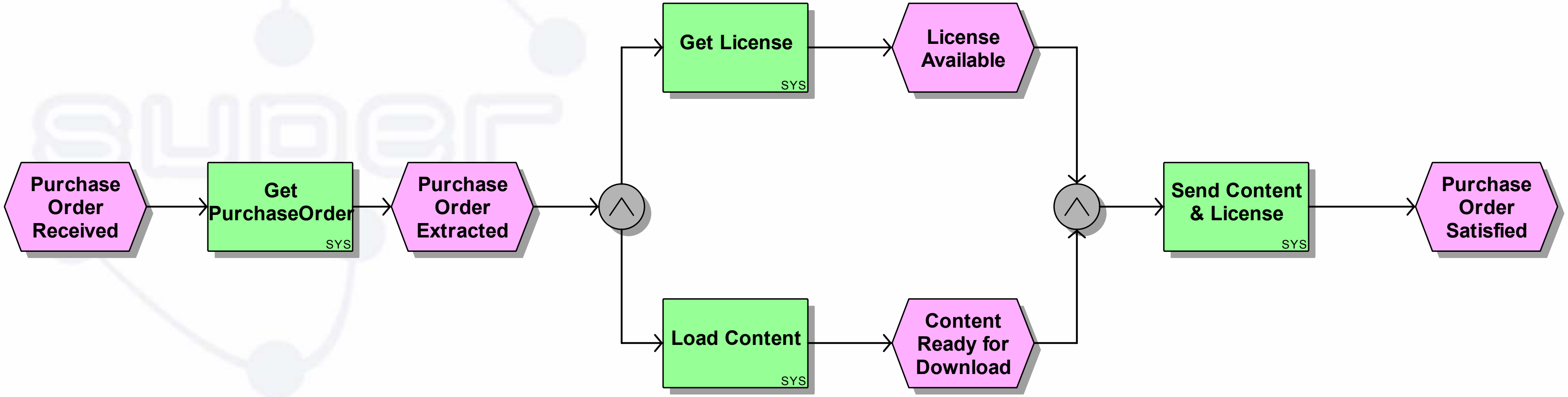
- there are several competing definitions
- own focus coins BPM definition:
  - focus on documentation
  - focus on process and execution
  - focus on IT architecture
  - focus on costs and risks
  - focus on business strategy
  - etc.
- in SUPER we have a strong process and execution focus

© Sebastian Stein

Enterprise Model

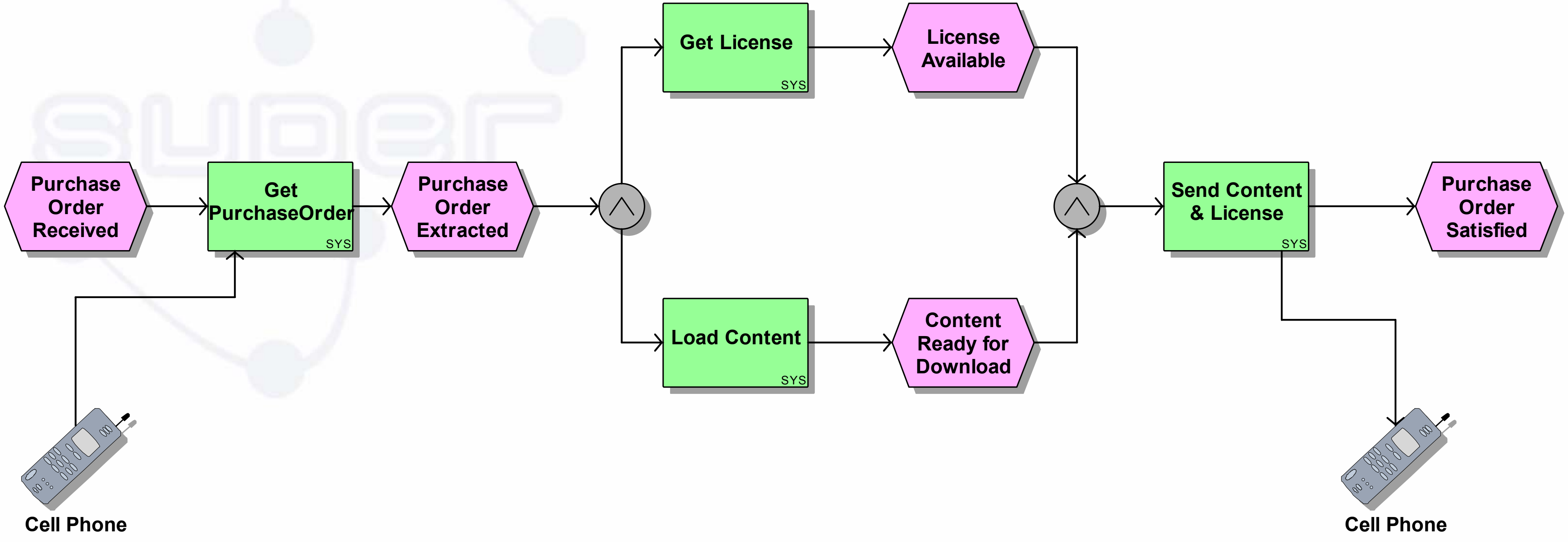
- model of an enterprise
  - internal enterprise architecture
  - internal requirements
  - interfaces
  - business processes
  - external integration
  - external requirements
  - ...
- model is an abstraction of reality
- used by many different stakeholders
  - views needed
  - abstraction levels needed
  - lifecycle concept needed
  - different languages, notations and formalisms needed

Enterprise Model



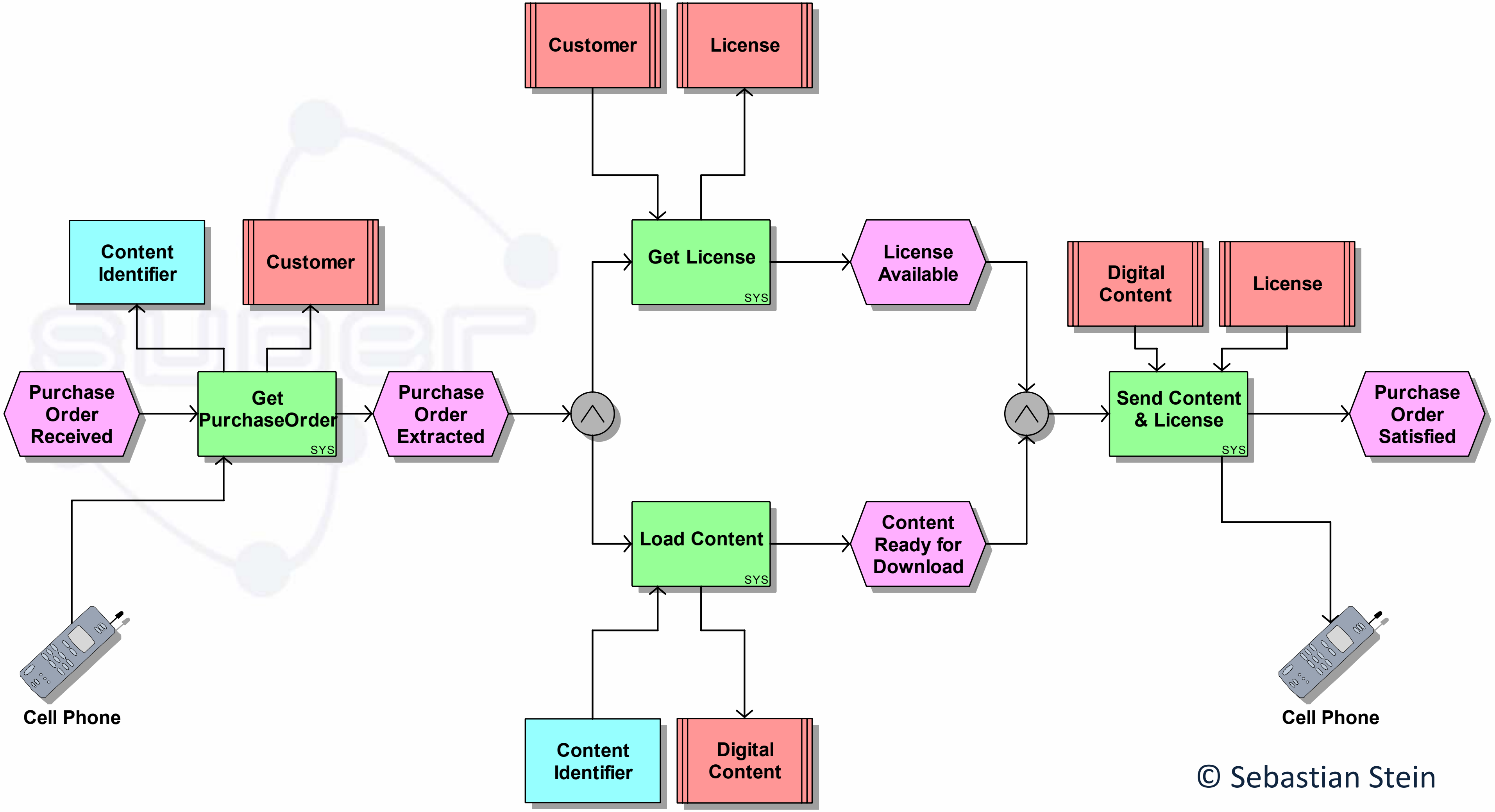
© Sebastian Stein

Enterprise Model



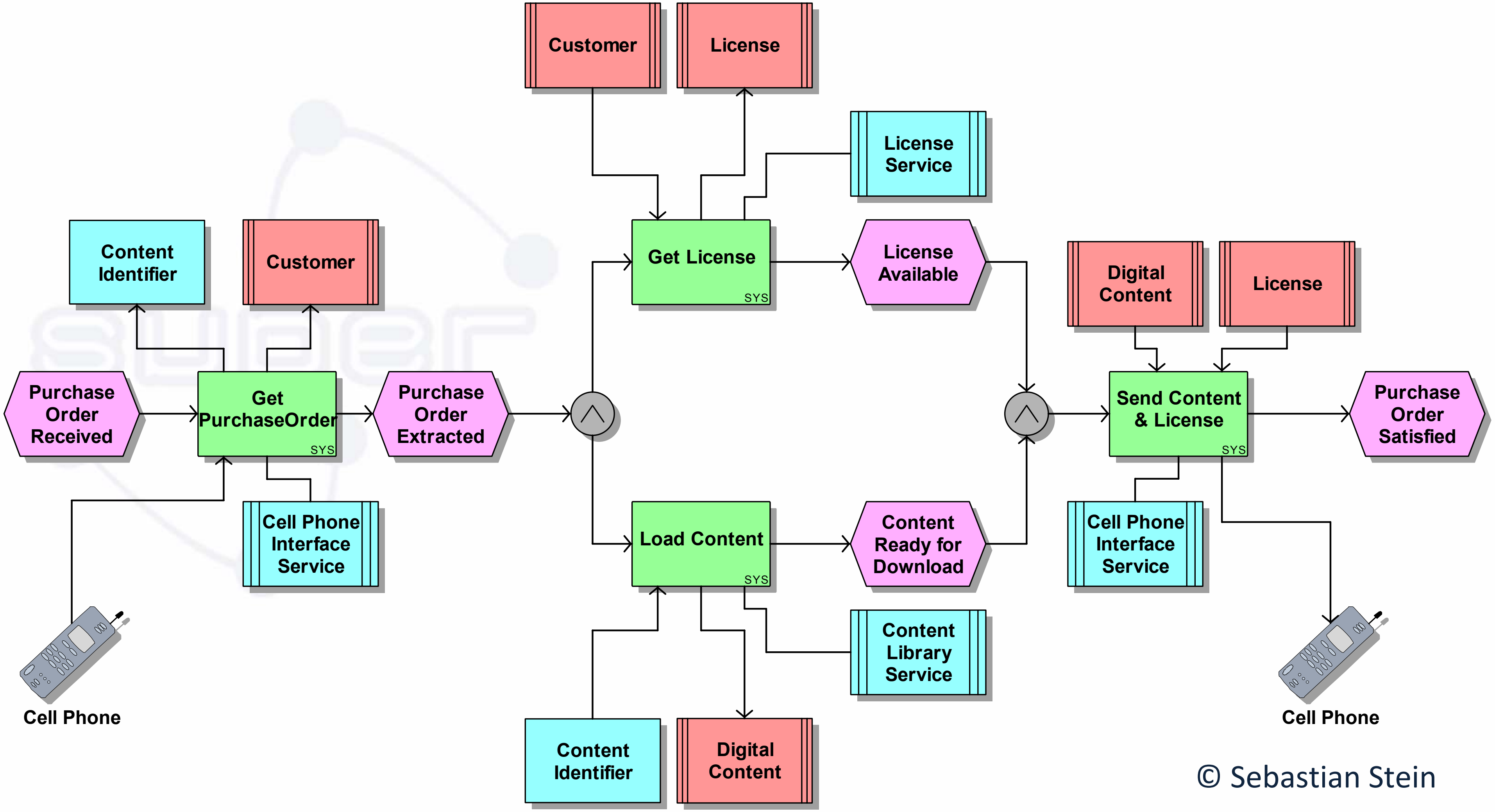
© Sebastian Stein

Enterprise Model



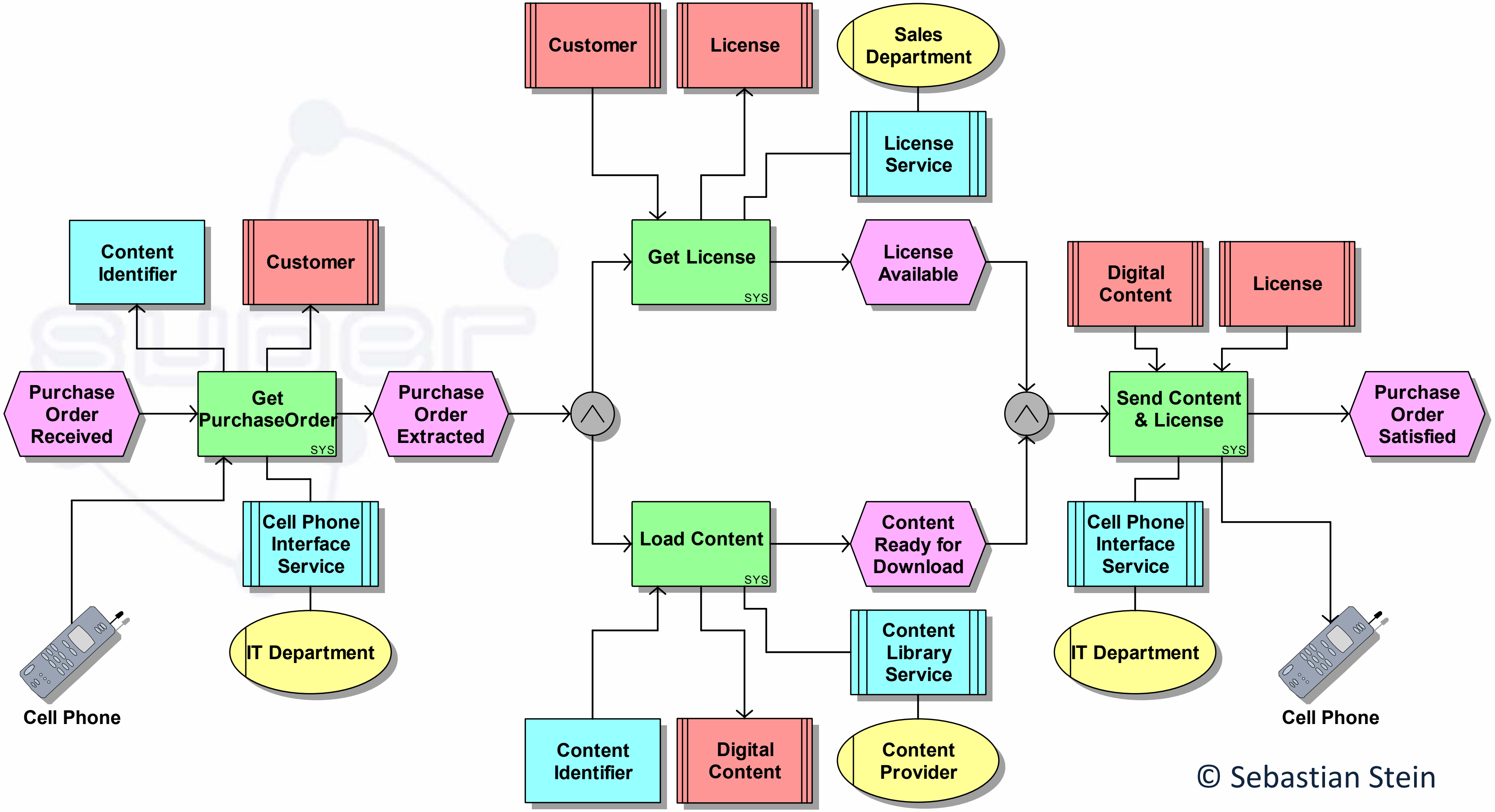
© Sebastian Stein

Enterprise Model



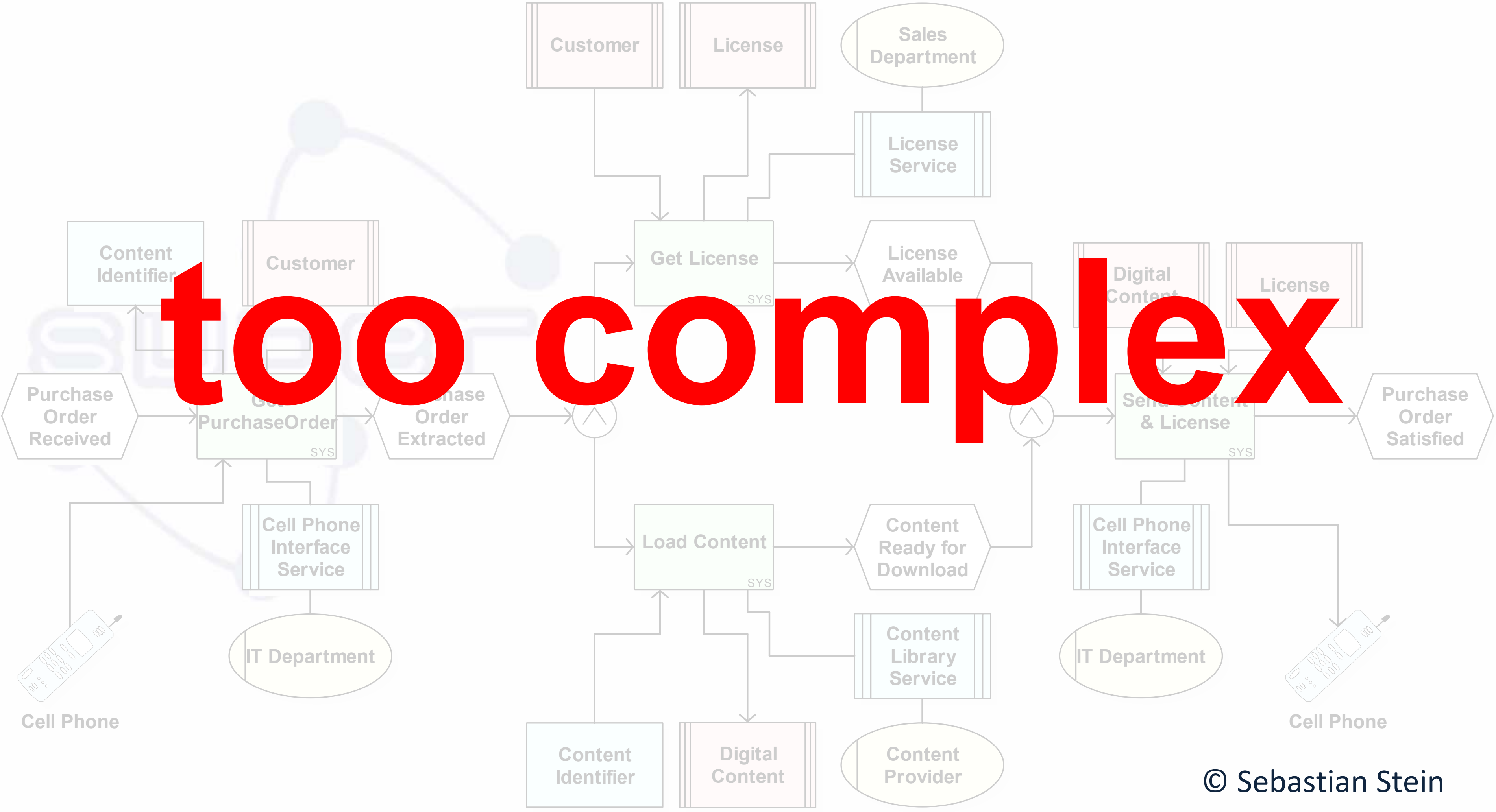
© Sebastian Stein

Enterprise Model



© Sebastian Stein

Enterprise Model



too complex

© Sebastian Stein

Enterprise Model

- possible abstraction layers are:
  - requirements definition
  - design specification
  - implementation specification
  - execution and run-time models
- possible views are:
  - organisational view
  - product view
  - data view (information architecture)
  - function and IT view
  - process view

© Sebastian Stein

Enterprise Model

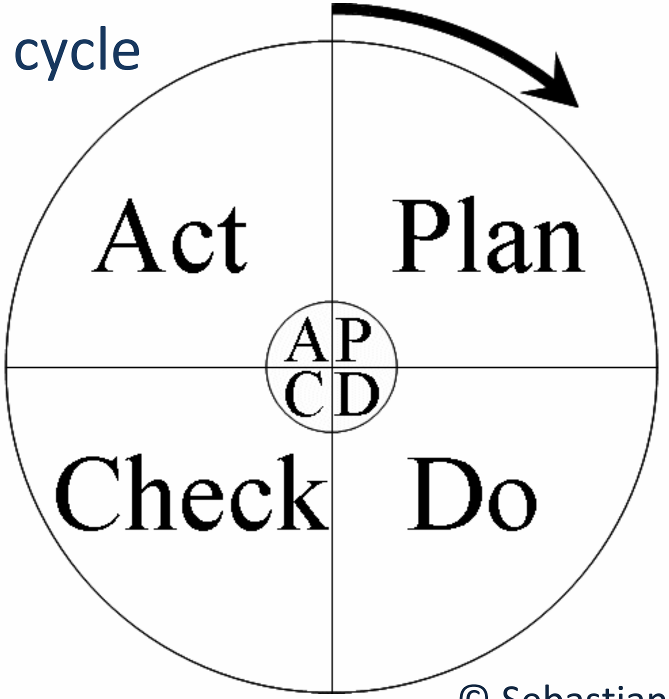
- many different frameworks for enterprise architecture, e.g.:
  - Zachman Framework (very comprehensive)
  - ArchiMate (simplified version of Zachman)
  - ARIS (promoted by IDS Scheer)
  - TOGAF (strong IT focus)
  - IAF (promoted by Cap Gemini)
- currently many discussions around process design & execution, e.g.:
  - BPMN (notation for (IT oriented) business processes)
  - EPC (notation for business processes)
  - Petrinets (formalism often used for workflow modelling)
  - UML Activity diagrams
  - XPDL (execution language for process definitions)
  - BPEL (execution language for process definitions)
  - XLANG (execution language promoted by Microsoft)
  - ...

© Sebastian Stein

Process Lifecycle

- enterprise model evolves → lifecycle
- based on general Deming cycle for continuous process improvements
- sometimes also named Shewhart cycle

1. Plan
2. Do
3. Check
4. Act



© Sebastian Stein

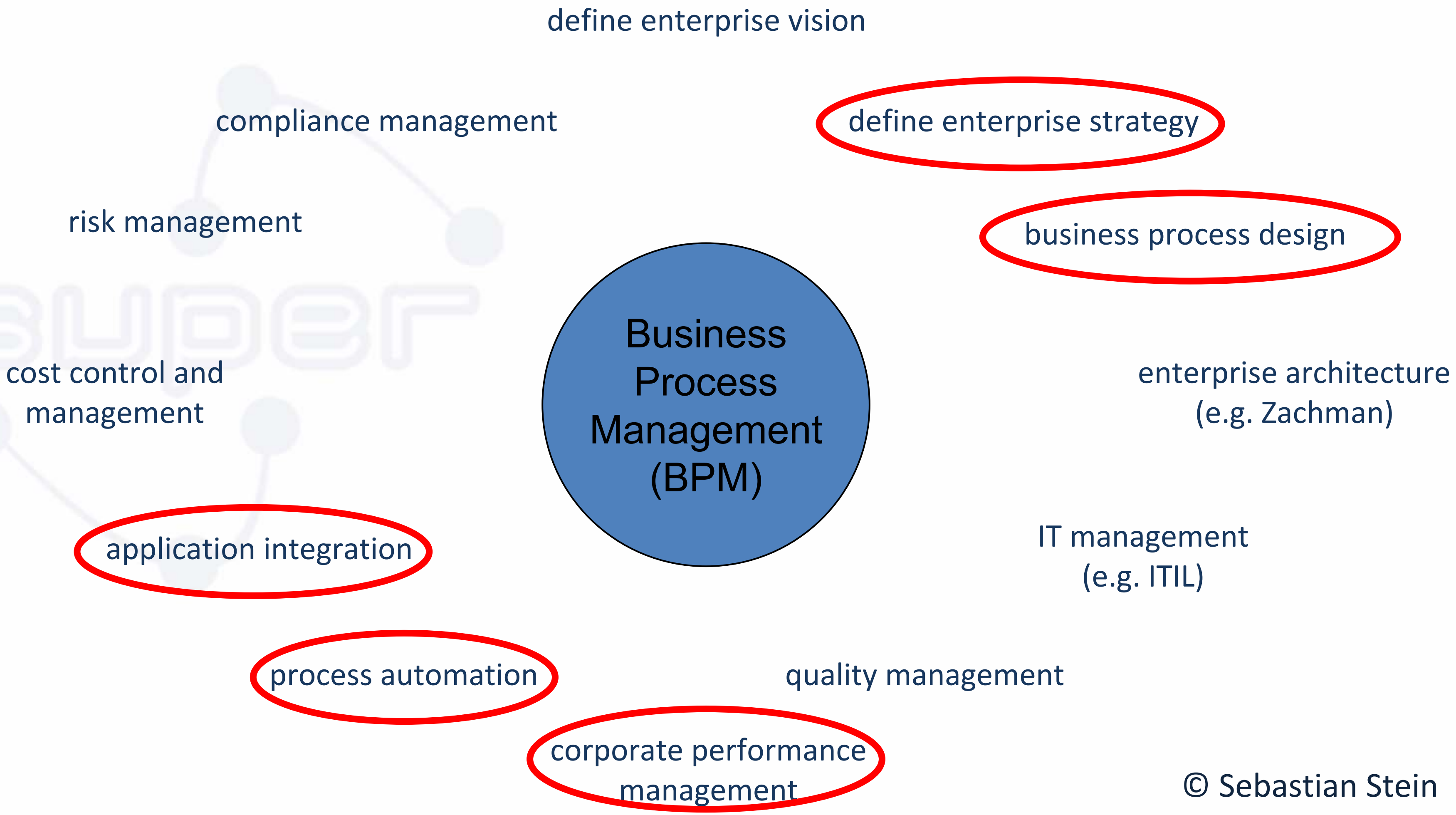
Process Lifecycle

- general business process lifecycle:

1. Analysis
   - gather requirements
   - document current state (as-is)
2. Design
   - document to-be
   - specify how to get there
3. Implementation/Execution
   - implement to-be in organisation and IT
   - change management
4. Control and Monitoring
   - monitor execution
   - measure outcome and analyse it

© Sebastian Stein

BPM Applications

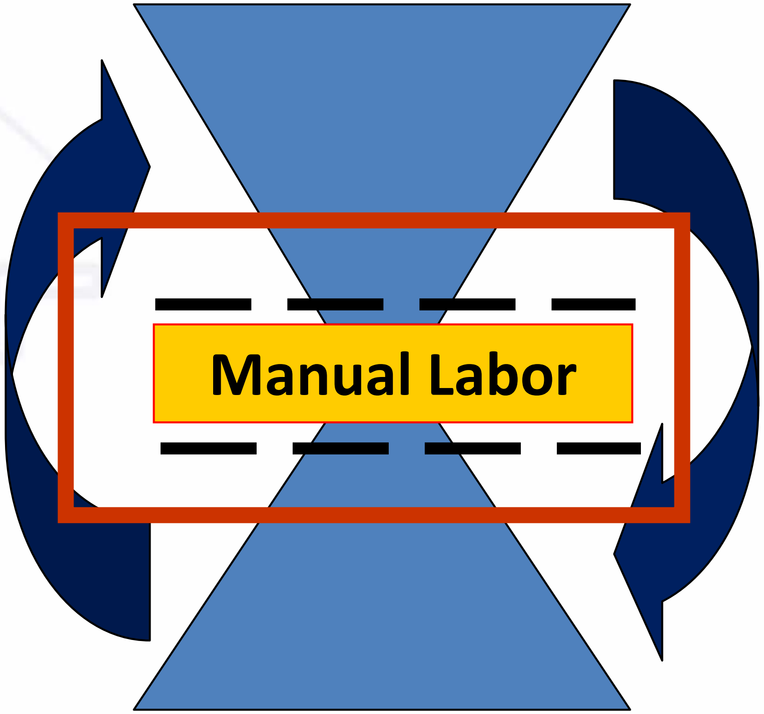

define enterprise vision
compliance management
define enterprise strategy
risk management
business process design
Business Process Management (BPM)
cost control and management
enterprise architecture (e.g. Zachman)
application integration
IT management (e.g. ITIL)
process automation
quality management
corporate performance management

© Sebastian Stein

BPM Applications

Business Experts' Perspective: Processes

Querying the Process Space



Process Implementation

IT Implementation Perspective

Summary

- BPM definition depends on your focus
- Enterprise Model describes all relevant aspects of your enterprise
- different stakeholders will have different views and information needs
- lifecycle for the different parts of the Enterprise Model
- BPM is done for many different purposes, but SUPER focus on:
  - business process design
  - business process execution
  - monitoring and analysis of execution

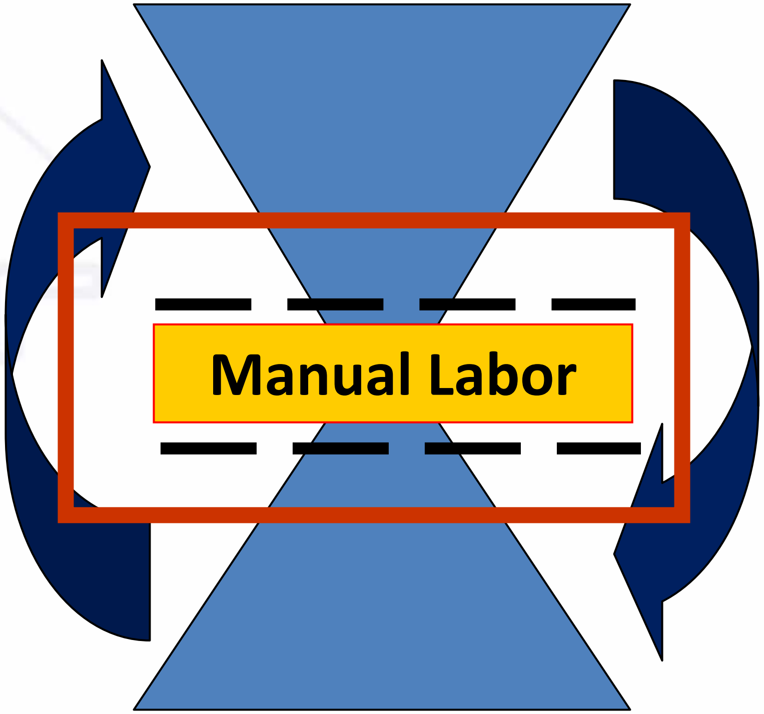
© Sebastian Stein

Business Process Management

Processes and Process Execution

BPM Applications

Business Experts’ Perspective: Processes



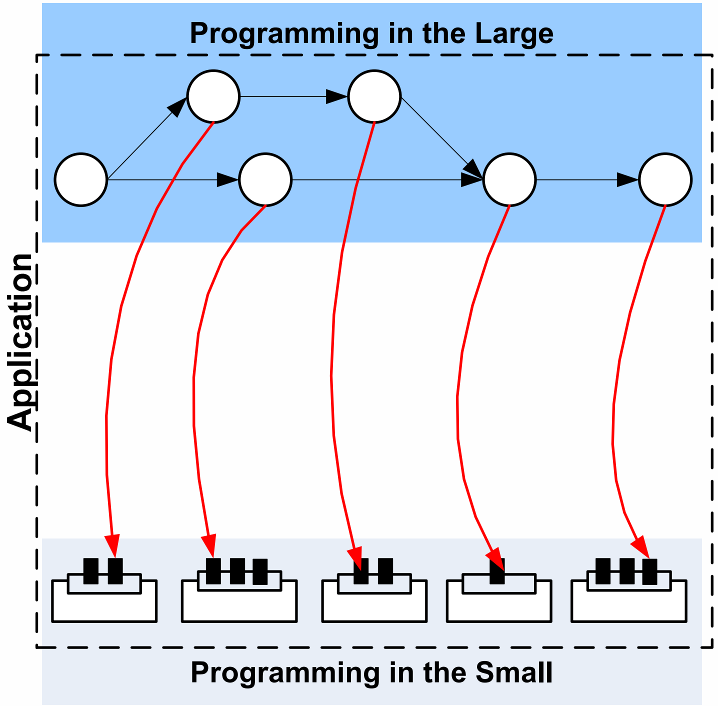
Querying the Process Space

Process Implementation

IT Implementation Perspective

Automating Business Processes

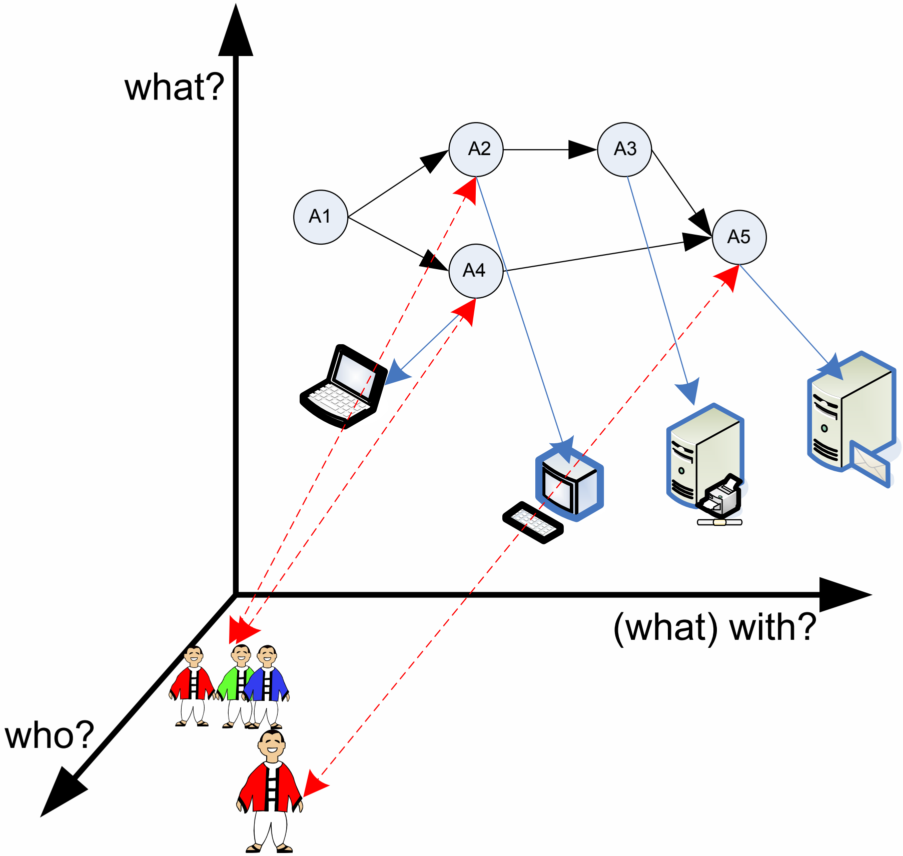
- automatic support of business processes has been enabled by Workflow Technology
- separation of control flow and business activities



© Dimka Karastoyanova

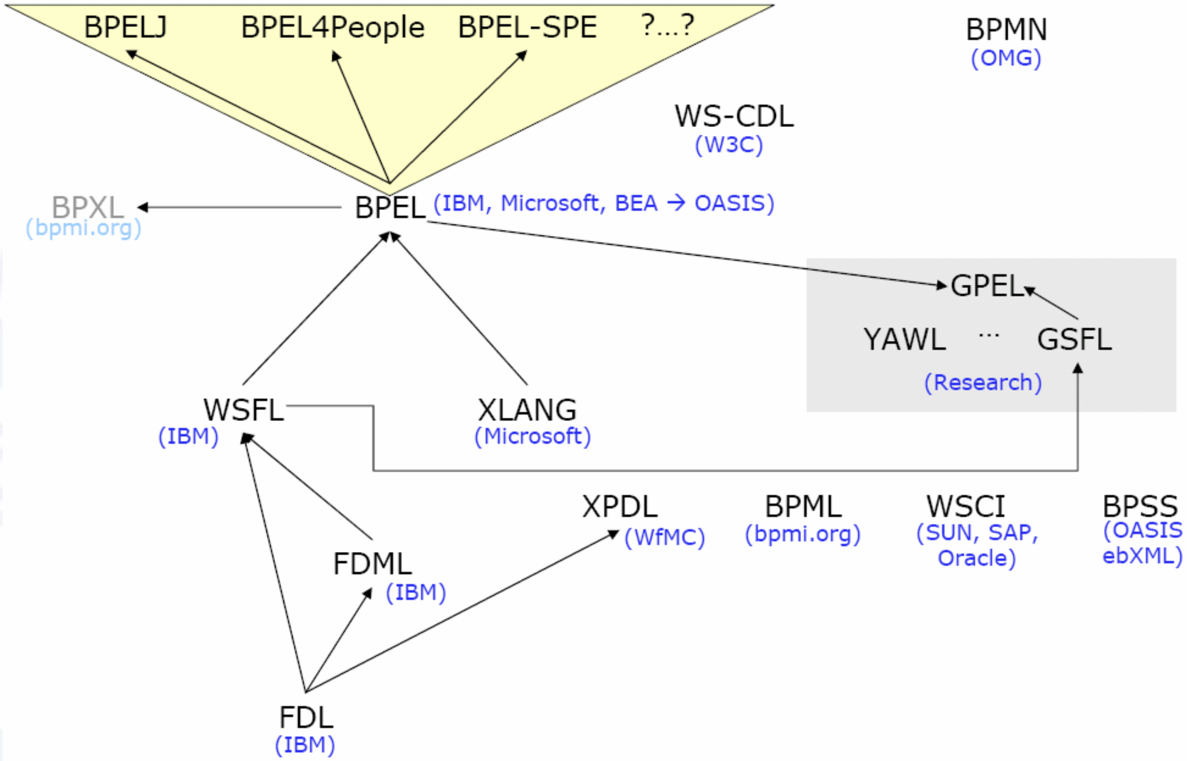
Dimensions in Workflow

- Workflows have at least 3 dimensions (Leymann, Roller [Production Workflow]):
  - What? = control logic dimension → what task should be executed
  - Who? = Organization dimension who must execute a task? → a role or person in an organization
  - What with? = infrastructure dimension → what program or tool must be used to execute a task
- agreement on the number of dimensions was never reached
- there are multiple notations and languages for workflows



© Dimka Karastoyanova

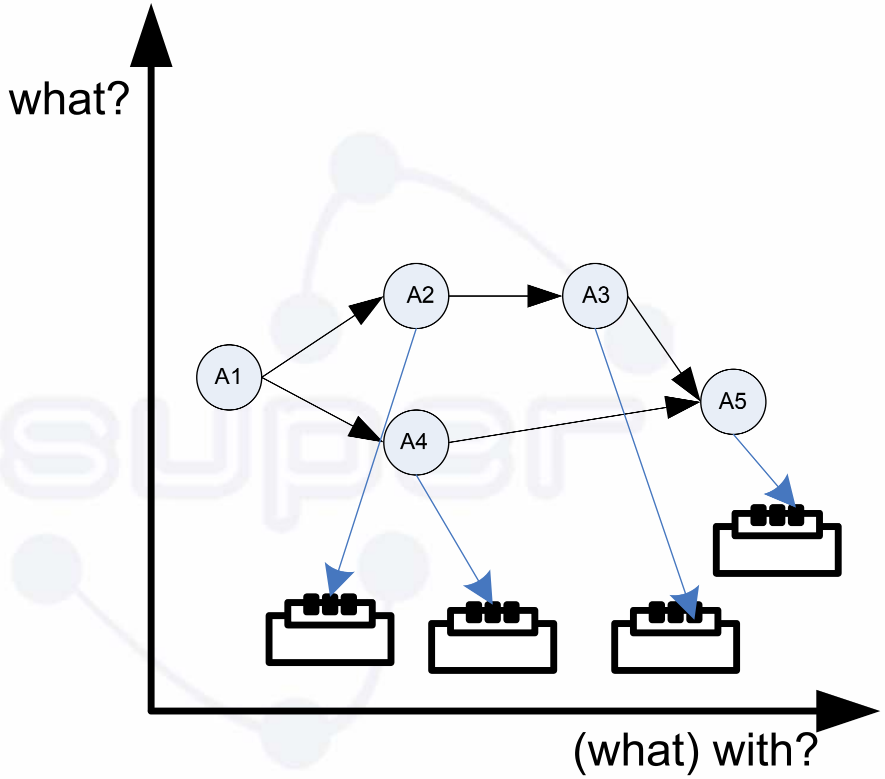
Workflow Languages



- Industry has tried to get an agreement on a common workflow language since the early 90's
- Now the industry agreed on BPEL (**that's why it's interesting**)
  - Portability
  - Interoperability

© Frank Leymann, Joerg Nitzsche

Dimensions in Web Service Flows



- WS-Flows are workflows that use only Web Services (as participants)
- Complete utilization of the advantages of Web Services - integration
- WS-Flows have only 2 dimensions
  - Control Logic
  - Infrastructure = Web Services
- No direct support for people/organizational dimension yet
- But efforts towards involving people in BPEL already exist: BPEL4People

Web Service Technology: The Key Thing!

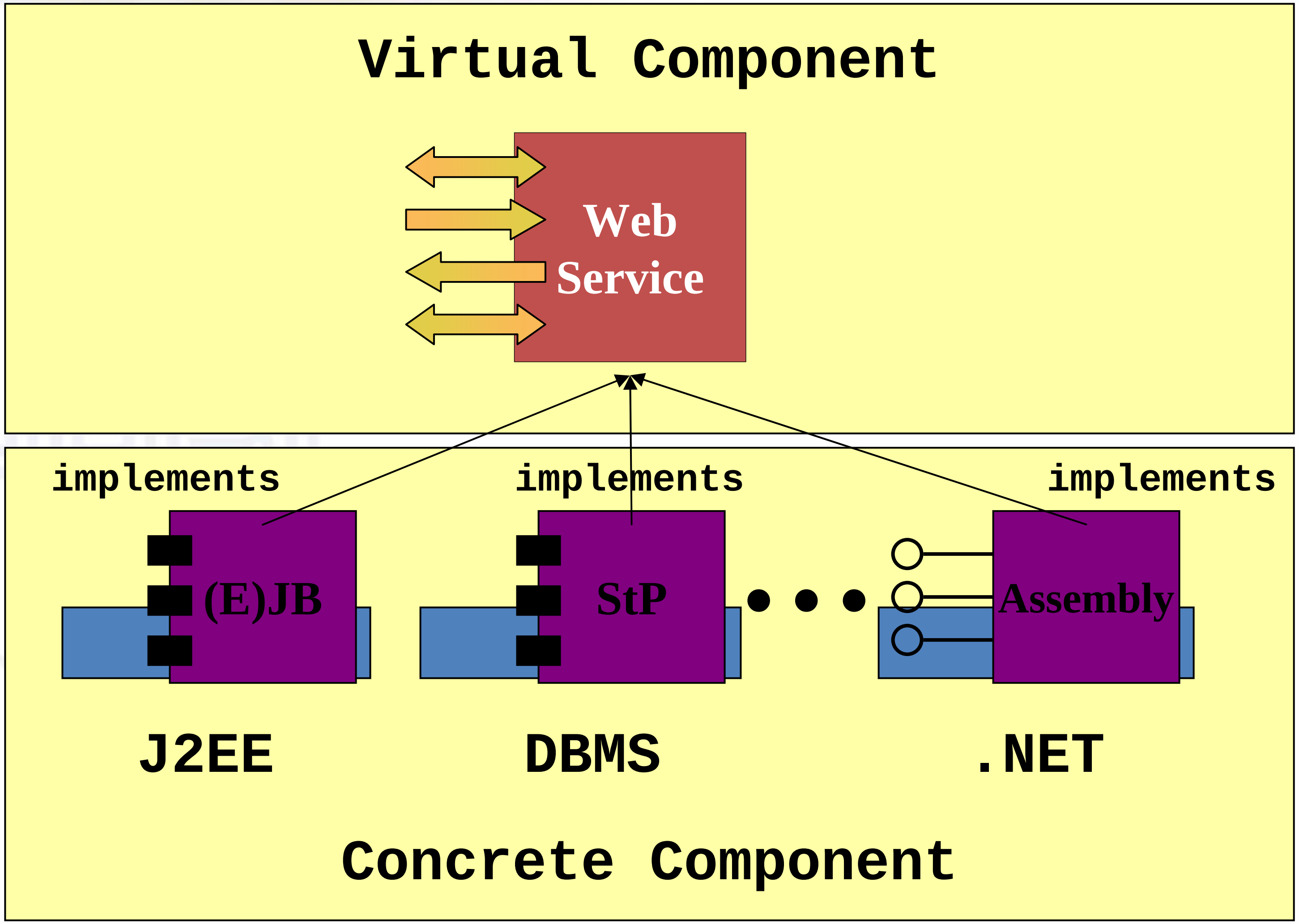
1. Web service technology provides a "virtual component model" for using components in a loosely coupled manner

2. When using a Web service the supporting container hides its "middleware idiosyncrasies"
   (component model behind the implementation of the Web service, the invocation protocol etc.)

3. Web service technology does *not* provide a new component model for implementing components
   (well, except for BPEL ☺)

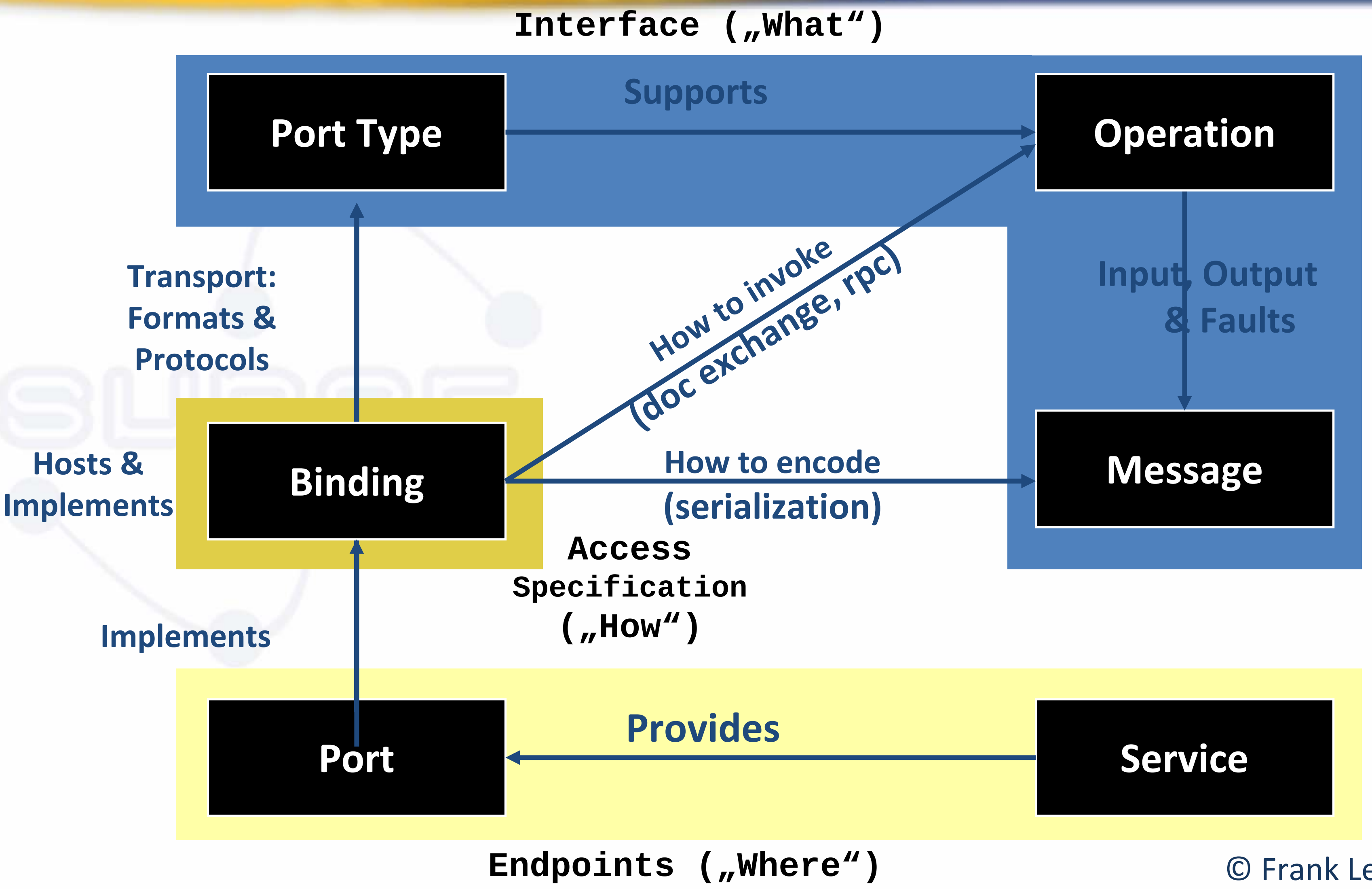
© Frank Leymann

Virtualizing Components



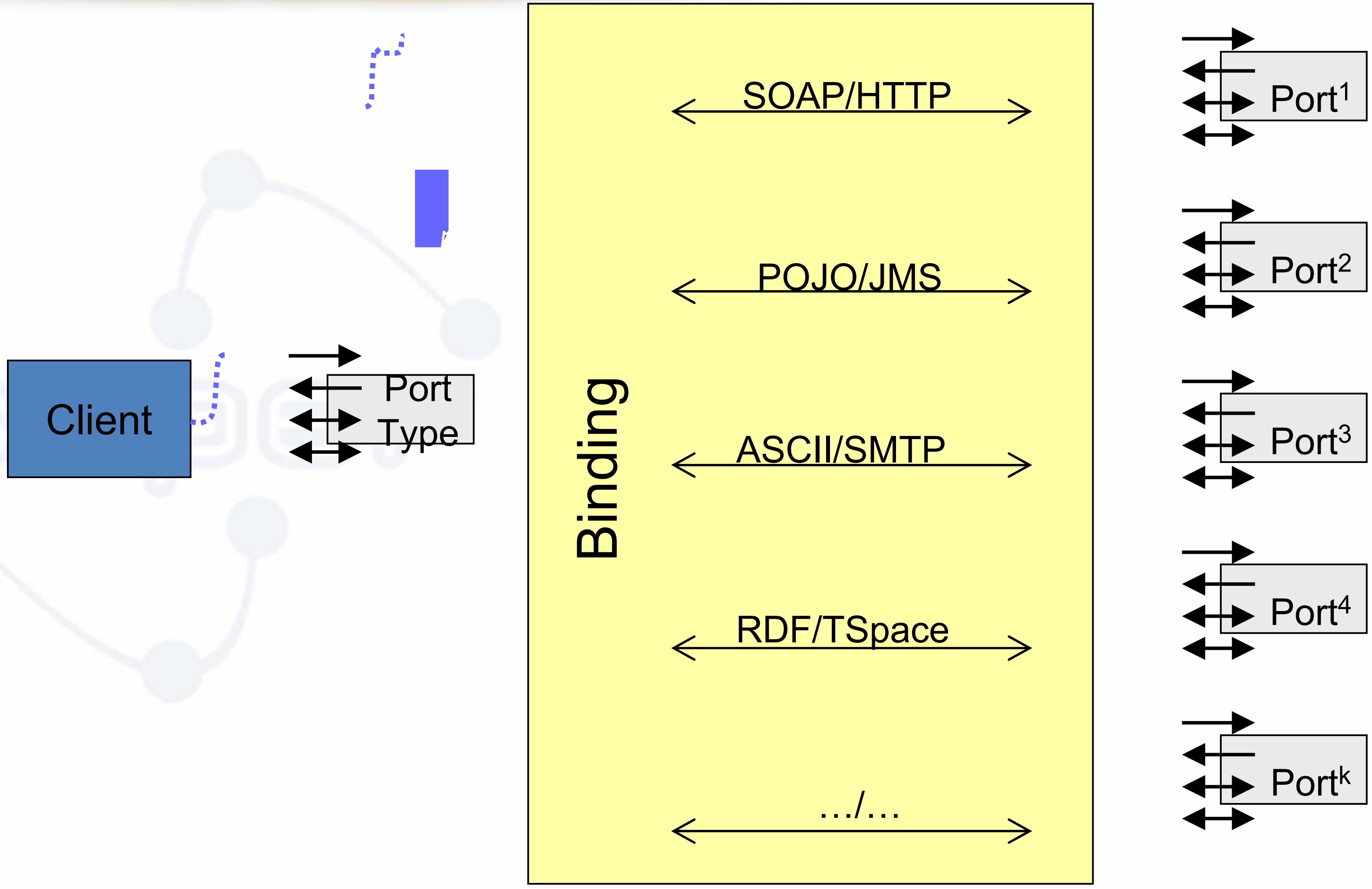
© Frank Leymann

Ingredients Of WSDL



© Frank Leymann

The Role of Bindings



© Frank Leymann, Joerg Nitzsche

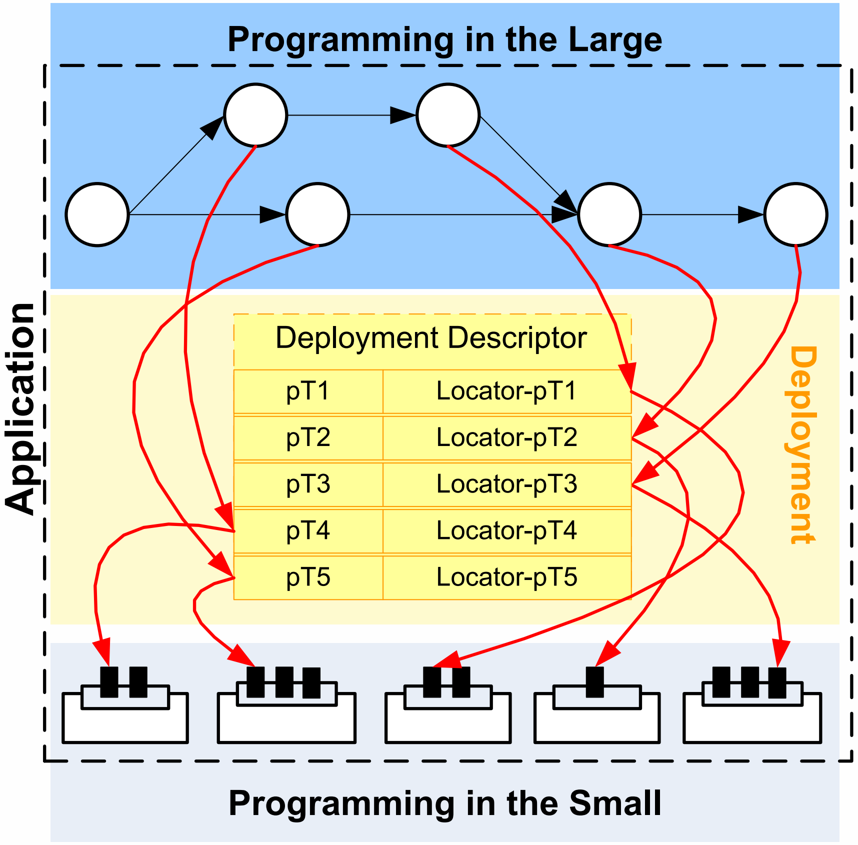
WS-Interoperability

- WS-I (Web Services Interoperability)
  - Consortium to ensure interoperability
    - Specification of so-called *profiles* (“regulation” of how and what to use)
    - Tools (test, analysis, monitor)
    - Sample implementation
- WS-I profile(s) compliance ensures common behavior
  - Specifications sometimes “terse”
    - Interpreted differently by different vendors
    - WS-I provides common interpretation resolving ambiguities
  - Subsetting of features of a specification

→ Interoperability is the result

Processes, Web Services and Bindings

- WS-Flows maintain the two-level programming approach
- And comply with the life cycle of Workflows
- BUT an additional phase in the life cycle has been introduced
  - Deployment
- During deployment one specifies binding information for partner services
- Improved configurability and reusability
- Criteria for dynamic binding can also be provided in the deployment descriptor



Deployment Descriptor	
pT1	Locator-pT1
pT2	Locator-pT2
pT3	Locator-pT3
pT4	Locator-pT4
pT5	Locator-pT5

© Dimka Karastoyanova

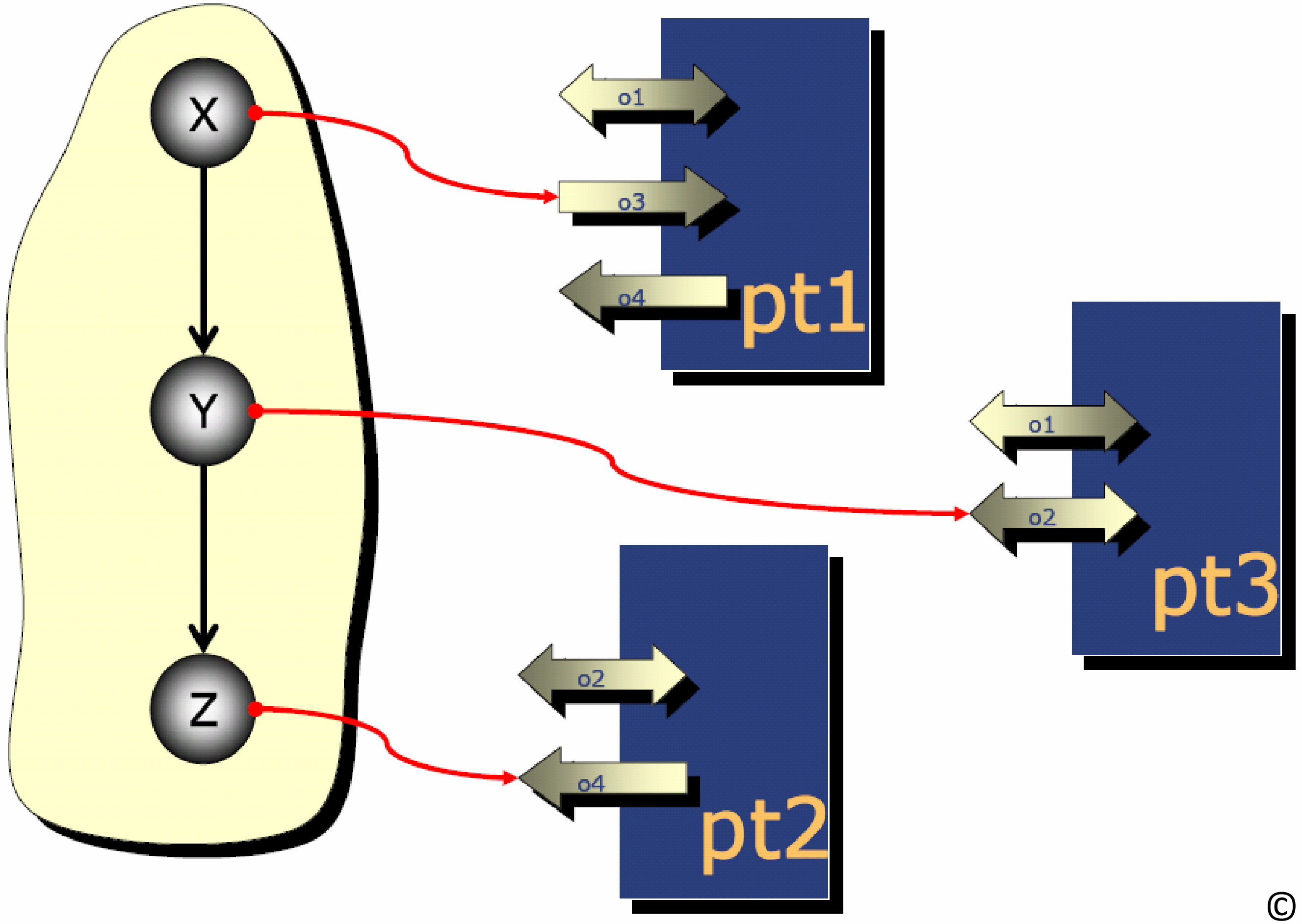
What is BPEL?

- A language to specify behaviour of business processes
  - Between Web services…
  - …and as Web service
- Executable processes
  - Can be performed within all compliant environments (portability)
  - Interoperability between heterogeneous environments
- Abstract processes
  - Specify constraints of message exchange
  - Are “views” on internal processes
- Combination of graph-based language (IBM WSFL) and calculus-based language (Microsoft XLANG)

© Frank Leymann

Business Processes Compose Web Services

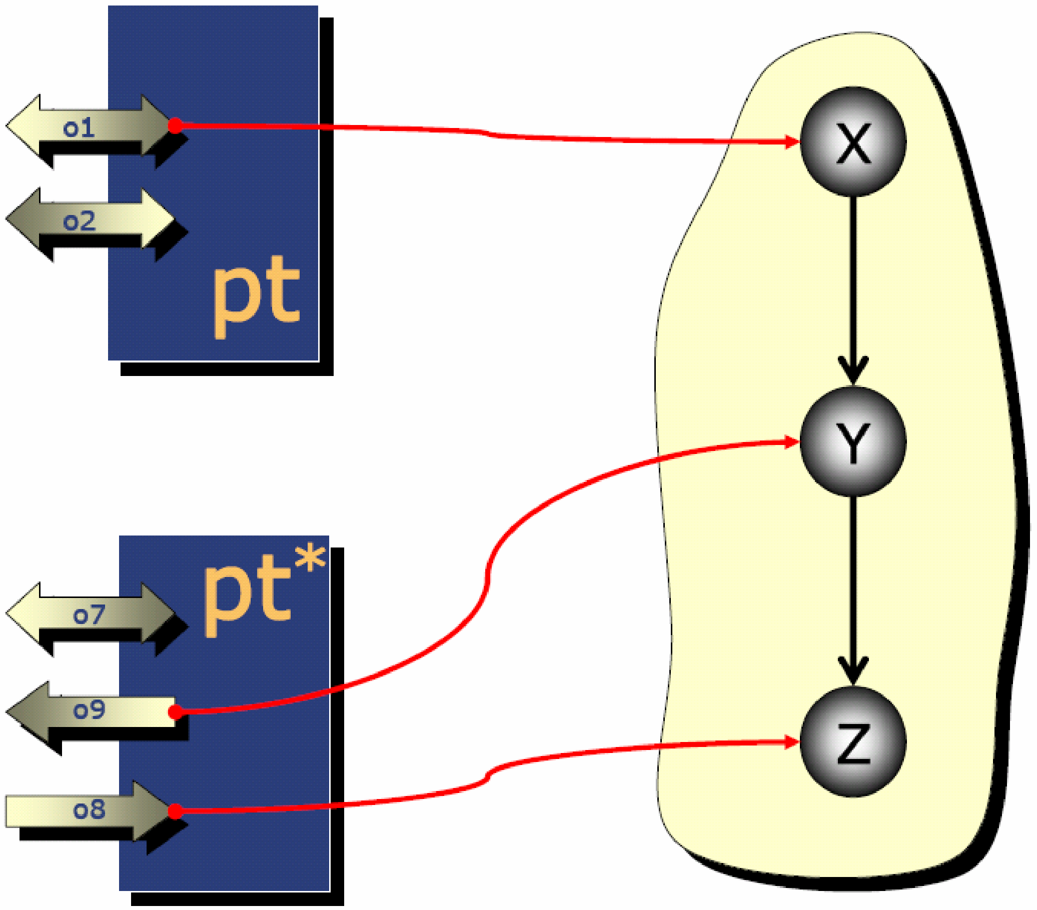
- BPEL Process uses Web services



© Frank Leymann

Business Processes as Web Services

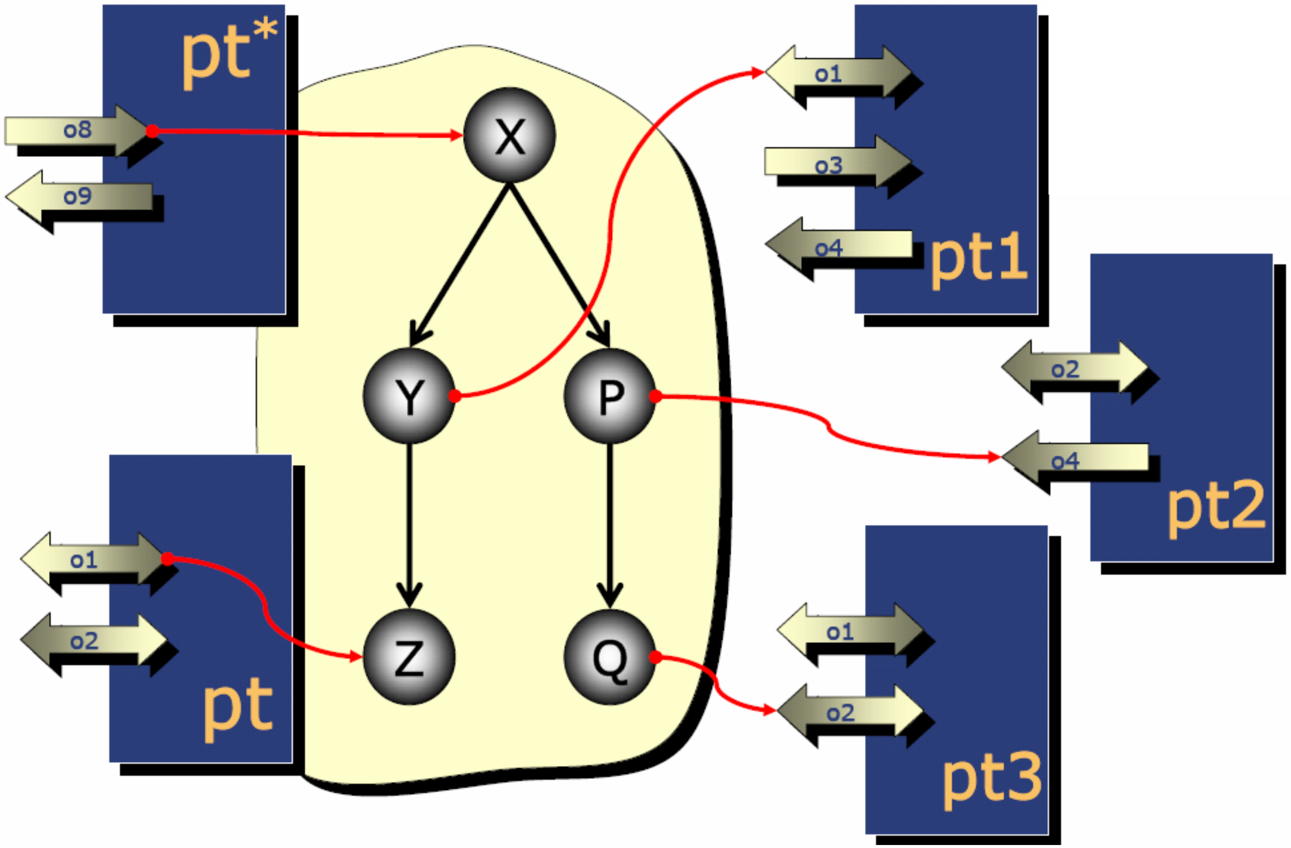
- A BPEL Process is also a Web Service
  - ▶ Provides functionality in terms of WSDL port types and operations



© Frank Leymann

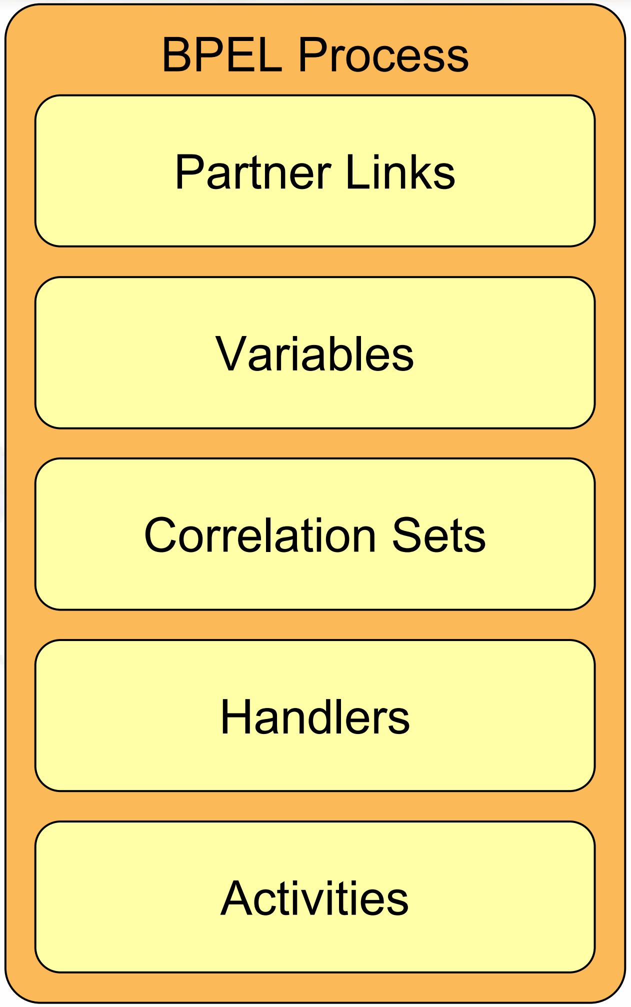
Aggregating Web services

BPEL provides a recursive aggregation model for Web services



© Frank Leymann

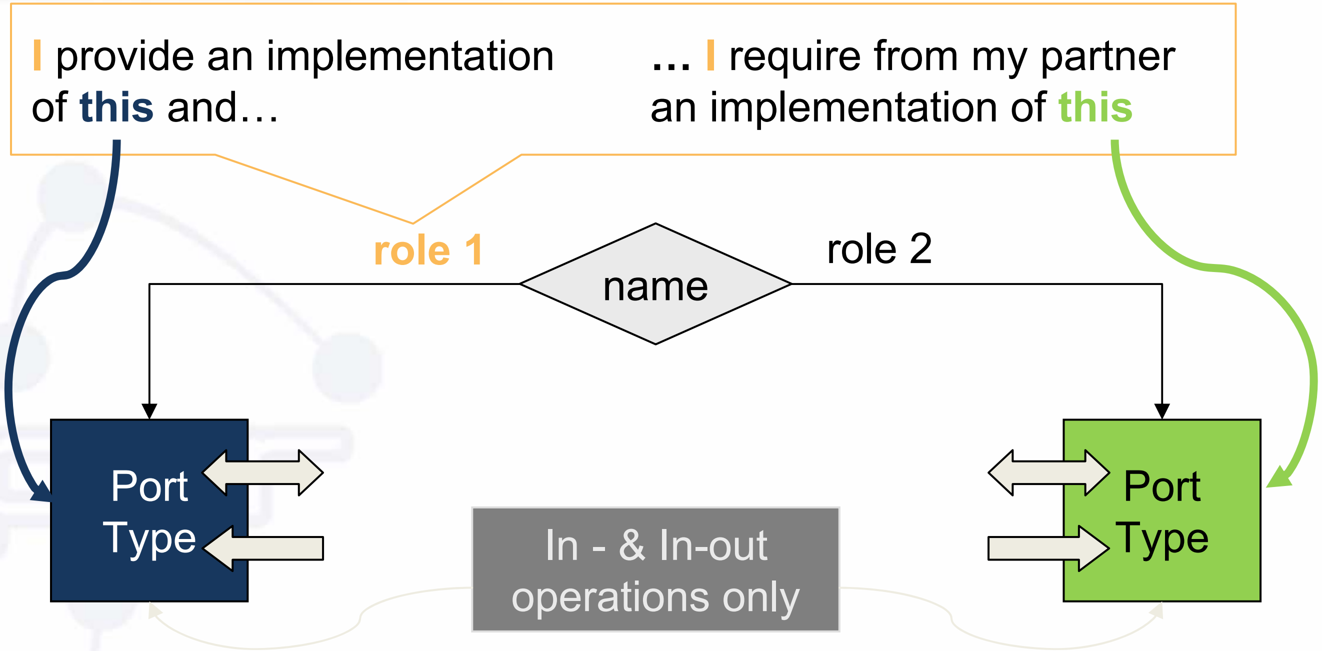
BPEL Elements



- Partner Links specify the roles of all external partners involved in the process as well as the role(s) of the process itself
- Variables can be defined either in the process or in a scope
  - They are used as input- and output-containers of interaction activities as well as assign activities
- Correlation Sets are used to correlate messages that belong to the same process instance
- Handlers can be used to define exception handling and compensation
- **The activities define the actual control logic**

© Joerg Nitzsche

partnerLinkTypes



I provide an implementation of **this** and…

… I require from my partner an implementation of **this**

role 1

name

role 2

Port Type

In - & In-out operations only

Port Type

- partnerLinkType is:
  - Bi-directional typed connector
  - A mutual call-back dependency
  - Specifies one or two roles; a port type per role
  - Messages exchanged between partners
  - A promise to playing a role is equal to a contract

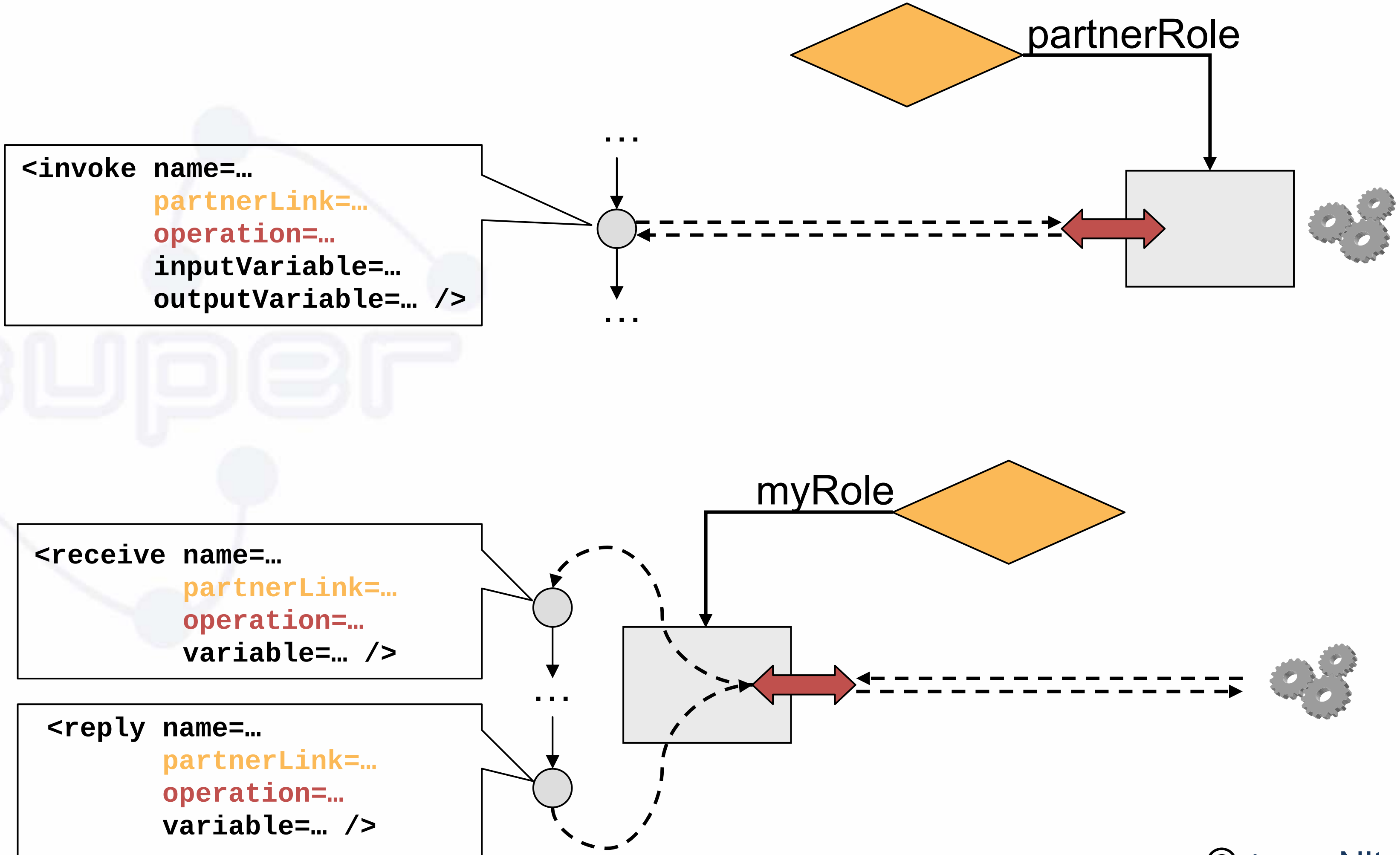
© Frank Leymann, Joerg Nitzsche, Dimka Karastoyanova

Communication and Control Flow

- Elements:
  - Interaction activities
    - Receive, Reply, Invoke, Pick
  - Complex activities for control flow
    - Sequence, Flow, If, While, RepeatUntil, ForEach
  - Data manipulation
    - Assign
  - Exception handling
    - Throw, Rethrow, Fault Handlers, Compensation Handlers
  - Reaction to Events
    - Pick, Event Handlers
- Instantiation is implicit – use <receive> or <pick>
  - With the “createInstance” attribute set to "yes"
  - To instantiate a process using Pick
    - The events in the <pick> MUST all be <onMessage> events
    - Pick reacts on one <onMessage> event

© Zhilei Ma

Communication: Synchronous invocation



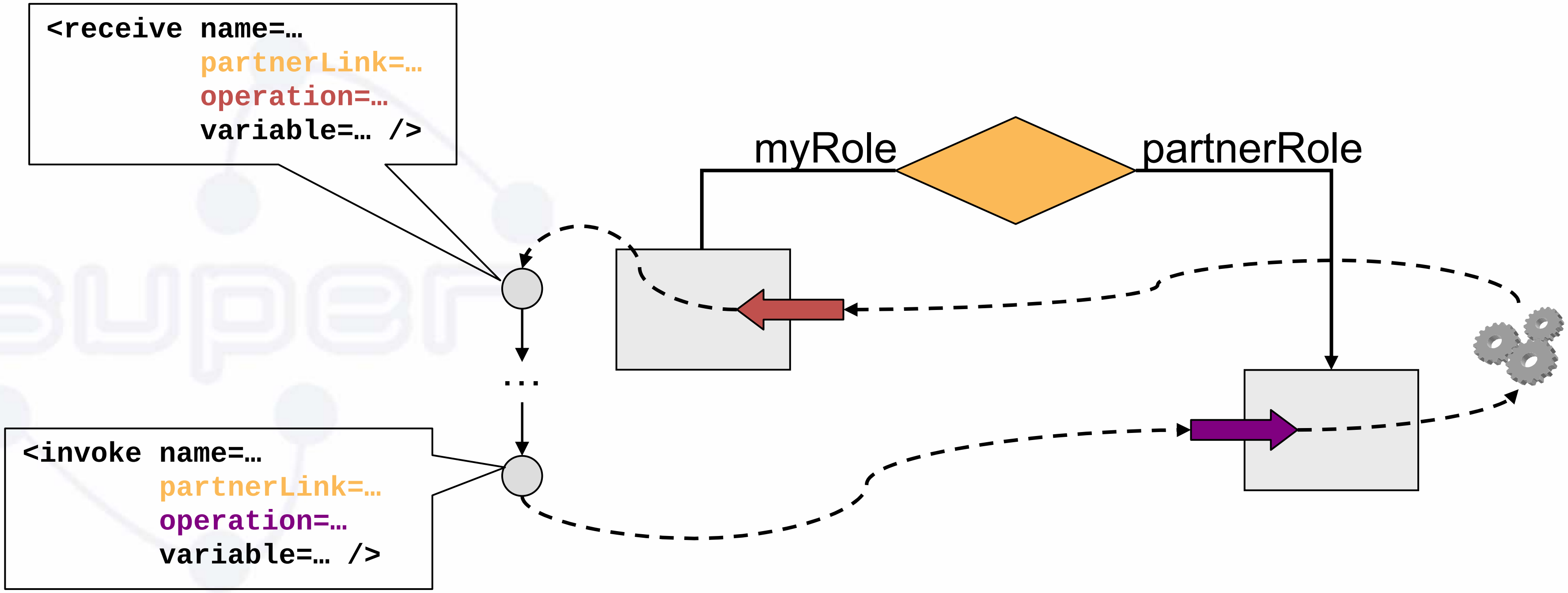
```
<invoke name=…
        partnerLink=…
        operation=…
        inputVariable=…
        outputVariable=… />
```

```
<receive name=…
         partnerLink=…
         operation=…
         variable=… />
```

```
<reply name=…
       partnerLink=…
       operation=…
       variable=… />
```

© Joerg Nitzsche

Communication: Asynchronous invocation (1)

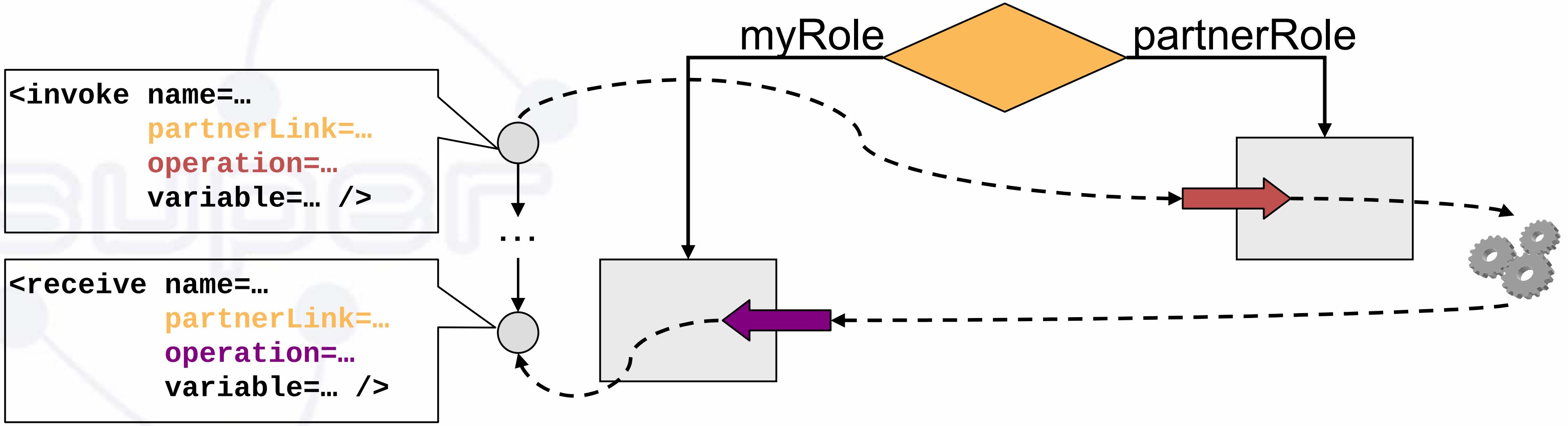


```
<receive name=…
         partnerLink=…
         operation=…
         variable=… />
```

```
<invoke name=…
        partnerLink=…
        operation=…
        variable=… />
```

© Joerg Nitzsche

Communication: Asynchronous invocation (2)



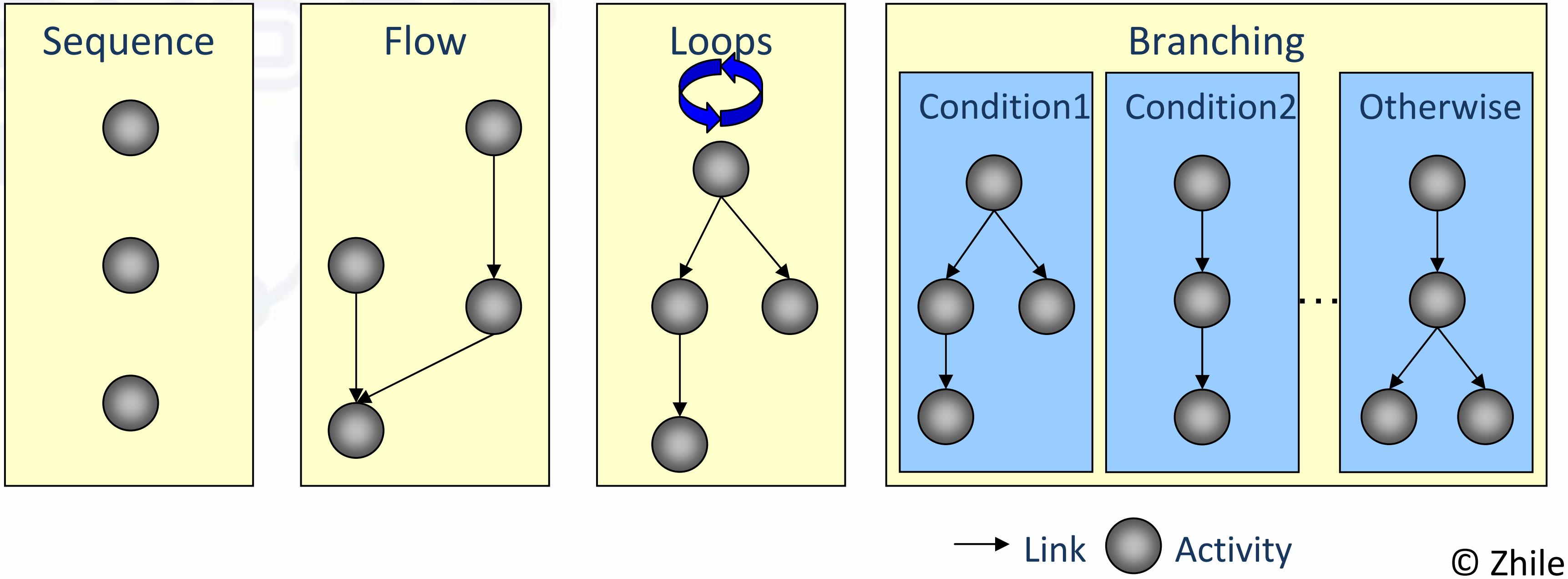
```
<invoke name=…
        partnerLink=…
        operation=…
        variable=… />
```

```
<receive name=…
         partnerLink=…
         operation=…
         variable=… />
```

© Joerg Nitzsche

Control Logic

- Sequential execution – sequence
- Parallel execution of tasks →Flow, links
- Branching → if then else
- Loops: while, repeat until



© Zhilei Ma

Data Manipulation

- Data Flow in BPEL
  - No explicit data flow modelled
  - Data flow implicit via global/scoped variables, access via name
  - Variables can be defined as
    - WSDL message type
    - XML Schema type (simple or complex)
    - XML Schema element
- Data Manipulation
  - BPEL Assign activity
  - Allows copying (parts of) variables
    - XPath can be used to identify these parts
- Example

```
<assign>
    <copy>
        <from>$po/lineItem[@prodCode=$myProd]/amt * $exchangeRate</from>
        <to>$convertedPO/lineItem[@prodCode=$myProd]/amt</to>
    </copy>
</assign>
```

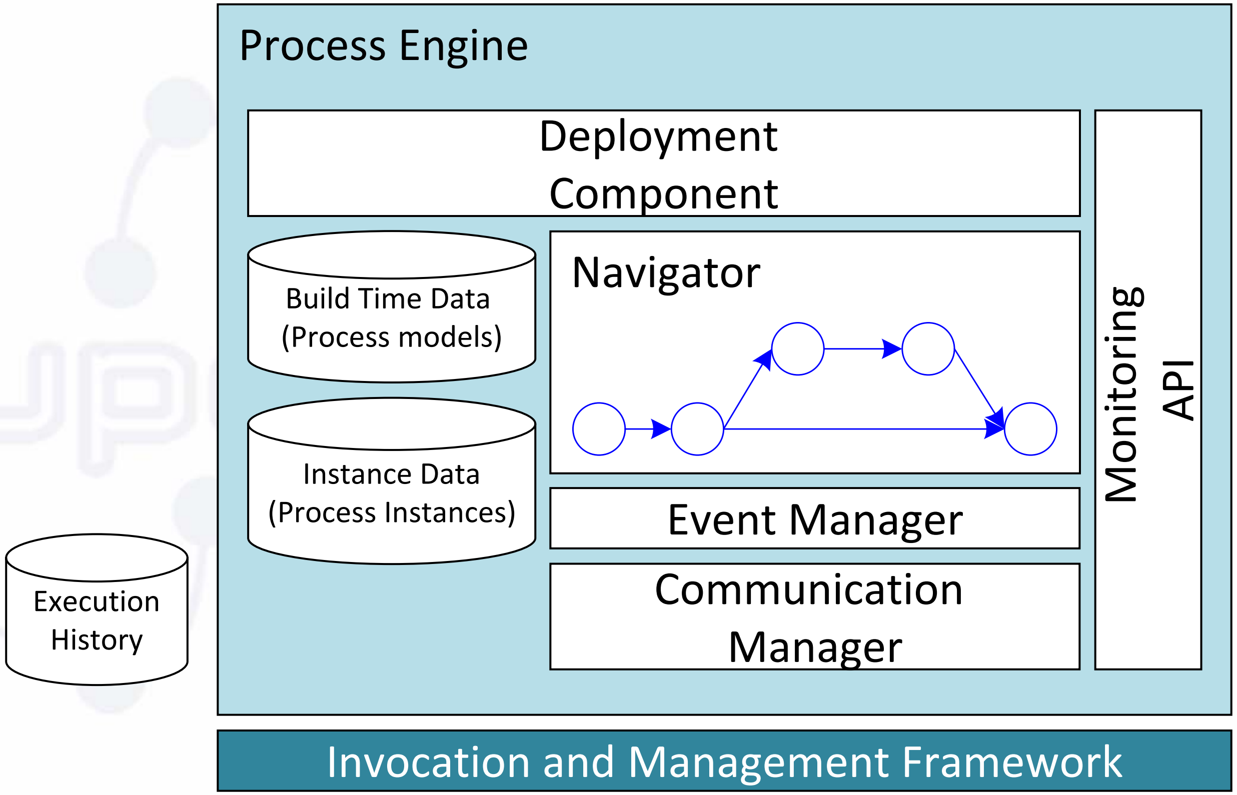
© Tammo van Lessen

Exception Handling

- Scopes are constructs denoting
  - Units of functionality
  - Have all-or-nothing semantics
- Exception Handling
  - FaultHandler – provide an alternative path in a process, executed for a particular type of failure
  - CompensationHandler – reverse the effects of successfully executed activities
    - Can be called by compensation handlers
    - Default and custom-defined compensation handlers
- Reaction to external events
  - EventHandler – used to specify actions to be taken upon an external event
  - Executed in parallel to the process/activity/scope

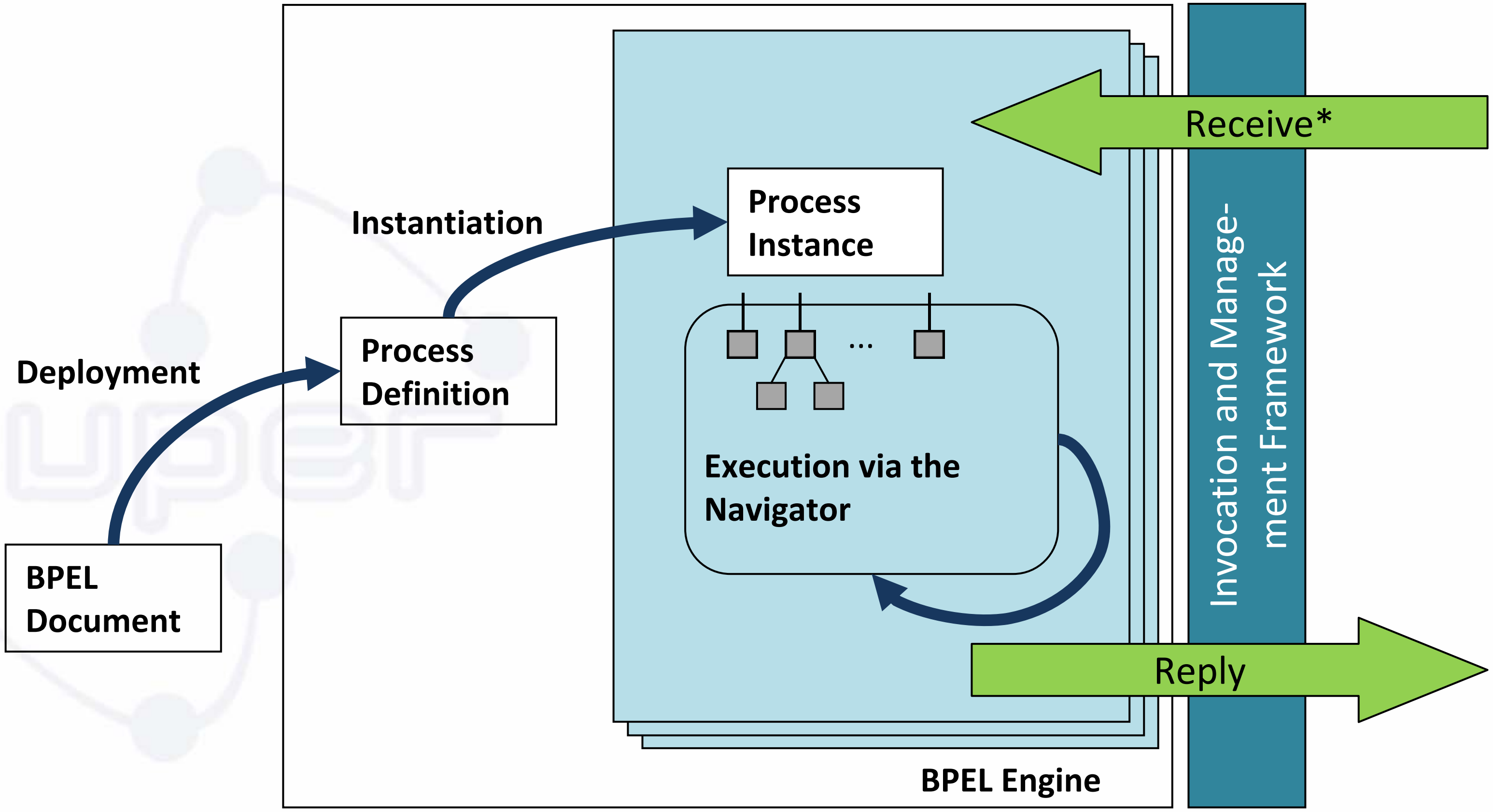
© Dimka Karastoyanova

Architecture of a BPEL Engine



© Dimka Karastoyanova, Tammo van Lessen, Joerg Nitzsche

Process Lifecycle within the Engine



* Receive may cause an Instantiation of a Process

© Joerg Nitzsche

Semantics Utilised for Process Management within and between Enterprises

Semantic Web Services

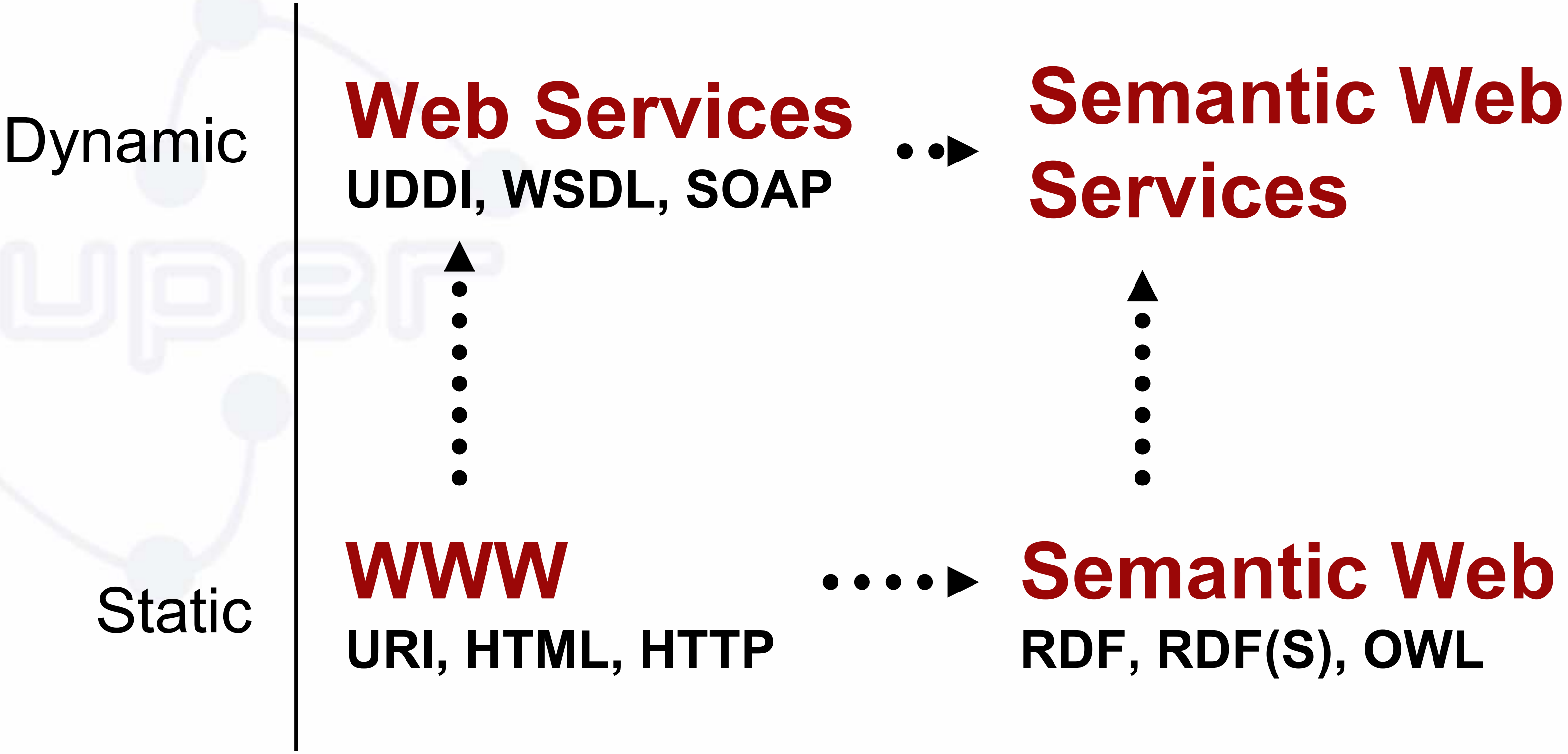
Outline

- Introduction
  - The need of semantics for realizing the SOA vision
  - Semantic Web – Web Services – SWS

- SWS Frameworks
  - Requirements
  - The WSMO Approach
    1. Ontologies
    2. Goals
    3. Web services
    4. Mediators

- SWS Techniques
  - Discovery
  - Composition
  - Mediation

The Idea – Realizing "Semantic SOA"



Dynamic

Web Services

UDDI, WSDL, SOAP

Semantic Web Services

Static

WWW

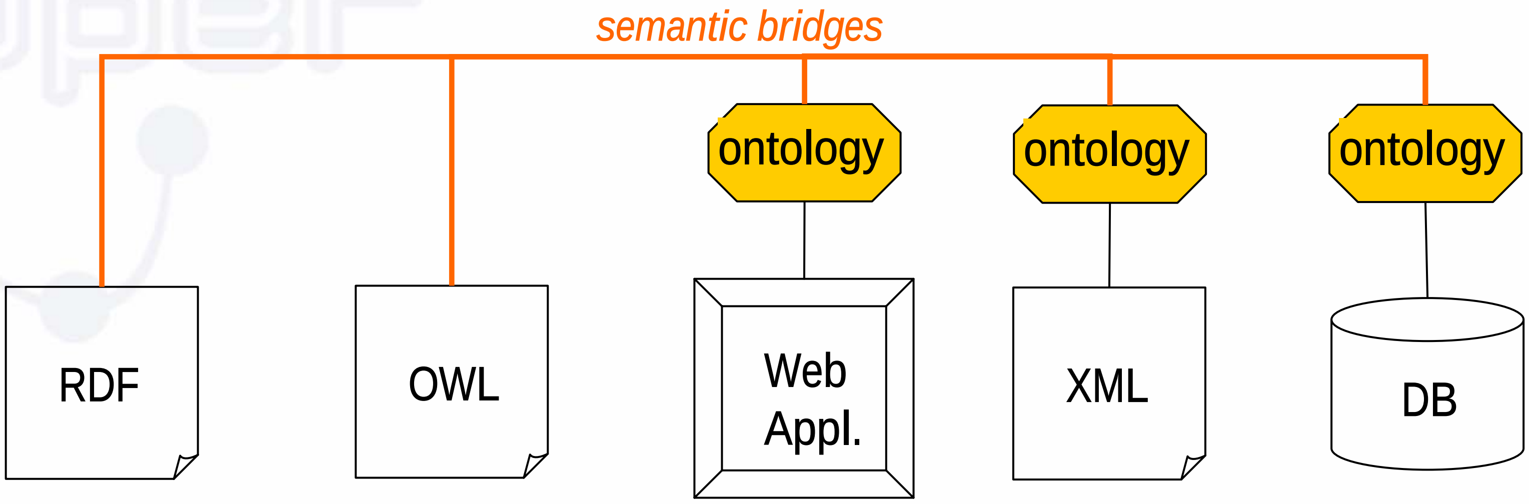
URI, HTML, HTTP

Semantic Web

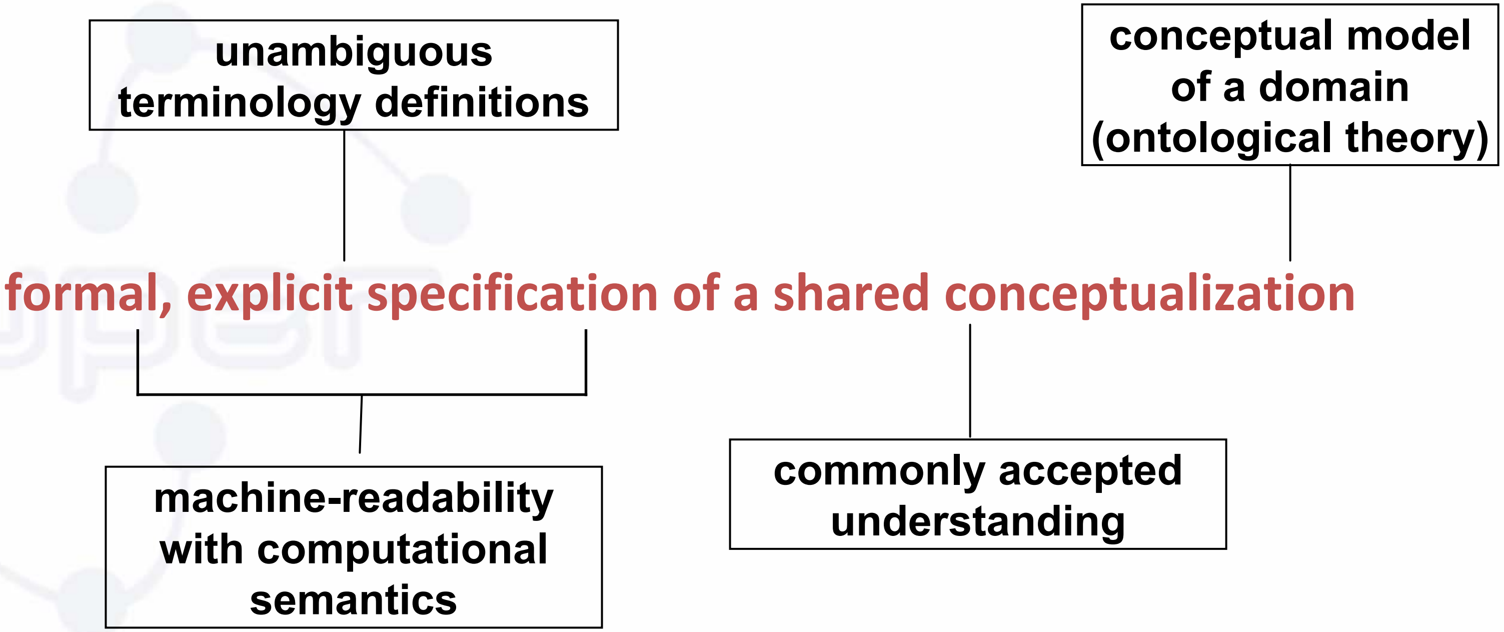
RDF, RDF(S), OWL

The Semantic Web

- next generation of the Internet (augmentation of the WWW)
- information has machine-processable and machine-understandable semantics
- ontologies as base technology for semantic interoperability



Ontology Definition



unambiguous terminology definitions

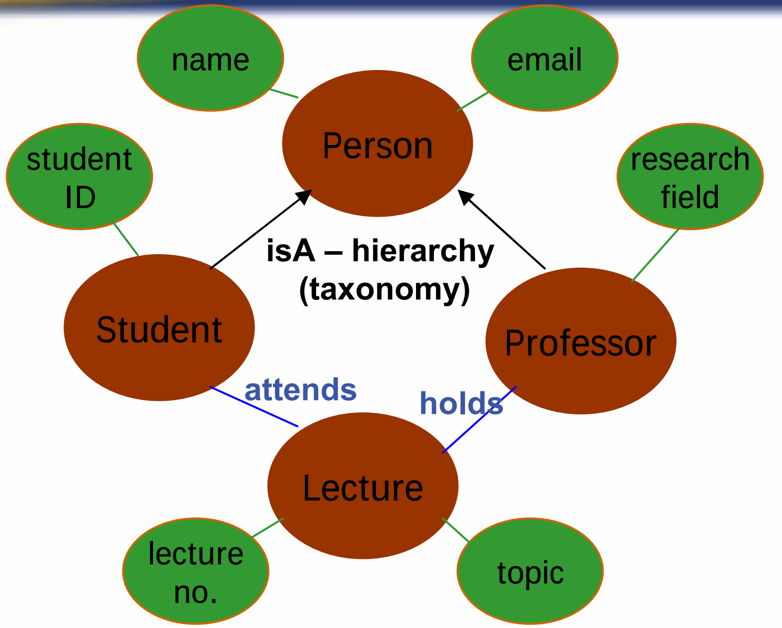
conceptual model of a domain (ontological theory)

formal, explicit specification of a shared conceptualization

machine-readability with computational semantics

commonly accepted understanding

Ontology Example



Concept

conceptual entity of the domain

Property

attribute describing a concept

Relation

relationship between concepts or properties

Axiom

coherency description between Concepts / Properties / Relations via logical expressions

holds(Professor, Lecture) =>
Lecture.topic = Professor.researchField

Instance

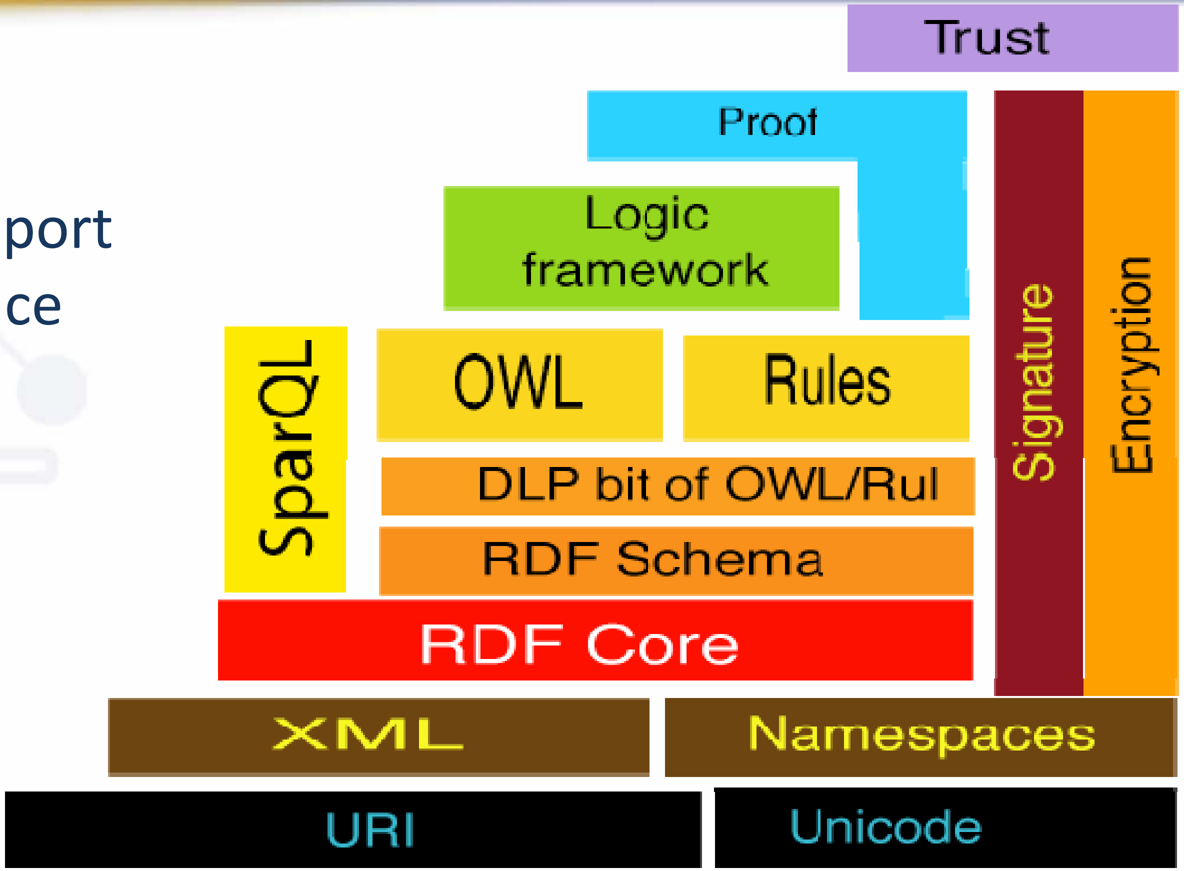
individual in the domain

Ann memberOf student
name = Ann Lee
studentID = 12345

Ontology Languages

Requirements

- expressivity
- reasoning support
- web compliance



W3C Semantic Web Language Layer Cake

revised version, Tim-Berners-Lee 2005

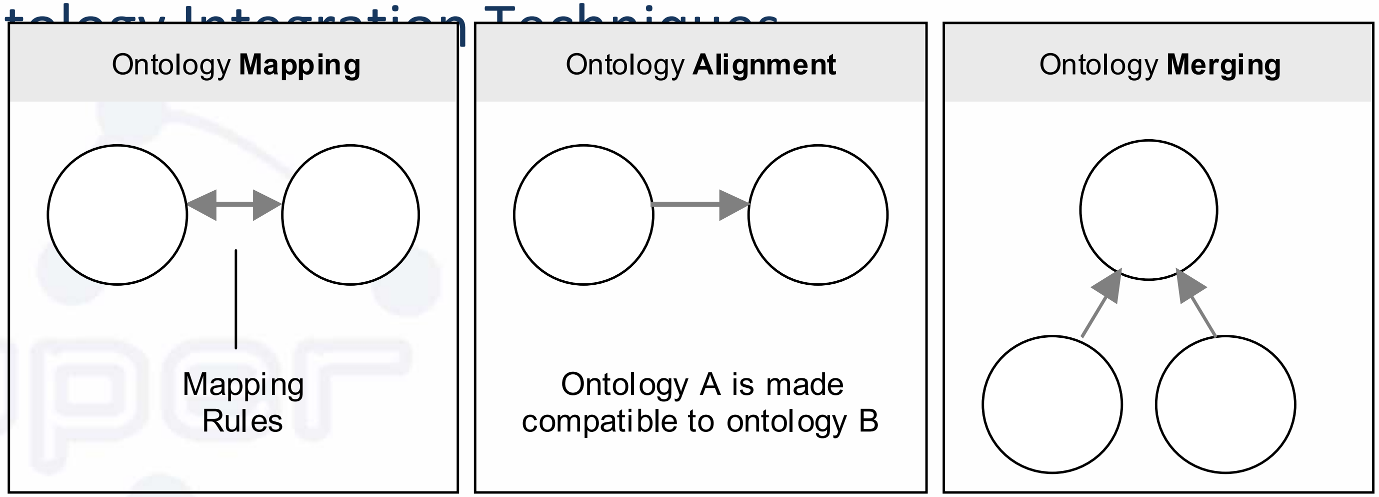
Ontology Technology

- **Ontology Reasoning**
  + advanced information processing
  – special requirements
    - large scale knowledge handling
    - fault-tolerant
    - stable & scalable inference machines

- **Ontology Management**
  – (collaborative) editing and browsing
  – storage and retrieval
  – versioning and evolution support

Ontology-Based Data Integration

- Ontology Integration Techniques



- integration on semantic level (domain independent)
- semi-automatic
  - human intervention needed for “integration decision
  - graphical support for ontology mapping as central technique

Web Services & SOA

- Web Service = program accessible over the Web
- Service-Oriented Architecture (SOA):

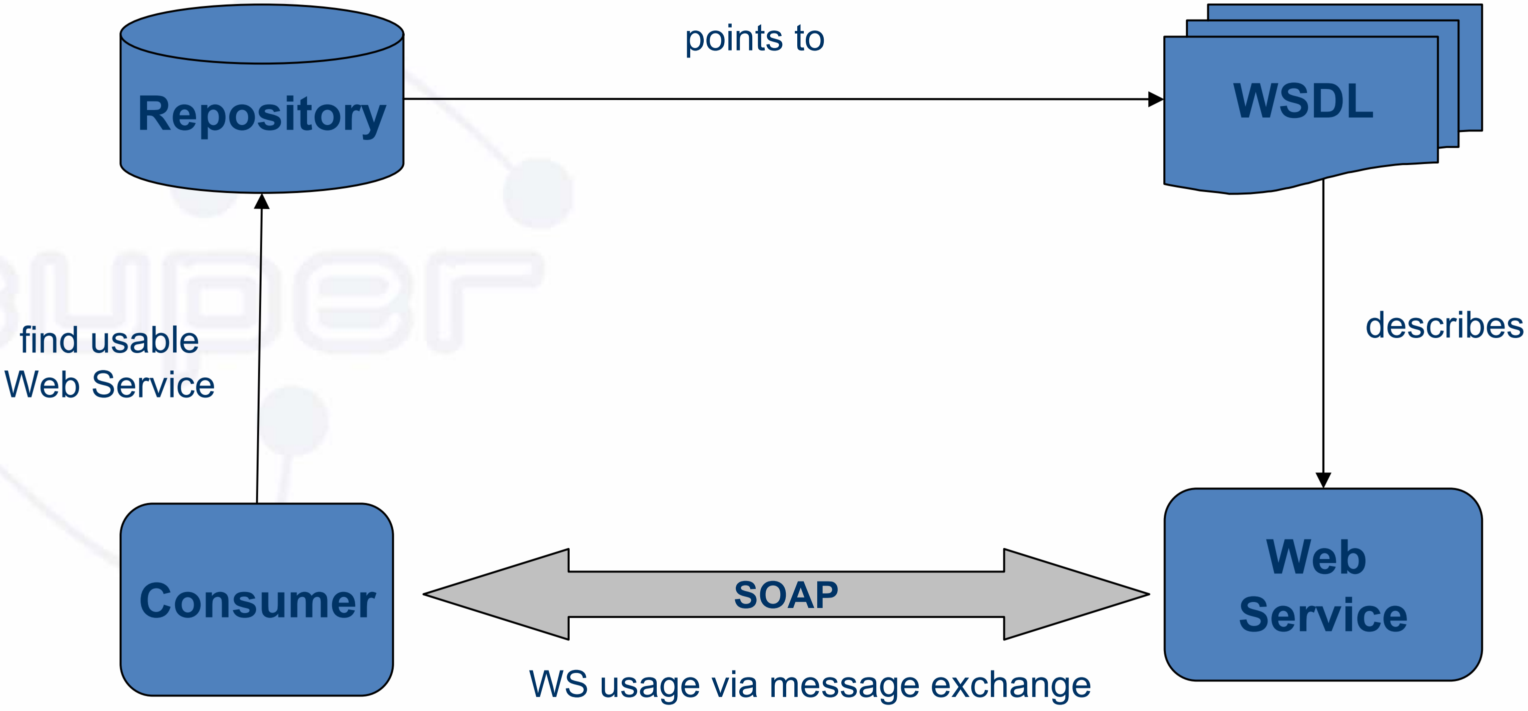
  dynamically find & invoke those Web services

  that allow to solve a particular request

- Web Service Technologies:
  1. **WSDL** Web Service Description Language
     - in- and outgoing messages
     - technical access (port type, protocol, etc.)
  2. **SOAP** XML data exchange protocol for the Web
  3. **UDDI** registry for Web Services

The Web Service Usage Process

web-based SOA as new system design paradigm



Deficiencies of WS Technology

- current technologies allow usage of Web Services
- but:
  - only syntactical information descriptions
  - syntactic support for discovery, composition and execution

  => Web Service usability, usage, and integration needs to be inspected manually

  - no semantically marked up content / services
  - no support for the Semantic Web

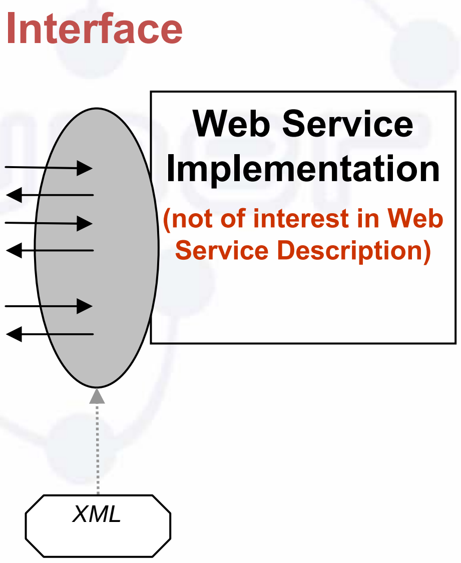
=> initial Web Service Technology Stack failed to realize the SOA Vision

Aim: Realize the SOA Vision

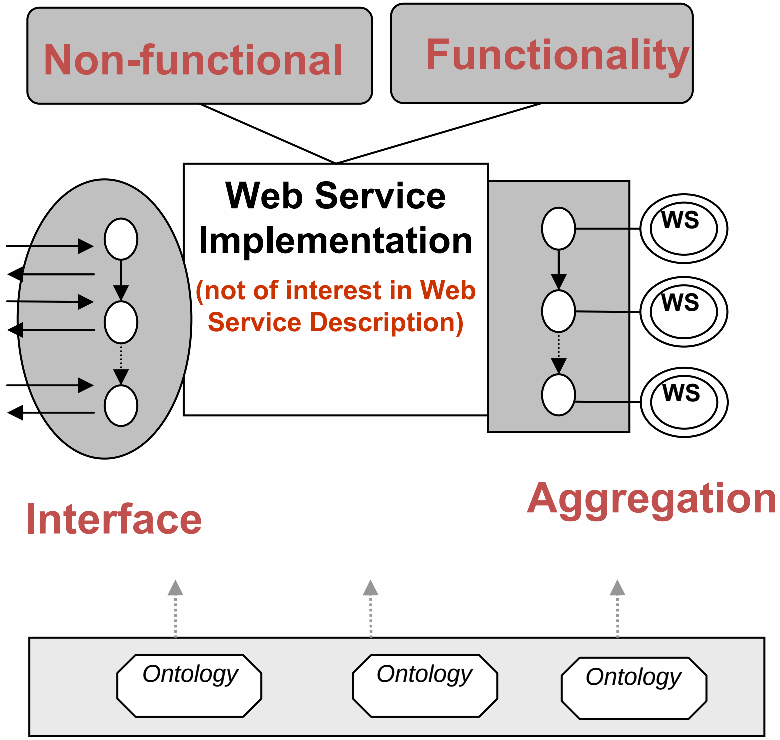
- automate Web Service technologies by
  1. **rich, formal annotation of Web Services**
  2. **inference-based techniques for automated discovery, composition, mediation, execution of Web Services**
- integration with the Semantic Web
  - ontologies as data model
  - Web Services as integral part
- semantic SOA
  - also semantically describe client requests
  - automate complete SOA process
  - semantically enhance SOA technology

Web Service Annotation

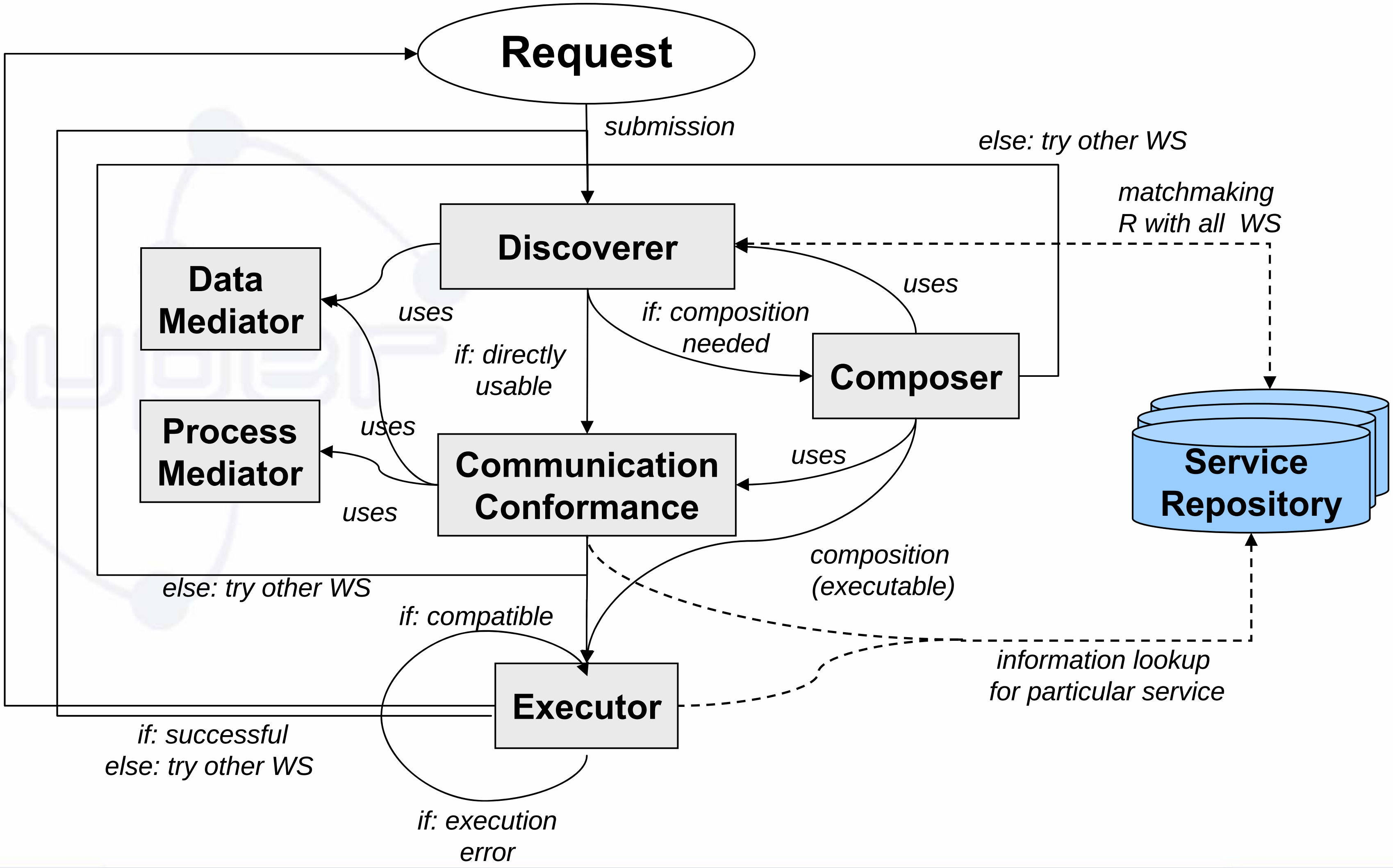
a) Web Service Description Structure


Interface
Web Service Implementation
(not of interest in Web Service Description)
XML

b) Semantic Web Service Description Structure



Semantic Web Service Technologies



Requirements & Frameworks

- **Requirements for SWS Frameworks**
  - **cover all aspects relevant for enabling automated Web service usage**
  - **define conceptual model & axiomatization (= semantics)**
  - **provide formal language for semantic descriptions**

- **Approaches (W3C Member Submissions)**

1. WSMO:	Ontologies, Goals, Web Services, Mediators
2. OWL-S	WS Description Ontology (Profile, Service Model, Grounding)
3. SWSF	Process-based Description Model & Language for WS
4. WSDL-S	semantic annotation of WSDL descriptions

Web Service Modeling Ontology (WSMO)

- **Comprehensive Framework for SESA**

  Semantically Empowered Service-Oriented Architecture

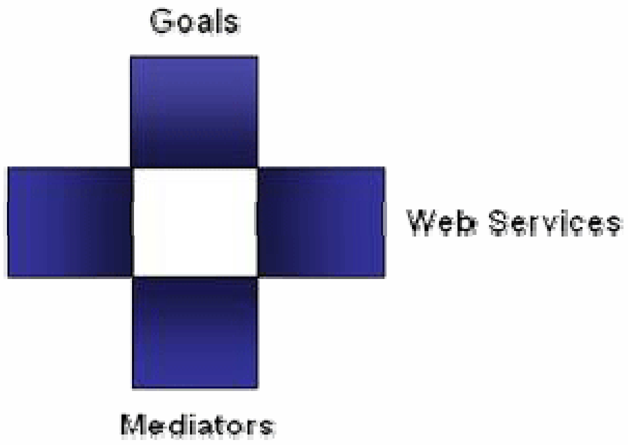
  – top level notions = SESA core elements
  – conceptual model + axiomatization
  – ontology & rule language

- **International Consortium (mostly European)**
  – started in 2004
  – 78 members from 20 organizations
  – W3C member submission in April 2005

www.wsmo.org

WSMO Top Level Notions

Objectives that a client wants to achieve by using Web Services



Formally specified terminology used by all other components

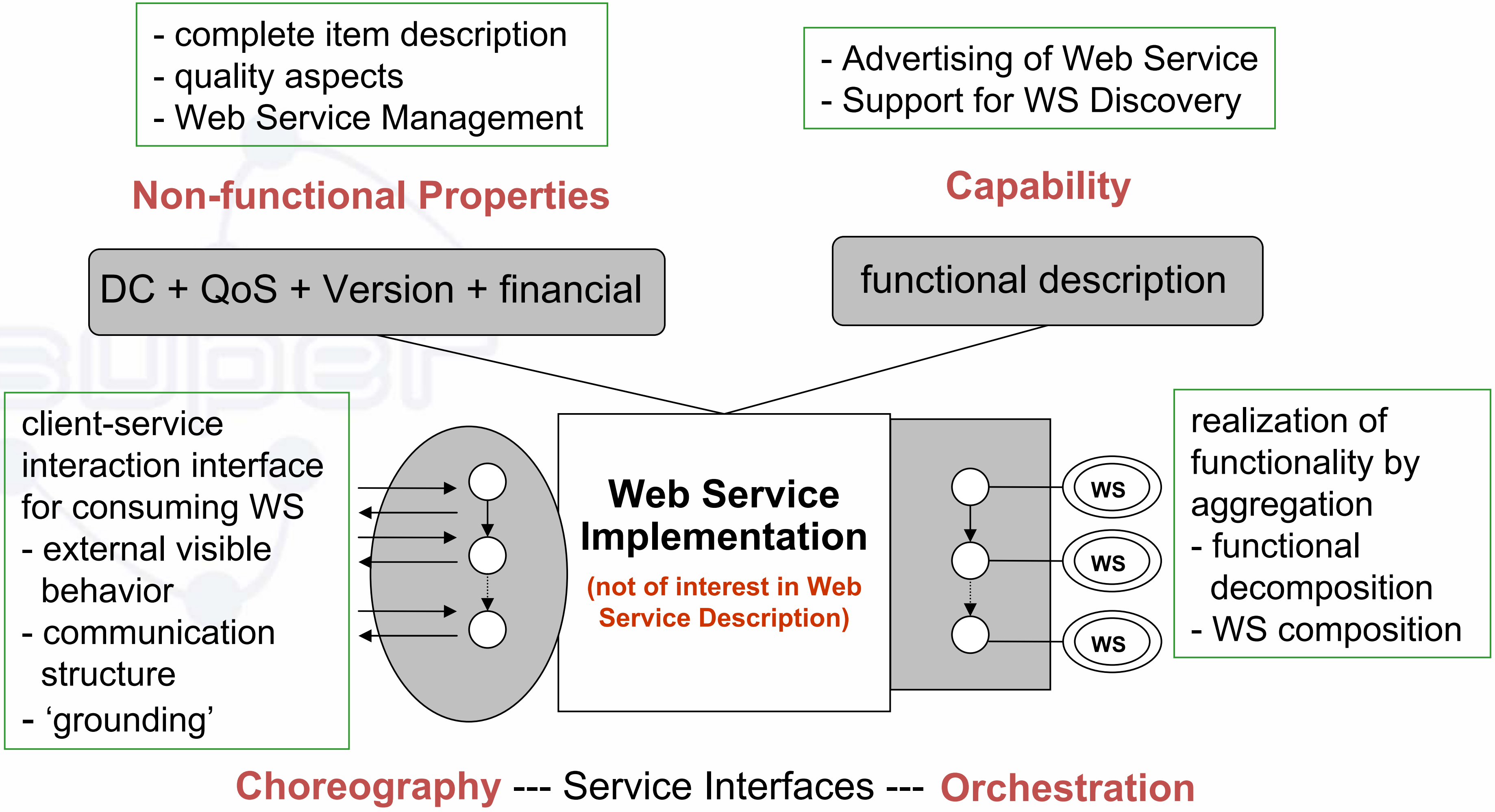
Semantic description of Web Services:

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

Connectors between components with mediation facilities for handling heterogeneities

W3C submission 13 April 2005

WSMO Web Service Description



- complete item description
- quality aspects
- Web Service Management

Non-functional Properties

DC + QoS + Version + financial

- Advertising of Web Service
- Support for WS Discovery

Capability

functional description

client-service interaction interface for consuming WS
- external visible behavior
- communication structure
- ‘grounding’

Web Service Implementation

(not of interest in Web Service Description)

realization of functionality by aggregation
- functional decomposition
- WS composition

Choreography --- Service Interfaces --- **Orchestration**

Capability Specification

- **Non functional properties**
- **Imported Ontologies**
- **Used mediators**
  - *OO Mediator:* importing ontologies with data level mismatch resolution
  - *WG Mediator:* link to a Goal wherefore service is not usable a priori
- **Shared Variables:** scope is entire capability
- **Pre-conditions**

  what a web service expects in order to be able to provide its service. They define conditions over the input.
- **Assumptions**

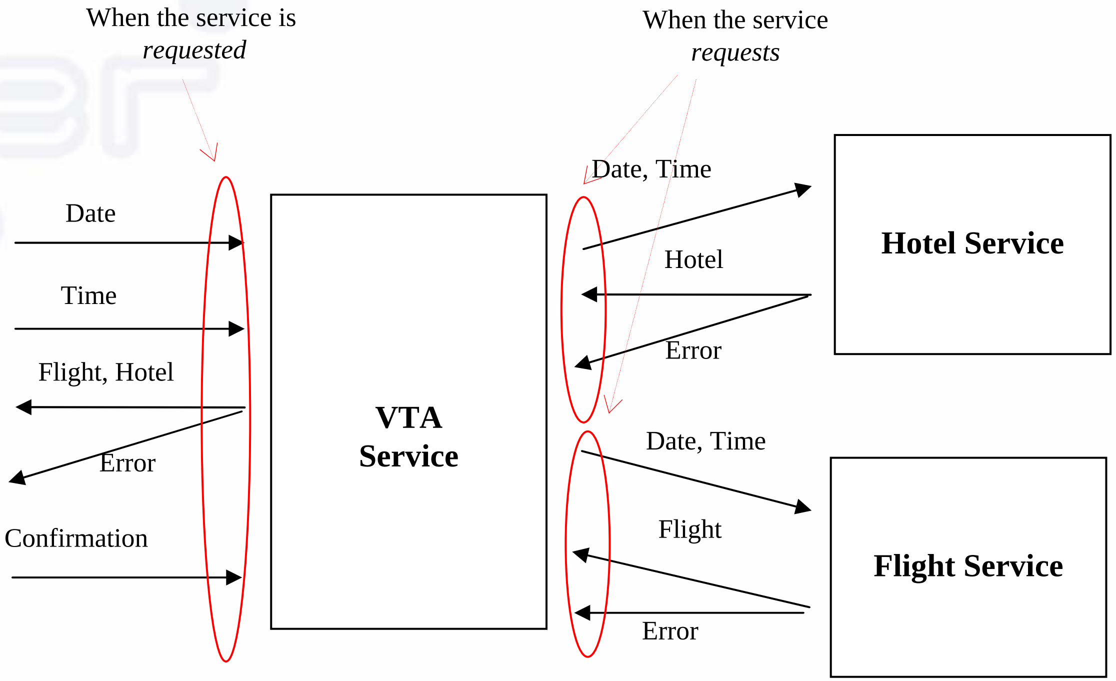
  conditions on the state of the world that has to hold before the Web Service can be executed
- **Post-conditions**

  describes the result of the Web Service in relation to the input, and conditions on it
- **Effects**

  conditions on the state of the world that hold after execution of the Web Service (i.e. changes in the state of the world)

Choreography & Orchestration

- **Choreography** = how to interact with the service to consume its functionality
- **Orchestration** = how service functionality is achieved by aggregating other Web Services



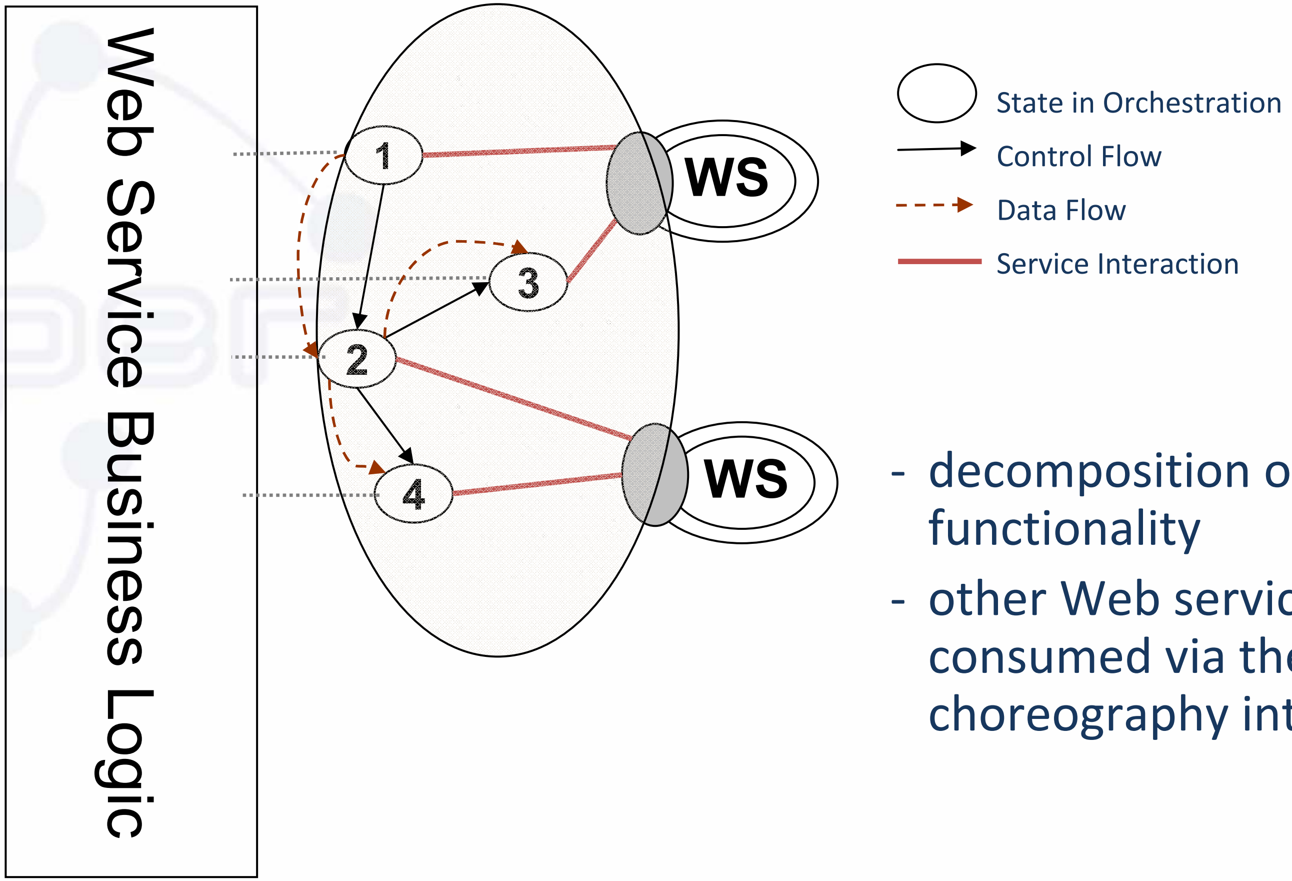
Choreography Interface

interface for consuming Web Service

- **External Visible Behavior**
  - those aspects of the workflow of a Web Service where Interaction is required
  - described by workflow constructs: sequence, split, loop, parallel
- **Communication Structure**
  - messages sent and received
  - their order (communicative behavior for service consumption)
- **Grounding**
  - executable communication technology for interaction
  - choreography related errors (e.g. input wrong, message timeout, etc.)
- **Formal Model**
  - reasoning on Web Service interfaces (service interoperability)
  - semantically enabled mediation on Web Service interfaces

Orchestration

interface for interaction with aggregated Web Services



- decomposition of service functionality
- other Web services consumed via their choreography interfaces

WSMO Web Service Interfaces

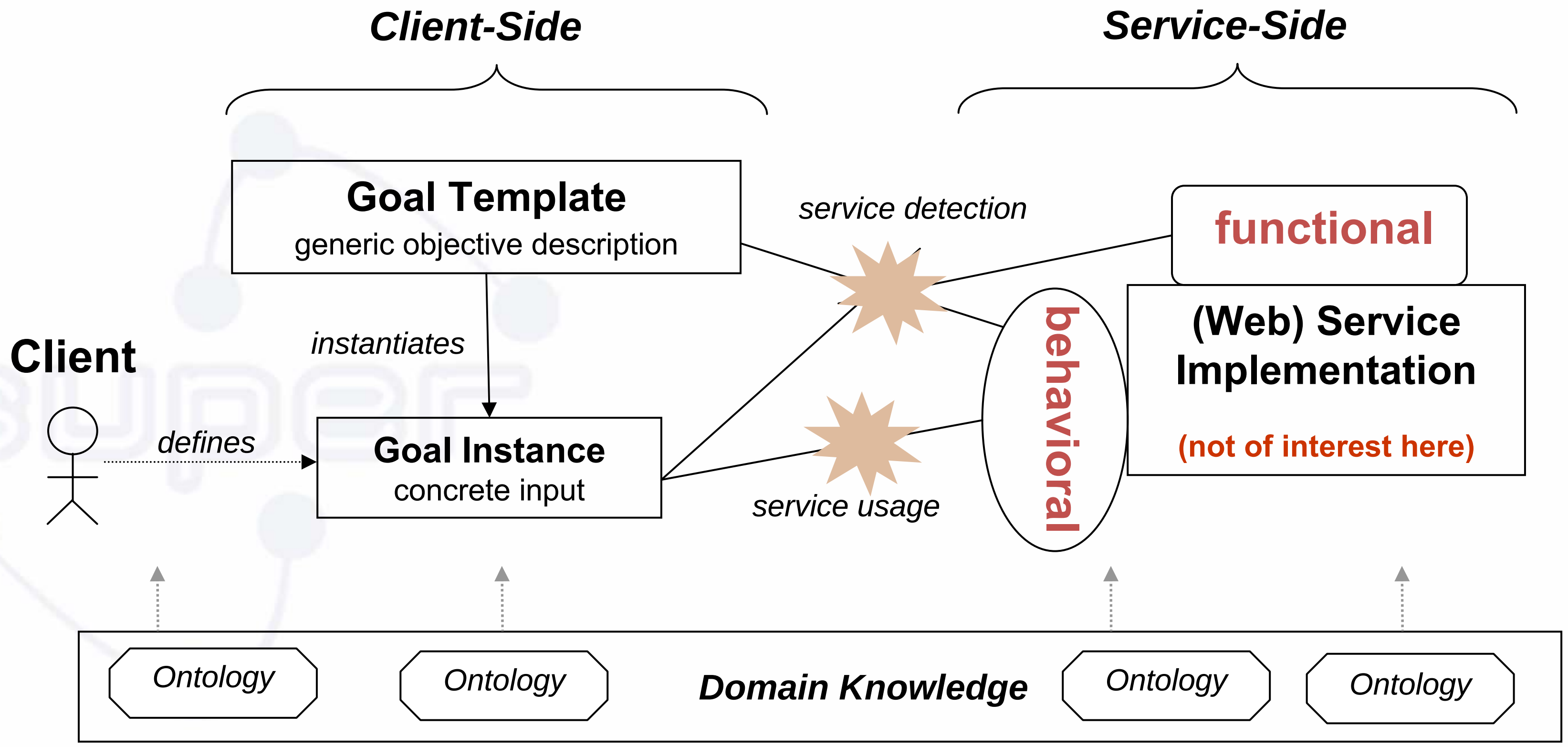
- behavior interfaces of Web services and clients for "peer-2-peer" interaction

- Choreography and Orchestration as sub-concepts of Service Interface with common description language

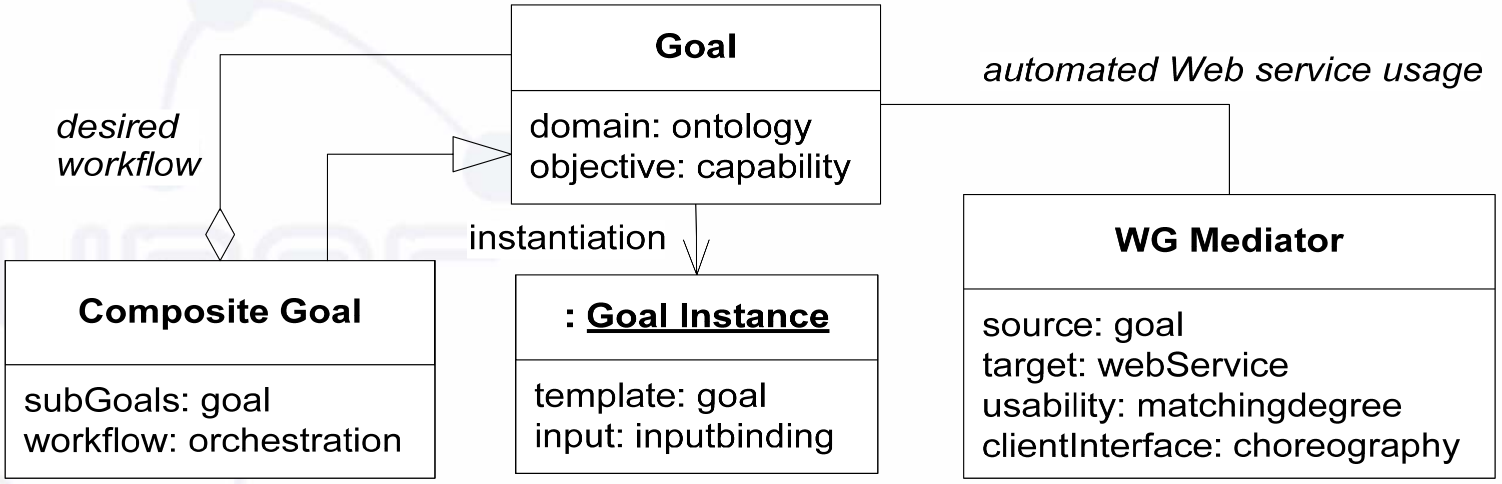
- Web Service Interface Description aspects:
  1. represent the **dynamics** of information interchange during service consumption and interaction
  2. support **ontologies** as the underlying data model
  3. appropriate **communication technology** for information interchange
  4. sound **formal model / semantics** of service interface specifications in order to allow advanced reasoning on them

=> "ontologized Abstract State Machines"

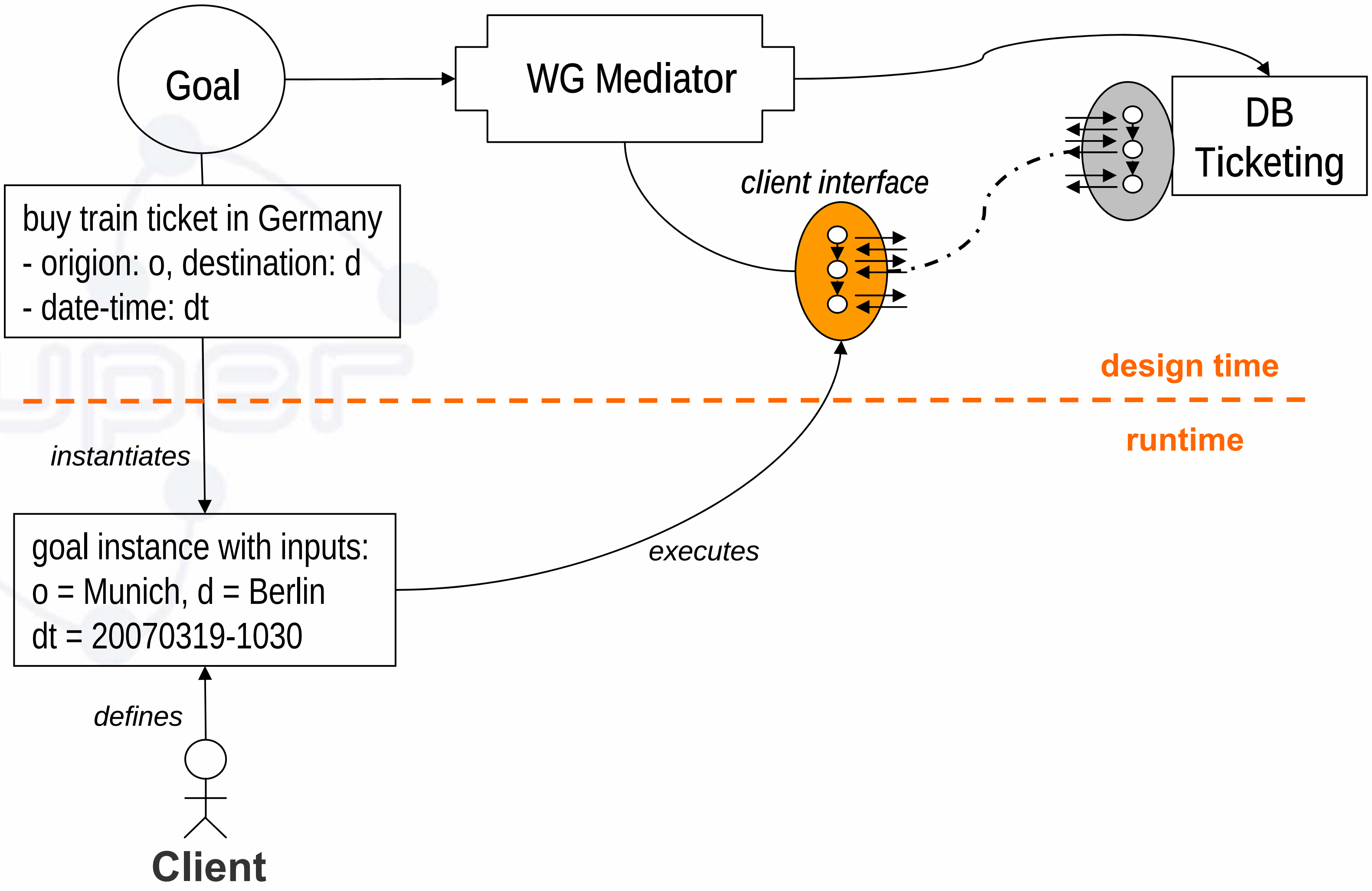
WSMO Goals


Client-Side
Service-Side
Goal Template
generic objective description
service detection
functional
instantiates
Client
(Web) Service Implementation
behavioral
defines
Goal Instance
concrete input
(not of interest here)
service usage
Ontology
Ontology
Domain Knowledge
Ontology
Ontology

Goal Model


Goal
domain: ontology
objective: capability
automated Web service usage
desired
workflow
instantiation
Composite Goal
subGoals: goal
workflow: orchestration
: Goal Instance
template: goal
input: inputbinding
WG Mediator
source: goal
target: webService
usability: matchingdegree
clientInterface: choreography

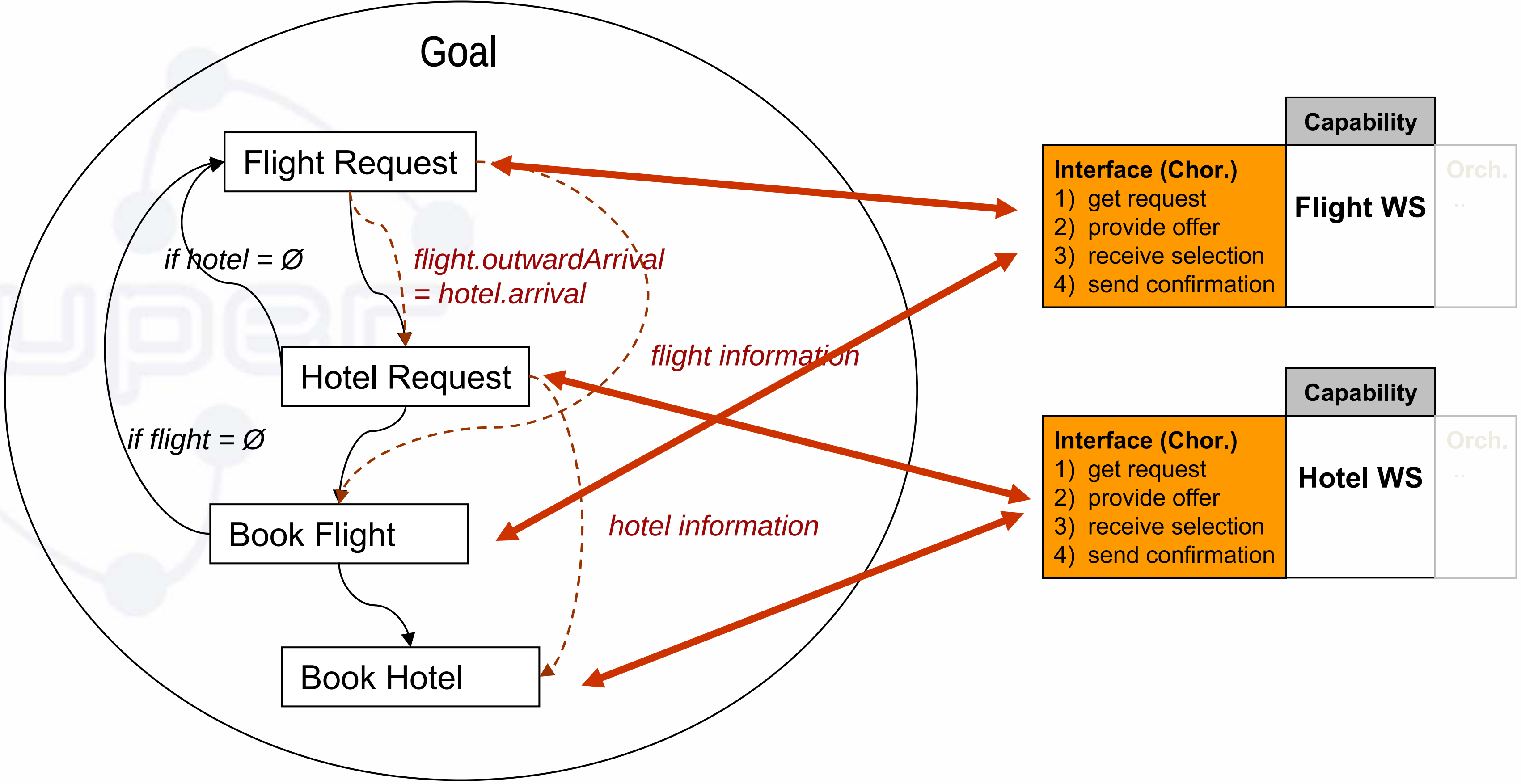
Basic Goal



Goal
WG Mediator
DB
Ticketing
client interface
buy train ticket in Germany
- origion: o, destination: d
- date-time: dt
design time
runtime
instantiates
goal instance with inputs:
o = Munich, d = Berlin
dt = 20070319-1030
executes
defines
Client

Composite Goal

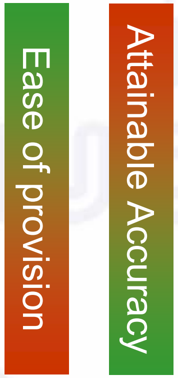
Flight-hotel booking with desired workflow



Web Service Discovery

detect directly usable Web services out of available ones

- Discovery Techniques (functional as primary focus)



Ease of provision

Attainable Accuracy

Key Word Matching

match natural language key words in resource descriptions

Controlled Vocabulary

ontology-based key word matching

Semantic Matchmaking

… what Semantic Web Services aim at

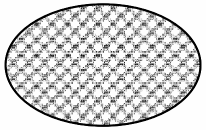
- Selection: choose most appropriate Web Service with respect to:
  - Quality of Service (security, robustness, availability)
  - context (regional, business / social communities)
  - preferences and policies
  - usage costs
  - …

Semantic Matchmaking

▨ = G ▧ = WS

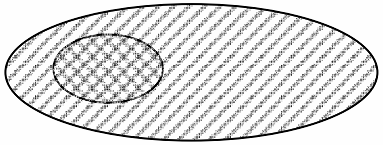
Exact Match:

$G, WS, O, M \models \forall x.\ (G(x) <=> WS(x)\ )$



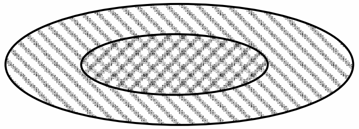
PlugIn Match:

$G, WS, O, M \models \forall x.\ (G(x) => WS(x)\ )$



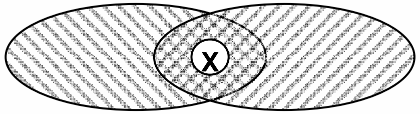
Subsumption Match:

$G, WS, O, M \models \forall x.\ (G(x) <= WS(x)\ )$



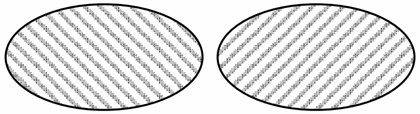
Intersection Match:

$G, WS, O, M \models \exists x.\ (G(x) \wedge WS(x)\ )$



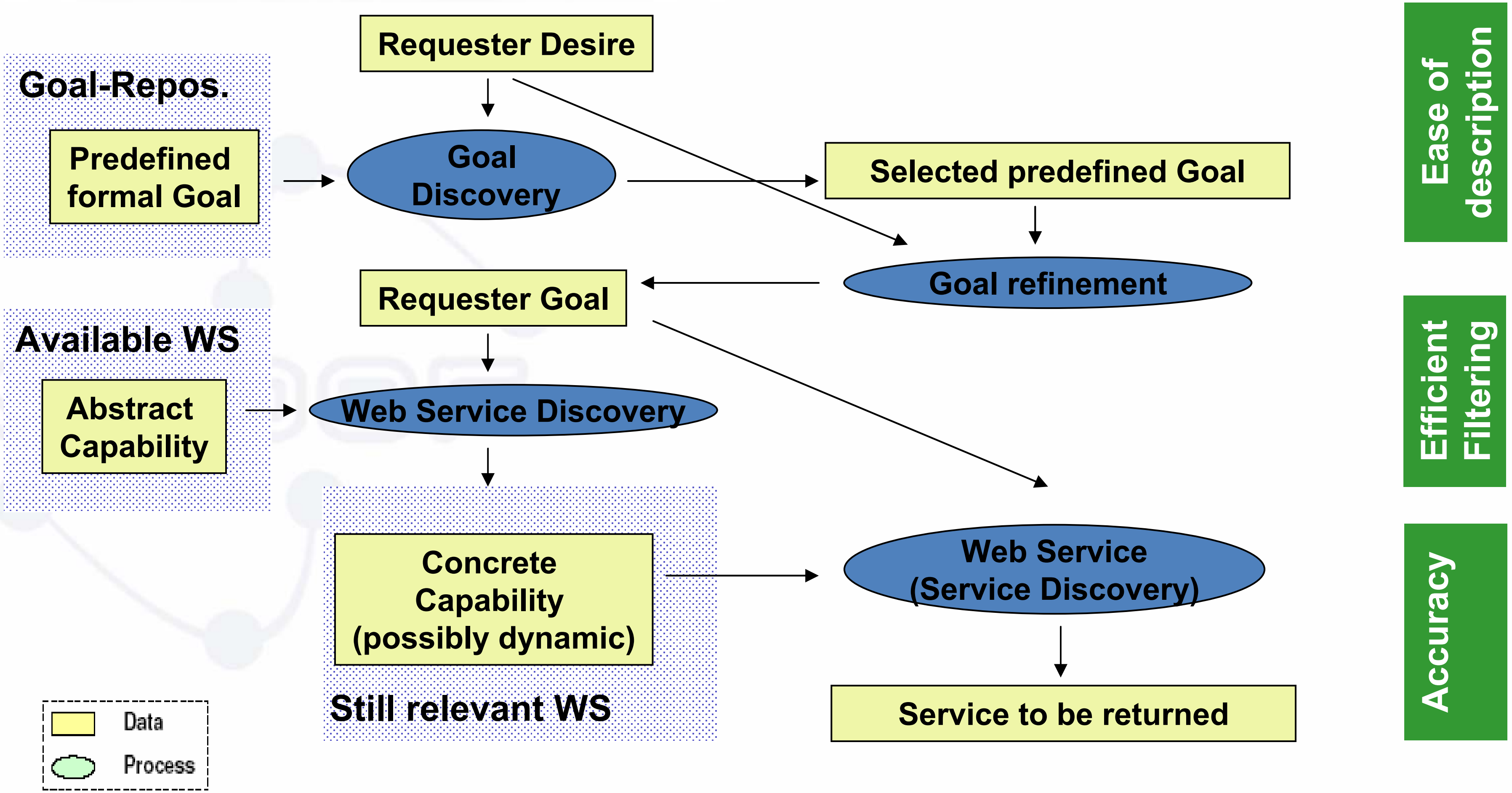
Non Match:

$G, WS, O, M \models \neg\exists x.\ (G(x) \wedge WS(x)\ )$



Keller, U.; Lara, R.; Polleres, A. (Eds): *WSMO Web Service Discovery*. WSML Working Draft D5.1, 12 Nov 2004.

WSMO Discovery Process



Web Service Composition

combine several Web services for solving a request

- composition of Web services is needed

  if no directly usable Web service exists …

  a) a WS can satisfy goal, but goal cannot invoke WS

  b) several WS need to be combined to achieve goal

- composition techniques:

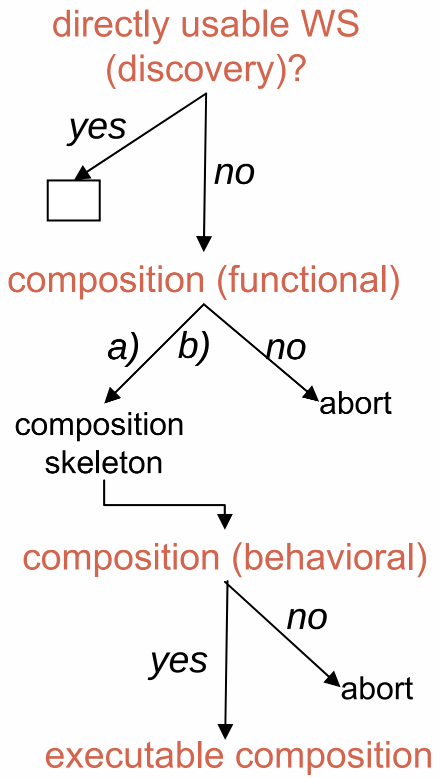
  functional = composition wrt ***functionalities***

  behavioral = composition wrt ***behavioral interfaces***

  ⇒ ***need to be integrated:***

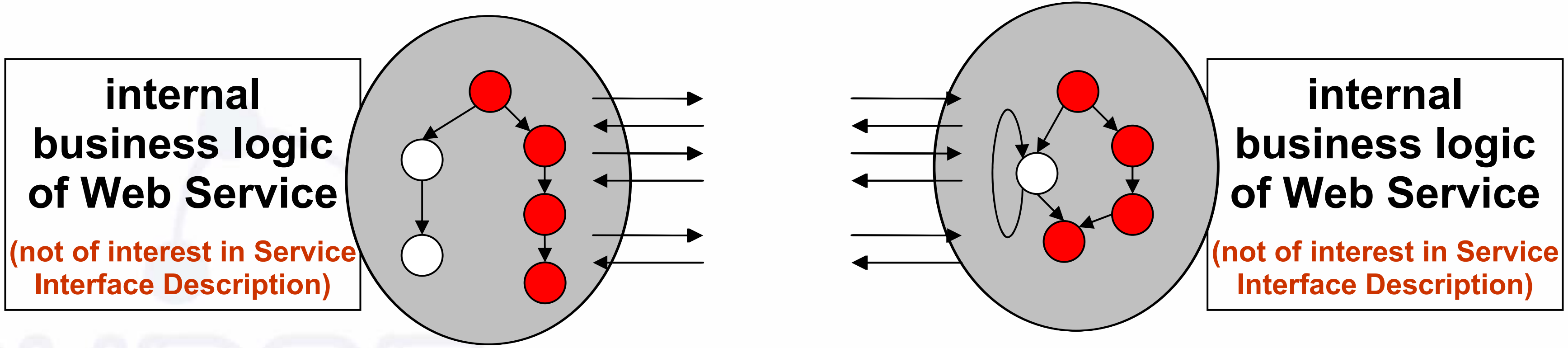
  1. skeleton by functional composition
  2. refinement + executable code by behavioral composition

Procedure:



Choreography Discovery

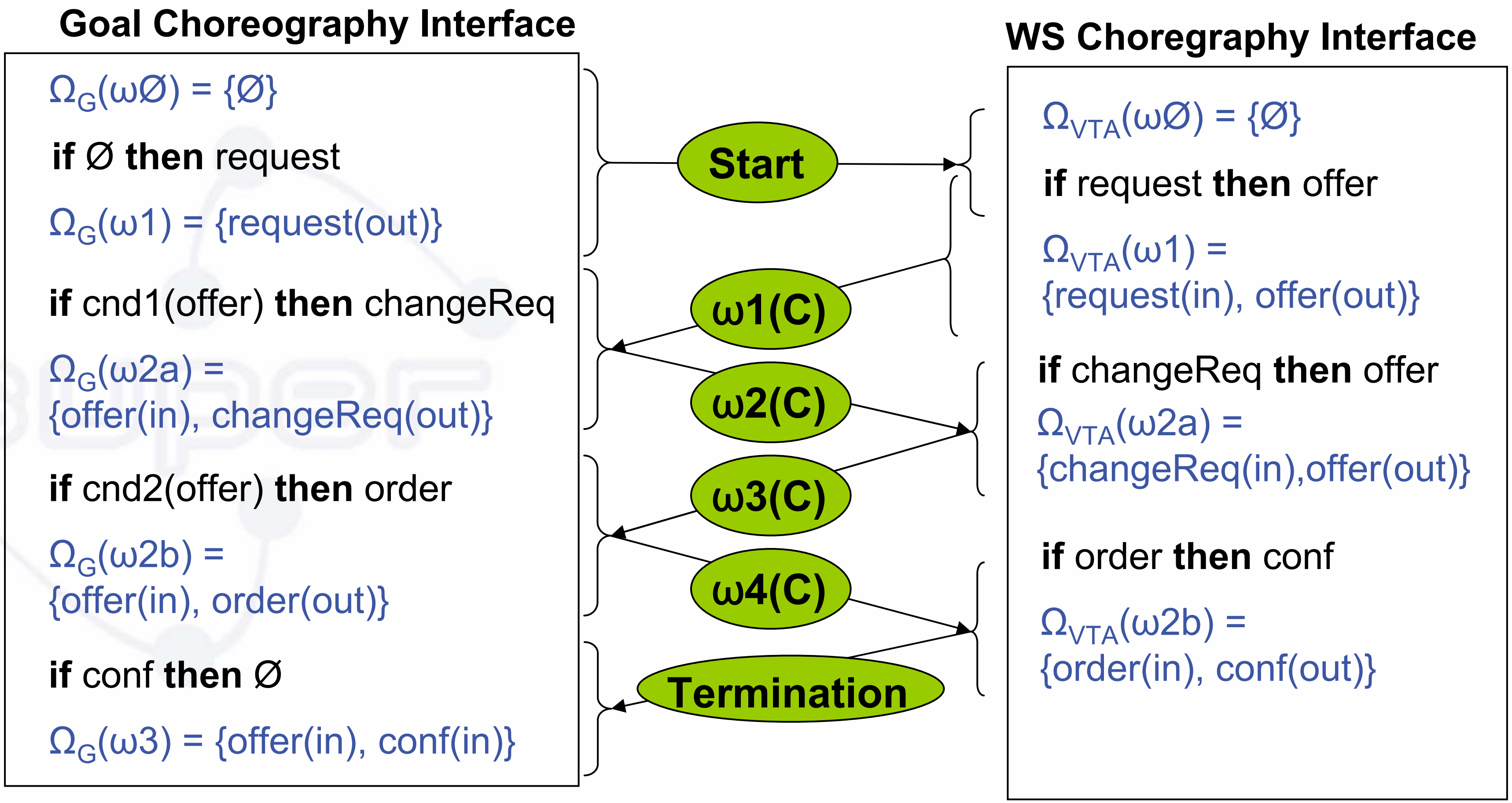
determine behavioral compatibility


internal business logic of Web Service
(not of interest in Service Interface Description)
internal business logic of Web Service
(not of interest in Service Interface Description)

a valid choreography exists if:

1) **Signature Compatibility**
   - homogeneous ontologies
   - compatible in- and outputs
2) **Behavior Compatibility**
   - start state for interaction
   - a termination state can be reached without any additional input

Behavior Compatibility Example



Goal Choreography Interface

$\Omega_G(\omega\emptyset) = \{\emptyset\}$

if ∅ **then** request

$\Omega_G(\omega 1) = \{request(out)\}$

if cnd1(offer) **then** changeReq

$\Omega_G(\omega 2a) = \{offer(in), changeReq(out)\}$

if cnd2(offer) **then** order

$\Omega_G(\omega 2b) = \{offer(in), order(out)\}$

if conf **then** ∅

$\Omega_G(\omega 3) = \{offer(in), conf(in)\}$

WS Choregraphy Interface

$\Omega_{VTA}(\omega\emptyset) = \{\emptyset\}$

if request **then** offer

$\Omega_{VTA}(\omega 1) = \{request(in), offer(out)\}$

if changeReq **then** offer

$\Omega_{VTA}(\omega 2a) = \{changeReq(in),offer(out)\}$

if order **then** conf

$\Omega_{VTA}(\omega 2b) = \{order(in), conf(out)\}$

valid choreography existent

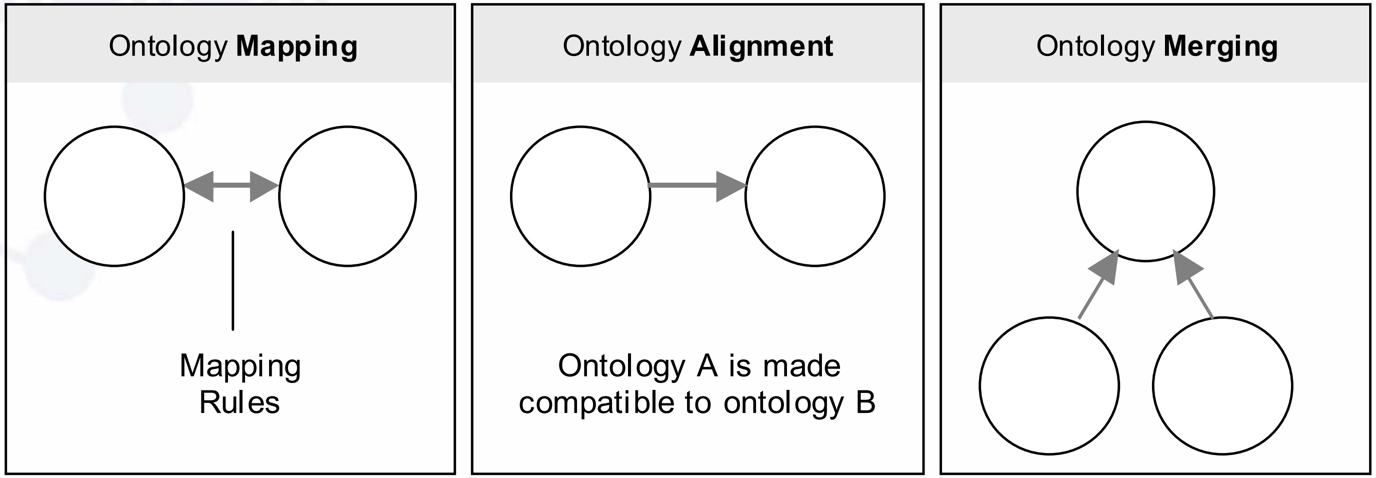
Mediation

- **Heterogeneity …**
  - mismatches on structural / semantic / conceptual / level
  - occur between different components that shall interoperate
  - especially in distributed & open environments like the Internet
- **Concept of Mediation** (Wiederhold, 94):
  - ***Mediators*** as components that resolve mismatches
  - declarative approach:
    - semantic description of resources
    - ‘intelligent’ mechanisms resolve mismatches independent of content
  - mediation cannot be fully automated (integration decision)
- **Levels of Mediation within Semantic Web Services**:

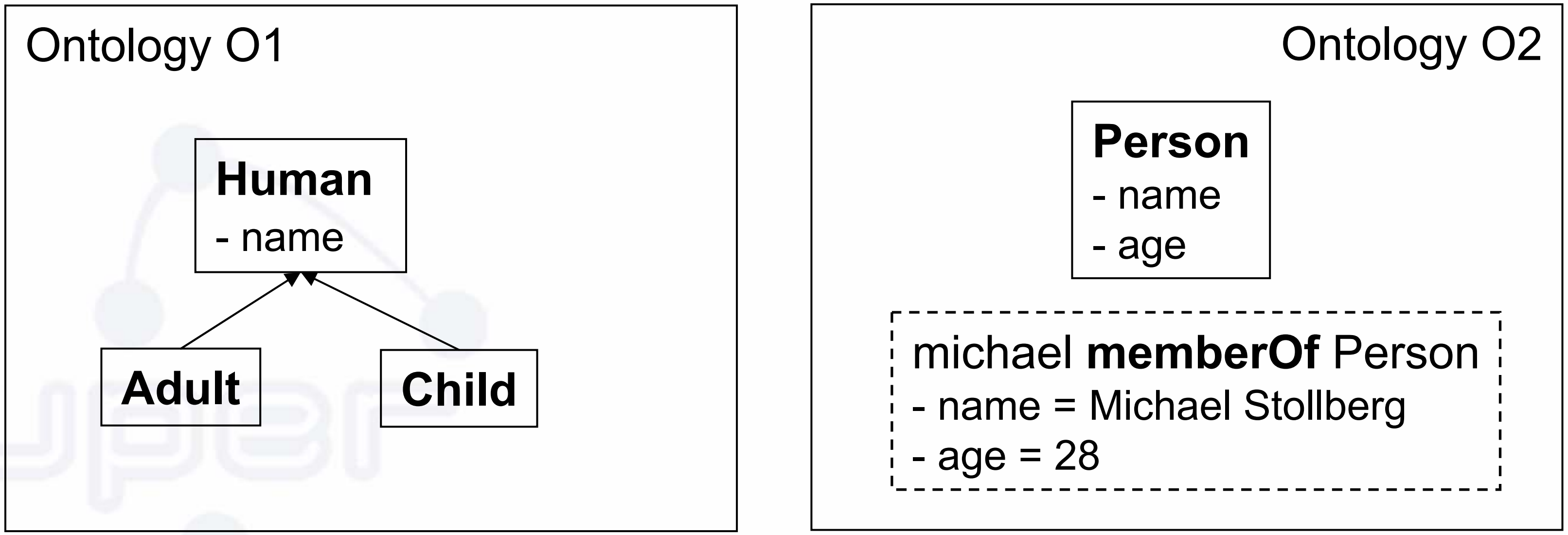
1. Representation Level:	heterogeneous Languages & Protocols
2. Data Level:	heterogeneous Data Sources
3. Functional Level:	heterogeneous Functionalities
4. Process Level:	heterogeneous Communication Processes

Data Mediation Techniques

- resolve semantic mismatches between terminologies
- realized by ontology integration
  - mappings between heterogeneous ontologies (design time)
  - data transformation (runtime)



Mapping Language Example



Ontology O1

Human
- name

Adult

Child

Ontology O2

Person
- name
- age

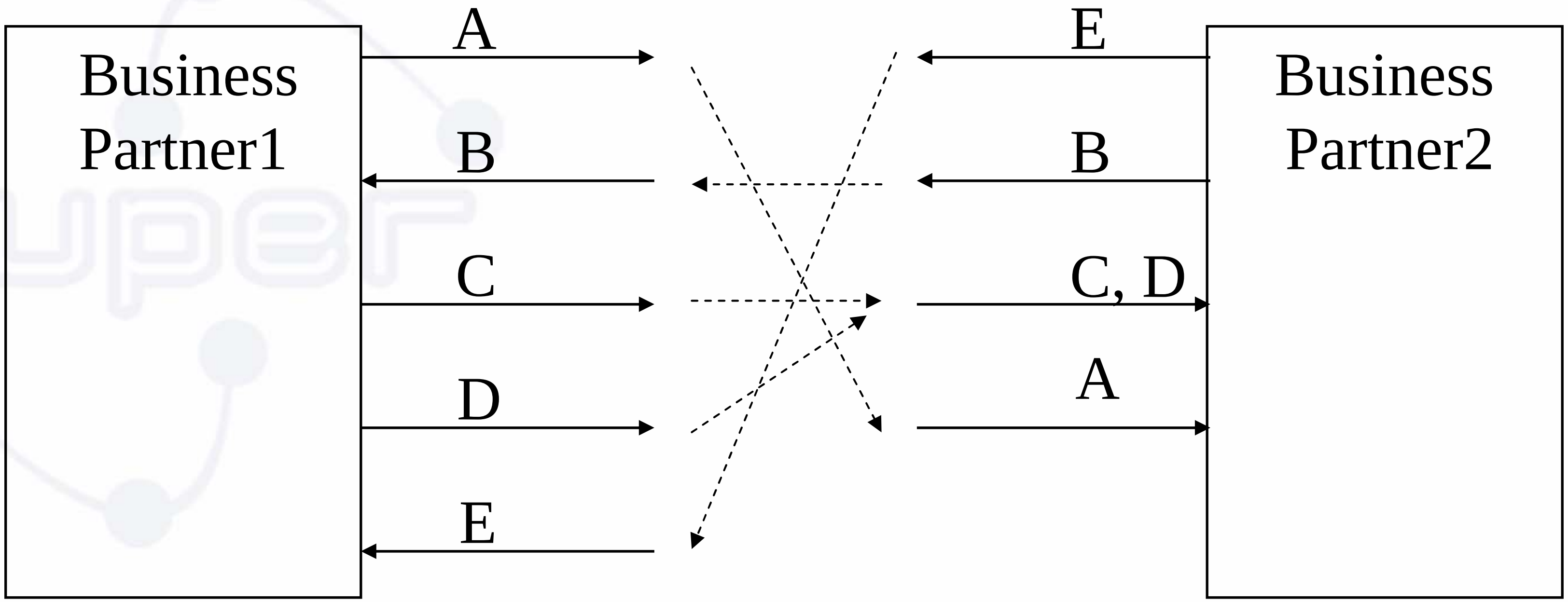
michael **memberOf** Person
- name = Michael Stollberg
- age = 28

classMapping(**unidirectional** o2:Person o1.Adult
attributeValueCondition(o2.Person.age **>=** 18))

this allows to transform the instance ‘michael’ of concept person in ontology O2 into a valid instance of concept ‘adult’ in ontology O1

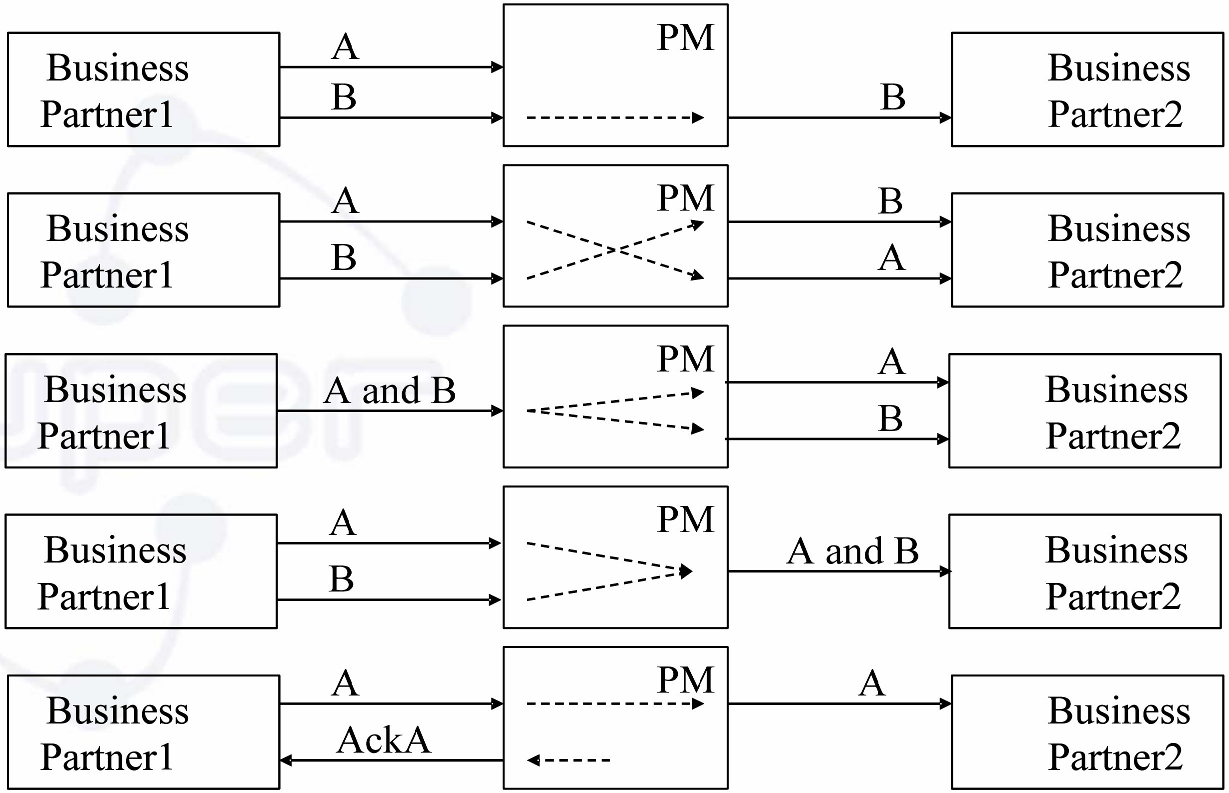
Process Level Mediation

- not a priori compatible behavior interfaces for communication & information interchange => **behavioral incompatibility**



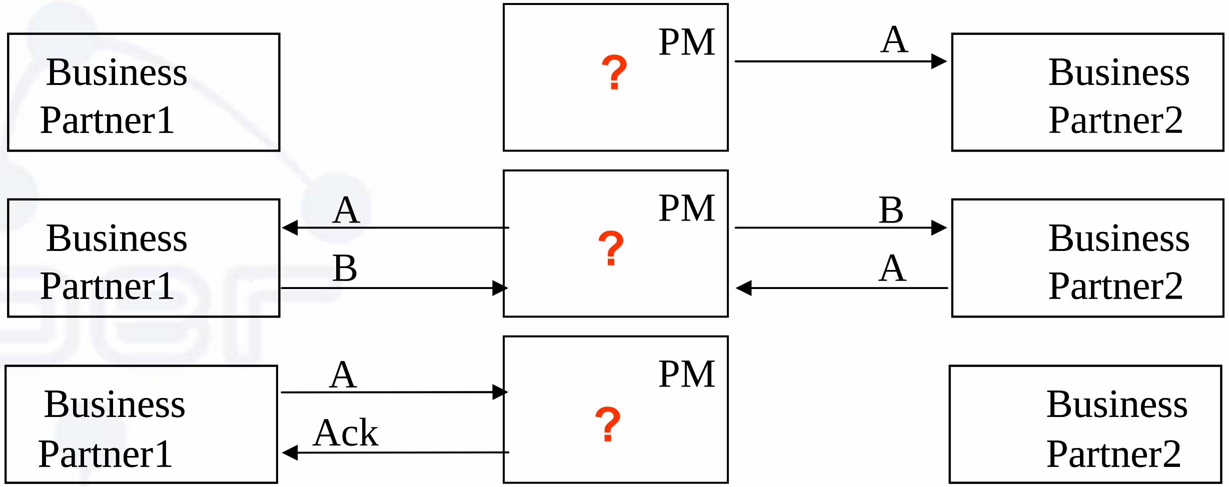
- partially resolvable by **process mediation patterns**

Patterns for Resolvable Mismatches

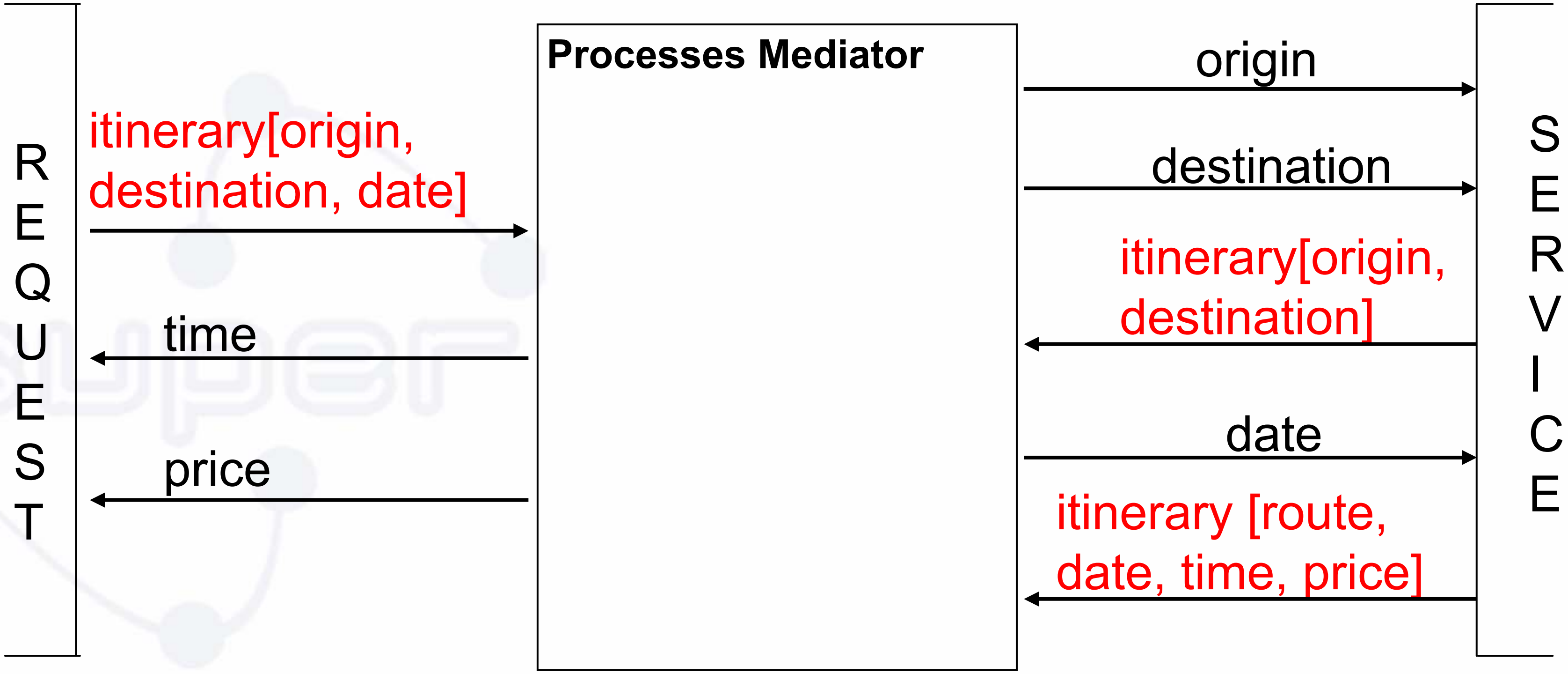


can resolve about 80 % of process level mismatches

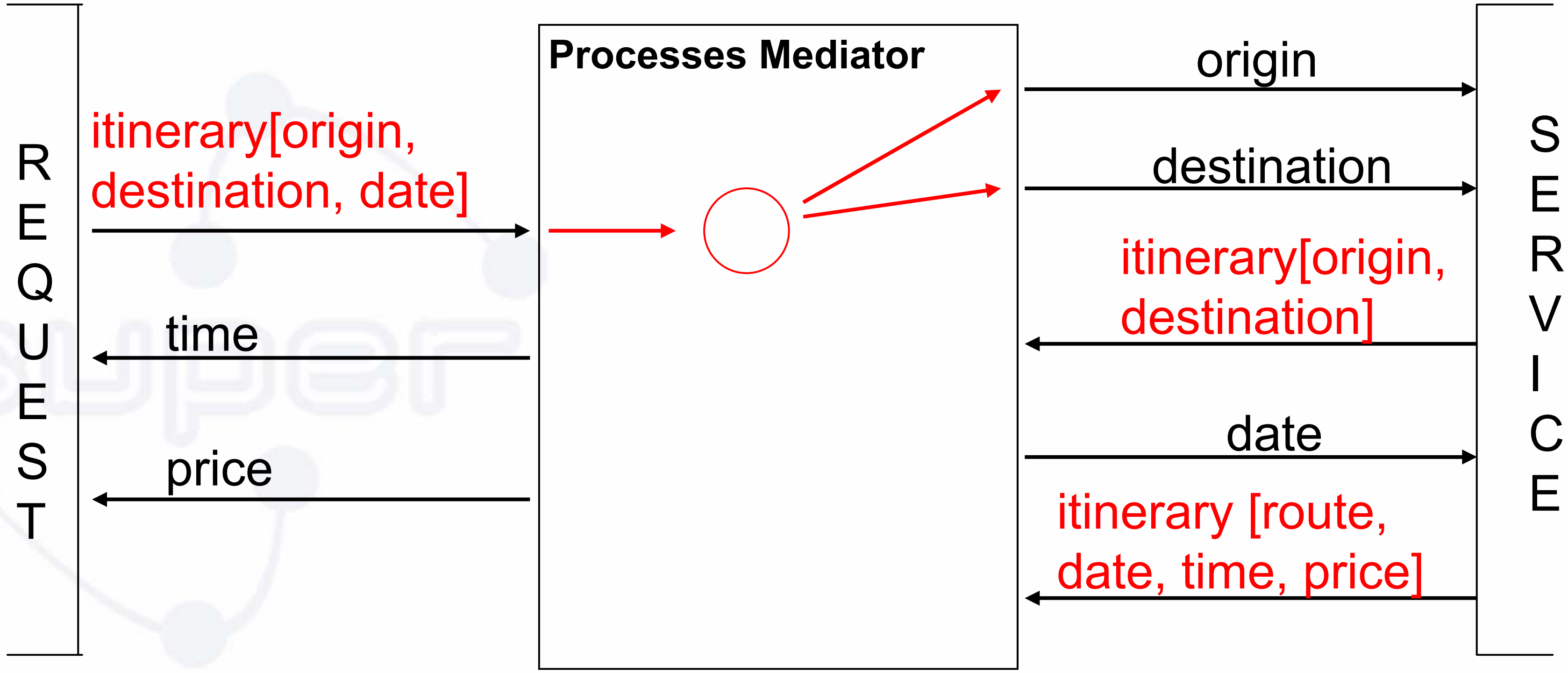
Unresolvable Process Mismatches



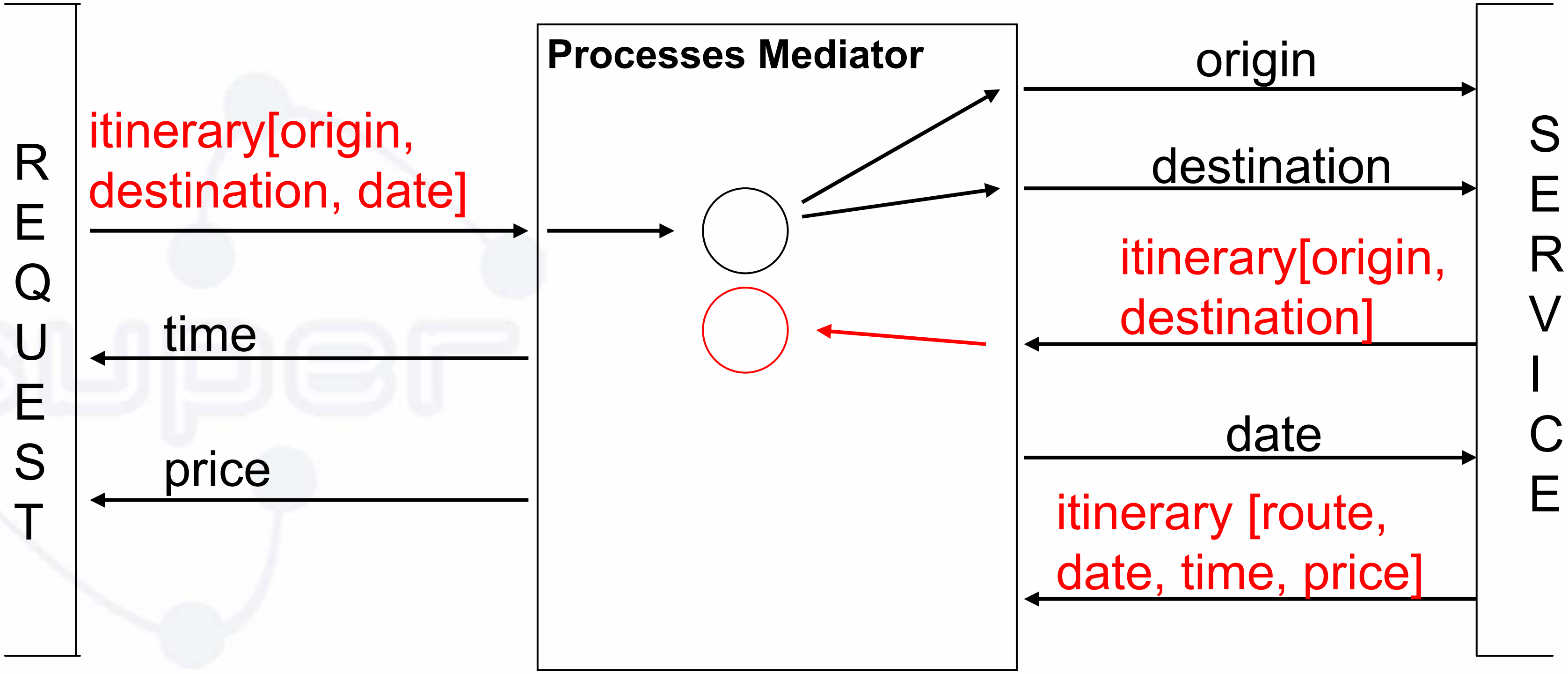
Process Mediation Example


REQUEST
itinerary[origin, destination, date]
time
price
Processes Mediator
origin
destination
itinerary[origin, destination]
date
itinerary [route, date, time, price]
SERVICE

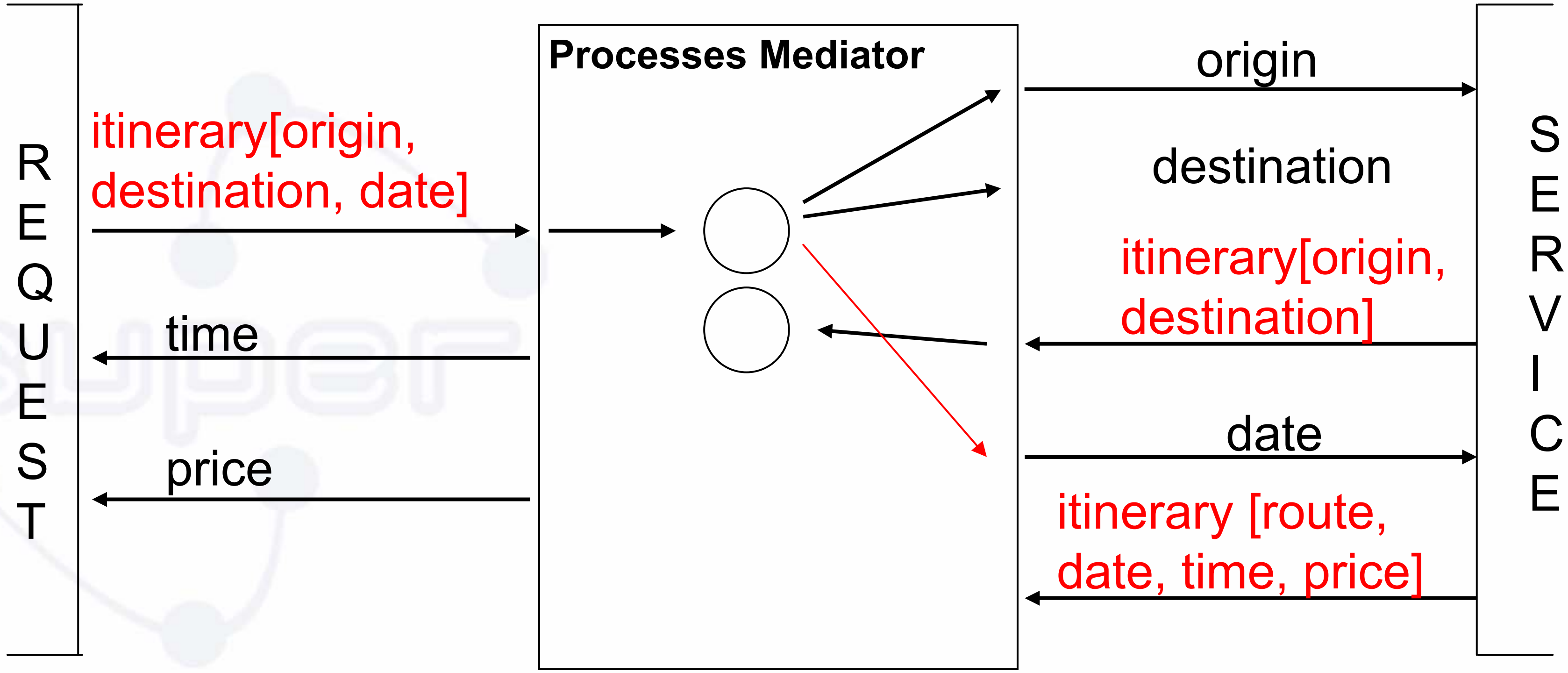
Process Mediation Example


REQUEST
itinerary[origin, destination, date]
time
price
Processes Mediator
origin
destination
itinerary[origin, destination]
date
itinerary [route, date, time, price]
SERVICE

Process Mediation Example

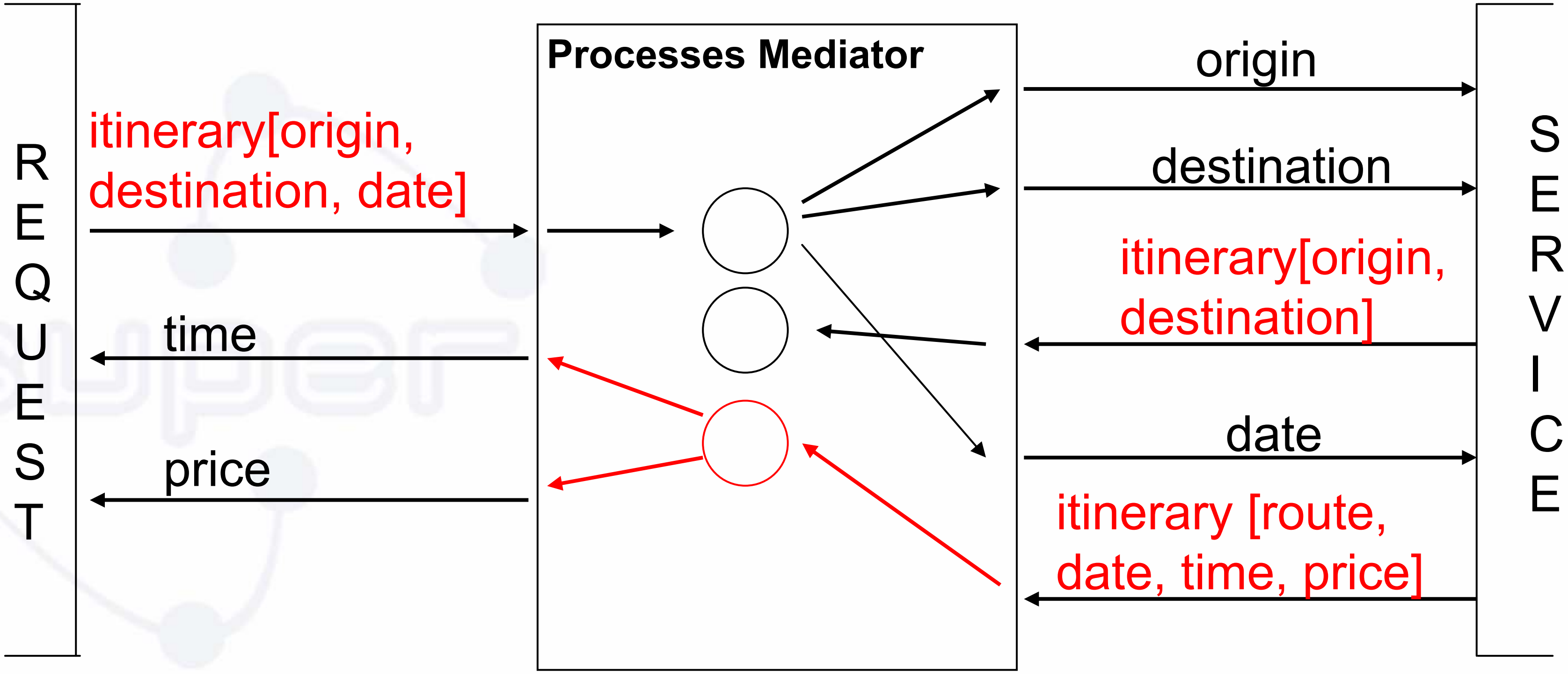

REQUEST
itinerary[origin, destination, date]
time
price
Processes Mediator
origin
destination
itinerary[origin, destination]
date
itinerary [route, date, time, price]
SERVICE

Process Mediation Example

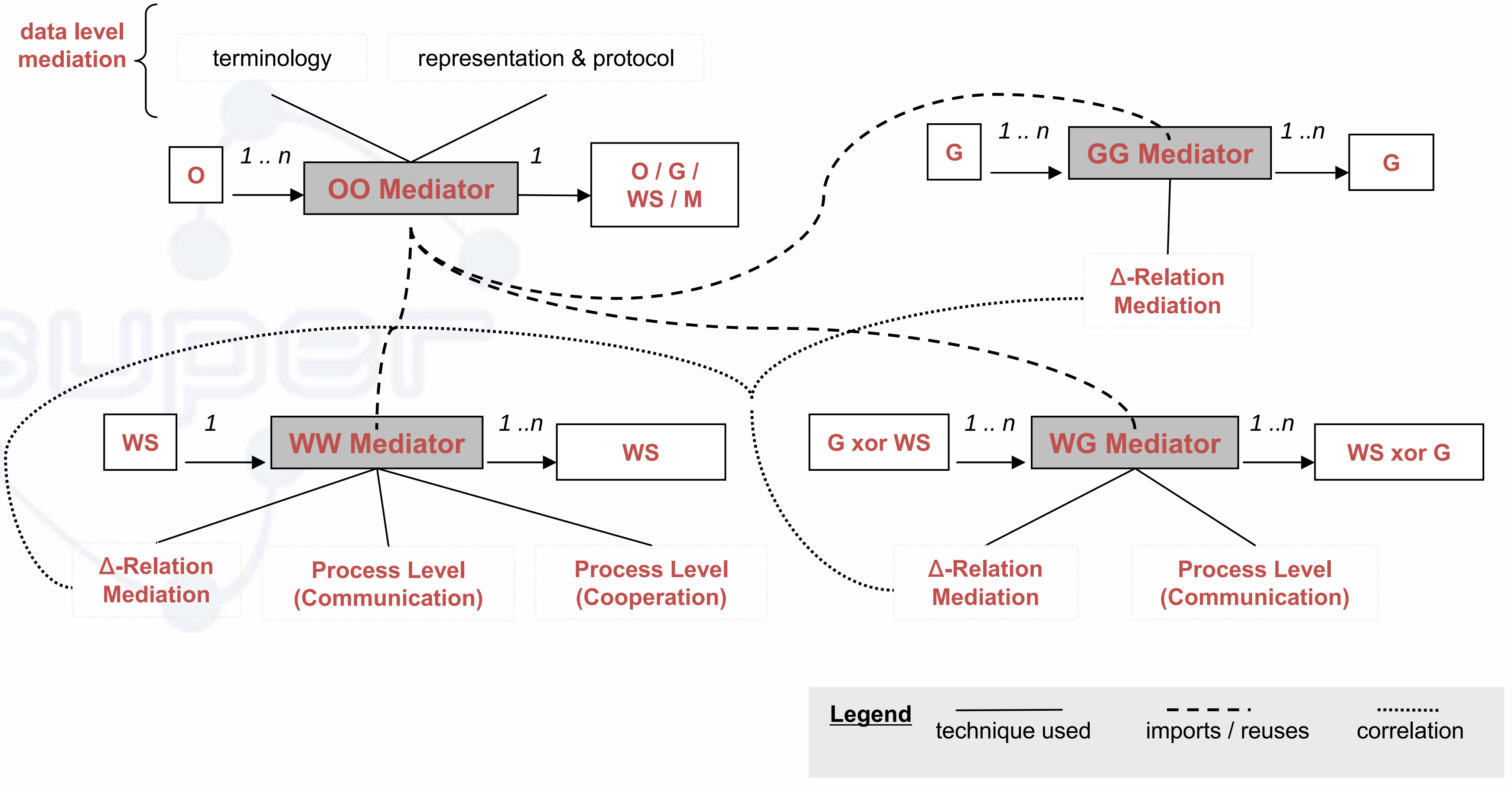


REQUEST
itinerary[origin, destination, date]
time
price
Processes Mediator
origin
destination
itinerary[origin, destination]
date
itinerary [route, date, time, price]
SERVICE

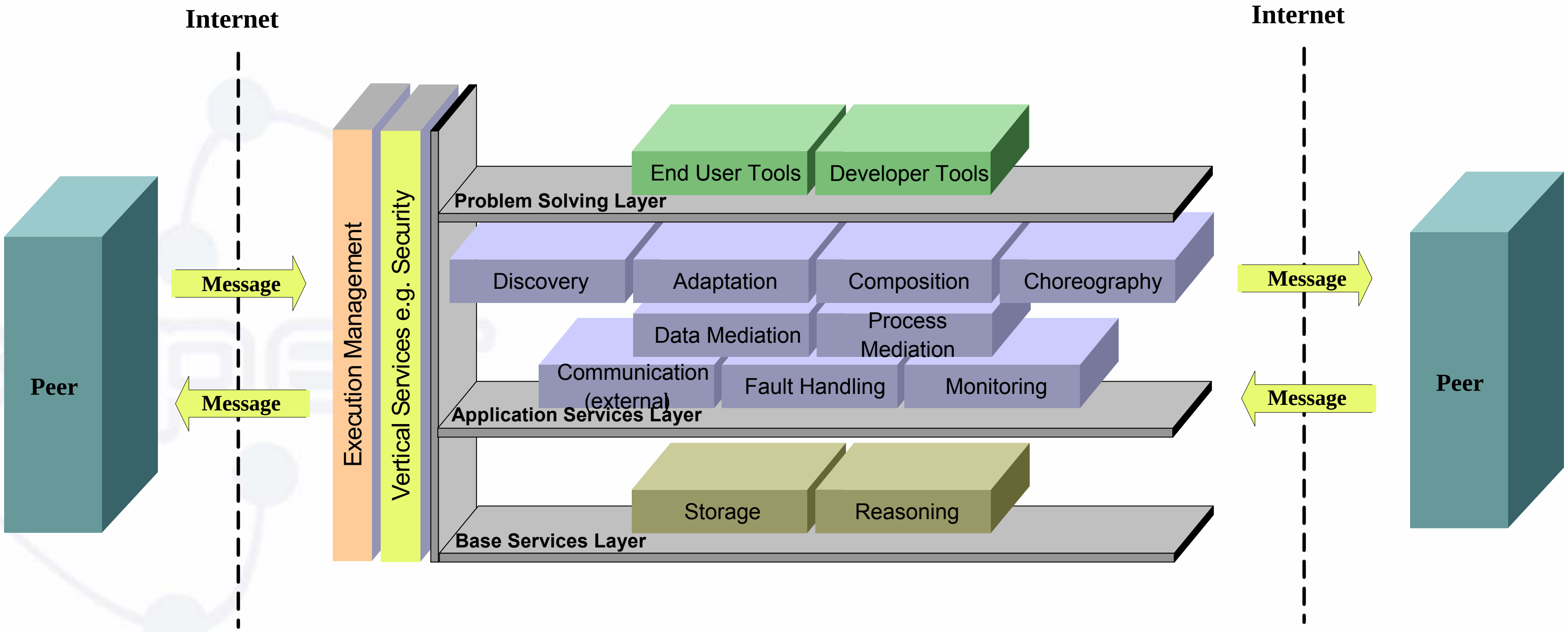
Process Mediation Example


REQUEST
itinerary[origin, destination, date]
time
price
Processes Mediator
origin
destination
itinerary[origin, destination]
date
itinerary [route, date, time, price]
SERVICE

WSMO Mediators Overview



The Web Service Execution Environment WSMX (WSMO Reference Implementation)



Open source code base at SourceForge: http://sourceforge.net/projects/wsmx/

Other WSMO Tools

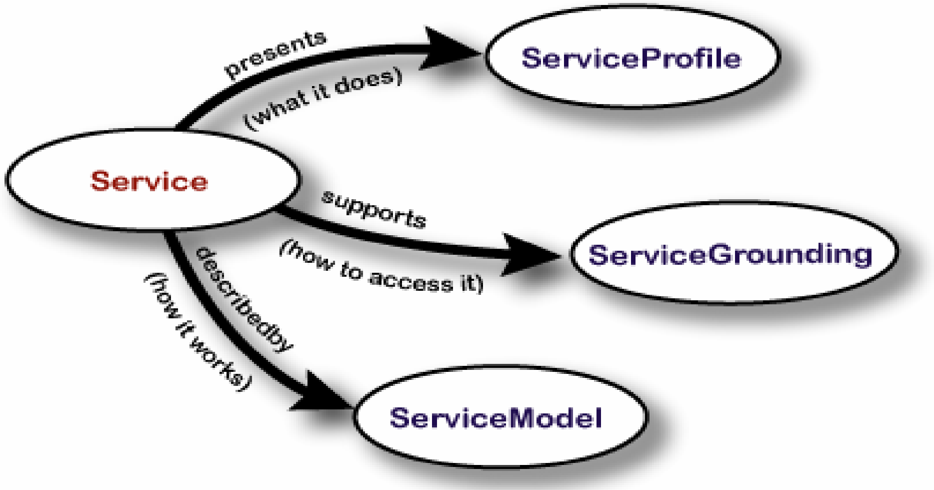
www.wsmo.org/tools

- WSML (Specification Language) www.wsmo.org/wsml
  - conceptual language for WSMO
  - ontology language with several variants
- WSMO Editors:
  - WSML editors + validation
  - WSMO Studio
  - WSMO Visualizer
- Ontology Technology:
  - WSML Reasoner (for DL and LP)
  - Ontology Management Suite
  - Data Mediator (incl. Abstract Mapping Language)

all: Eclipse plugins & open source (LGPL licence)

OWL-S

- Conceptual Model
  - A set of ontologies used to describe different aspects SWS
- Language: OWL
- Some OWL-S drawbacks
  - OWL not sufficiently expressive for all aspects of a service
    - more expressive languages have been syntactically integrated: SWRL, KIF, DRS, and PDDL – how do these languages interoperate?
  - Inherits some of the drawbacks of OWL (e.g. lack of proper layering, improper use of OWL for describing and reasoning about processes)
  - No explicit support for Mediation in the language

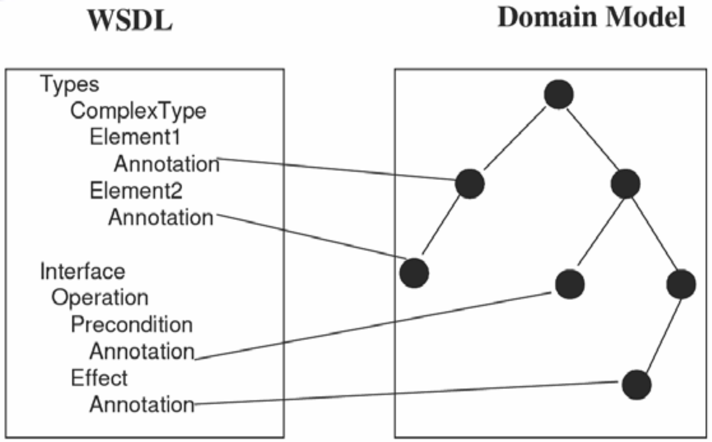


Semantic Web Services Framework (SWSF)

- Two major components: an ontology and a language used to axiomatize it
- Semantic Web Services Ontology (SWSO) – an extension of OWL-S conceptual model, e.g. a rich behavioural process model based on PSL
  - FLOWS – First-Order Logic Ontology for Web Services
  - ROWS - Rule Ontology for Web Services
- The Semantic Web Services Language (SWSL)
  - SWSL-FOL - based on First Order Logic; includes features from HiLog and F-Logic
  - SWSL-Rules - a logic programming language; includes features from Courteous logic programs, HiLog, and F-Logic
- Some SWSF drawbacks
  - unclear how all the paradigms part of this approach work together
  - first-order logic ontology for Web services, but not a Web language

Web Service Semantics - WSDL-S

- A mechanism to augment WSDL descriptions with semantics
  - a set of annotations can be created to semantically describe the inputs, outputs and operations of a Web service.
  - keeps the semantic model outside WSDL, making the approach agnostic to any ontology representation lang



- WSDL-S doesn't provide a conceptual model and language for SWS
  - a bottom up approach to SWS (annotating existing standards with metadata)
- Could be used as a grounding mechanism for SWS

SWS Conclusions

- Semantic Web Services
  - Initial technical solutions existing
  - High potential in BPM, B2B, EAI, eCommerce, etc.
- The WSMO Approach to SWS looks promising
  - A unifying approach for semantic SOA
  - Top-level entities: Ontologies, Web Services, Goals, Mediators
- Standardization activities are emerging in this area
  - OWL-S, SWSF, WSDL-S, WSMO – submitted to W3C
  - OASIS SEE technical committee formed (based on WSMX)
  - W3C SAWSDL Working Group formed; closed to recommendation
- Future Aspects:
  - Apply & extend towards usage scenarios
  - Educate Ontology & SWS Engineers

Semantics Utilised for Process Management within and between Enterprises

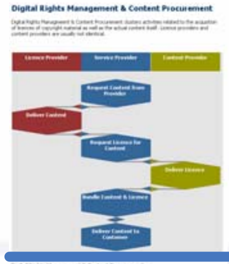
Integration of SWS into BPM

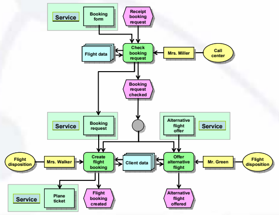
[The SUPER Approach]

Modelling Stack

- Making sense of a domain\problem
- Communication tool
- What is it all about?

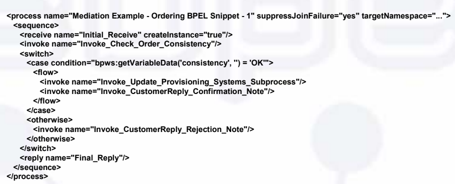
- Solution maps
- Mind maps
- Ad-hoc modelling techniques
- ...


Digital Rights Management & Content Procurement
Licence Provider
Service Provider
Content Provider
Request Content from Provider
Deliver Content
Request Licence for Content
Deliver Licence
Handle Content & Licence
Deliver Content to Customer



- Visualizing\specifying business process
- Focus: Business Problem
- Who does what, when, how and why?
- Usually multiple layers

- Business Scenario Maps
- Event-driven process chains
- Flowchart techniques
- BPMN
- ...



```
<process name="Mediation Example - Ordering BPEL Snippet - 1" suppressJoinFailure="yes" targetNamespace="...">
  <sequence>
    <receive name="Initial_Receive" createInstance="true"/>
    <invoke name="Invoke_Check_Order_Consistency"/>
    <switch>
      <case condition="bpws:getVariableData('consistency', '') = 'OK'">
        <flow>
          <invoke name="Invoke_Update_Provisioning_Systems_Subprocess"/>
          <invoke name="Invoke_CustomerReply_Confirmation_Note"/>
        </flow>
      </case>
      <otherwise>
        <invoke name="Invoke_CustomerReply_Rejection_Note"/>
      </otherwise>
    </switch>
    <reply name="Final_Reply"/>
  </sequence>
</process>
```

- Process execution specification
- Formal, clearly specified grammar
- Focus: Implementation
- Which component is called when, how, by whom with which data?

- BPEL
- ...

- Web service encapsulation
- Focus: Implementation
- Which components can and should be exposed how as services?

- WS*
- ...

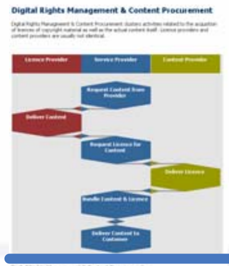
- Implementation of components

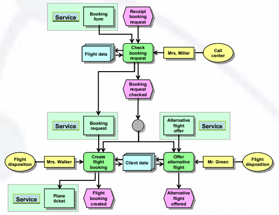
- Programming languages
- ...

Modelling Stack

- Making sense of a domain\problem
- Communication tool
- What is it all about?

- Solution maps
- Mind maps
- Ad-hoc modelling techniques
- ...


Digital Rights Management & Content Procurement
Licence Provider
Service Provider
Content Provider
Request Content from Provider
Deliver Content
Request Licence for Content
Deliver Licence
Bundle Content & Licence
Deliver Content to Customer



- Visualizing\specifying business process
- Focus: Business Problem
- Who does what, when, how and why?
- Usually multiple layers

- Business Scenario Maps
- Event-driven process chains
- Flowchart techniques
- BPMN
- ...

```
<process name="Mediation Example - Ordering BPEL Snippet - 1" suppressJoinFailure="yes" targetNamespace="...">
  <sequence>
    <receive name="Initial_Receive" createInstance="true"/>
    <invoke name="Invoke_Check_Order_Consistency"/>
    <switch>
      <case condition="bpws:getVariableData('consistency', '') = 'OK'">
        <flow>
          <invoke name="Invoke_Update_Provisioning_Systems_Subprocess"/>
          <invoke name="Invoke_CustomerReply_Confirmation_Note"/>
        </flow>
      </case>
      <otherwise>
        <invoke name="Invoke_CustomerReply_Rejection_Note"/>
      </otherwise>
    </switch>
    <reply name="Final_Reply"/>
  </sequence>
</process>
```

- Process execution specification
- Formal, clearly specified grammar
- Focus: Implementation
- Which component is called when, how, by whom with which data?

- BPEL
- ...

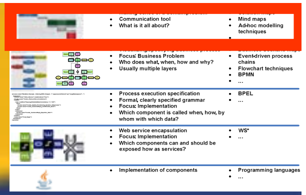
- Web service encapsulation
- Focus: Implementation
- Which components can and should be exposed how as services?

- WS*
- ...

- Implementation of components

- Programming languages
- ...

Telecommunications Solution Map



- Communication tool
- What is it all about?

- Mind maps
- Ad-hoc modelling techniques
-

- Focus: Business Problem
- Who does what, when, how and why?
- Usually multiple layers

- Event-driven process chains
- Flowchart techniques
- BPMN
- …

- Process execution specification
- Formal, clearly specified grammar
- Focus: Implementation
- Which component is called when, how, by whom with which data?

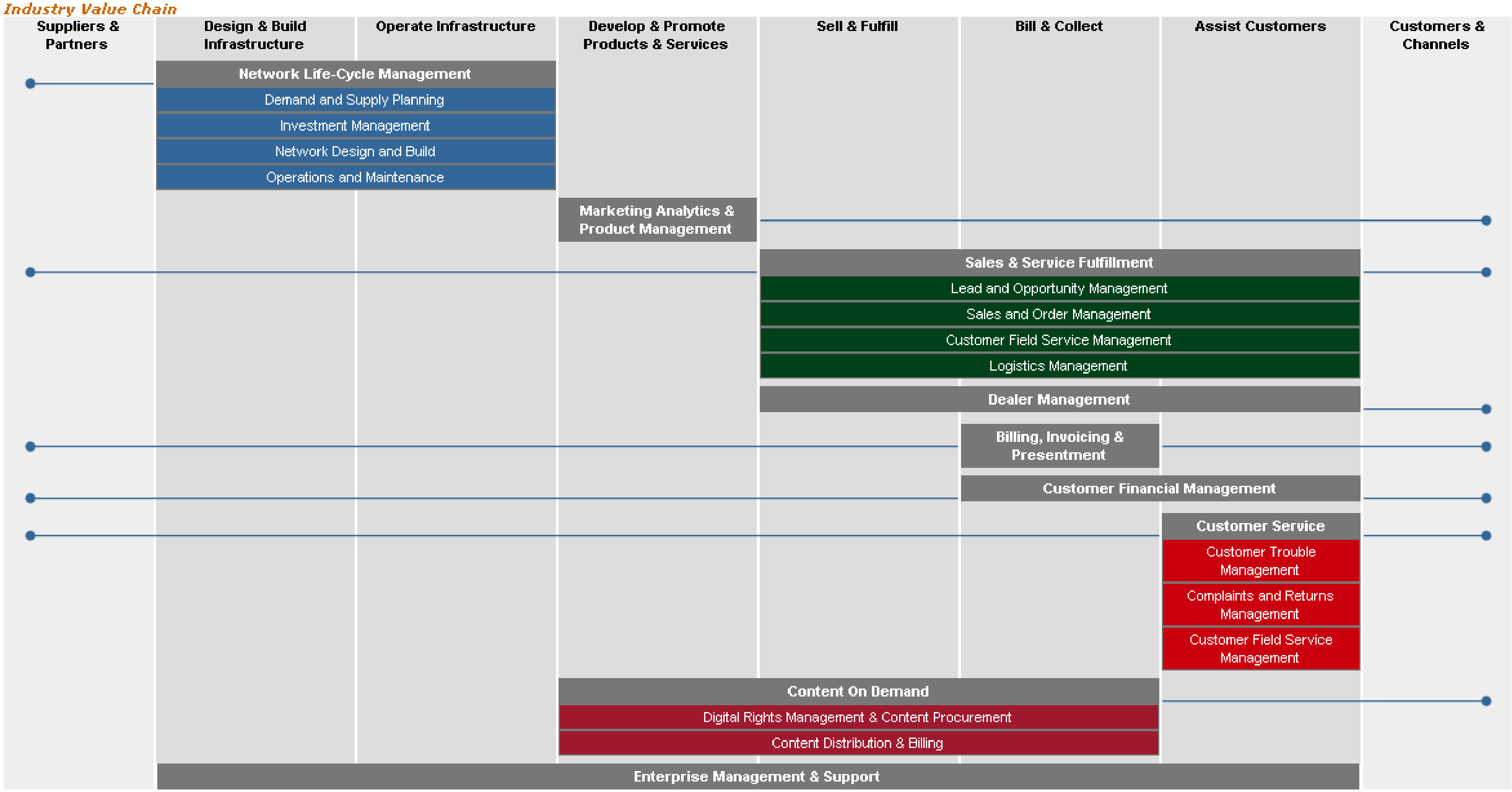
- BPEL
- …

- Web service encapsulation
- Focus: Implementation
- Which components can and should be exposed how as services?

- WS*
- …

- Implementation of components

- Programming languages
- …



Industry Value Chain

Suppliers & Partners | Design & Build Infrastructure | Operate Infrastructure | Develop & Promote Products & Services | Sell & Fulfill | Bill & Collect | Assist Customers | Customers & Channels

Network Life-Cycle Management
- Demand and Supply Planning
- Investment Management
- Network Design and Build
- Operations and Maintenance

Marketing Analytics & Product Management

Sales & Service Fulfillment
- Lead and Opportunity Management
- Sales and Order Management
- Customer Field Service Management
- Logistics Management

Dealer Management

Billing, Invoicing & Presentment

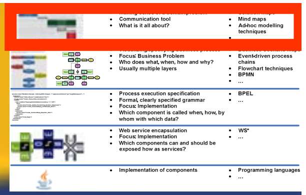
Customer Financial Management

Customer Service
- Customer Trouble Management
- Complaints and Returns Management
- Customer Field Service Management

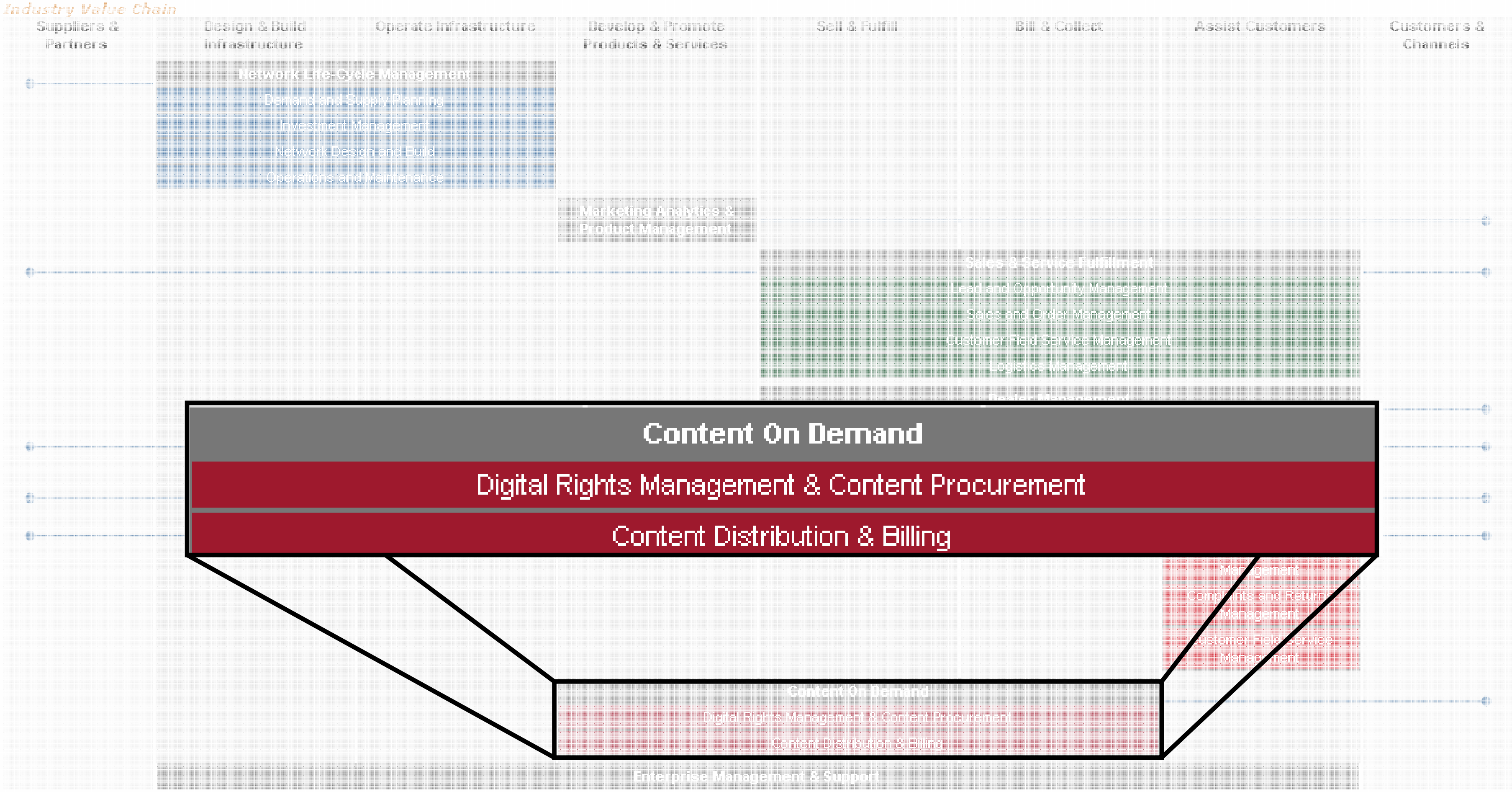
Content On Demand
- Digital Rights Management & Content Procurement
- Content Distribution & Billing

Enterprise Management & Support

Content on Demand



• Communication tool • What is it all about?	• Mind maps • Ad-hoc modelling techniques •
• Focus: Business Problem • Who does what, when, how and why? • Usually multiple layers	• Event-driven process chains • Flowchart techniques • BPMN • …
• Process execution specification • Formal, clearly specified grammar • Focus: Implementation • Which component is called when, how, by whom with which data?	• BPEL • …
• Web service encapsulation • Focus: Implementation • Which components can and should be exposed how as services?	• WS* • …
• Implementation of components	• Programming languages • …



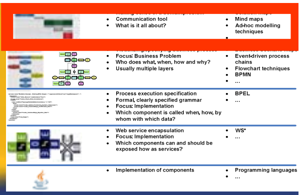
Content On Demand

Digital Rights Management & Content Procurement

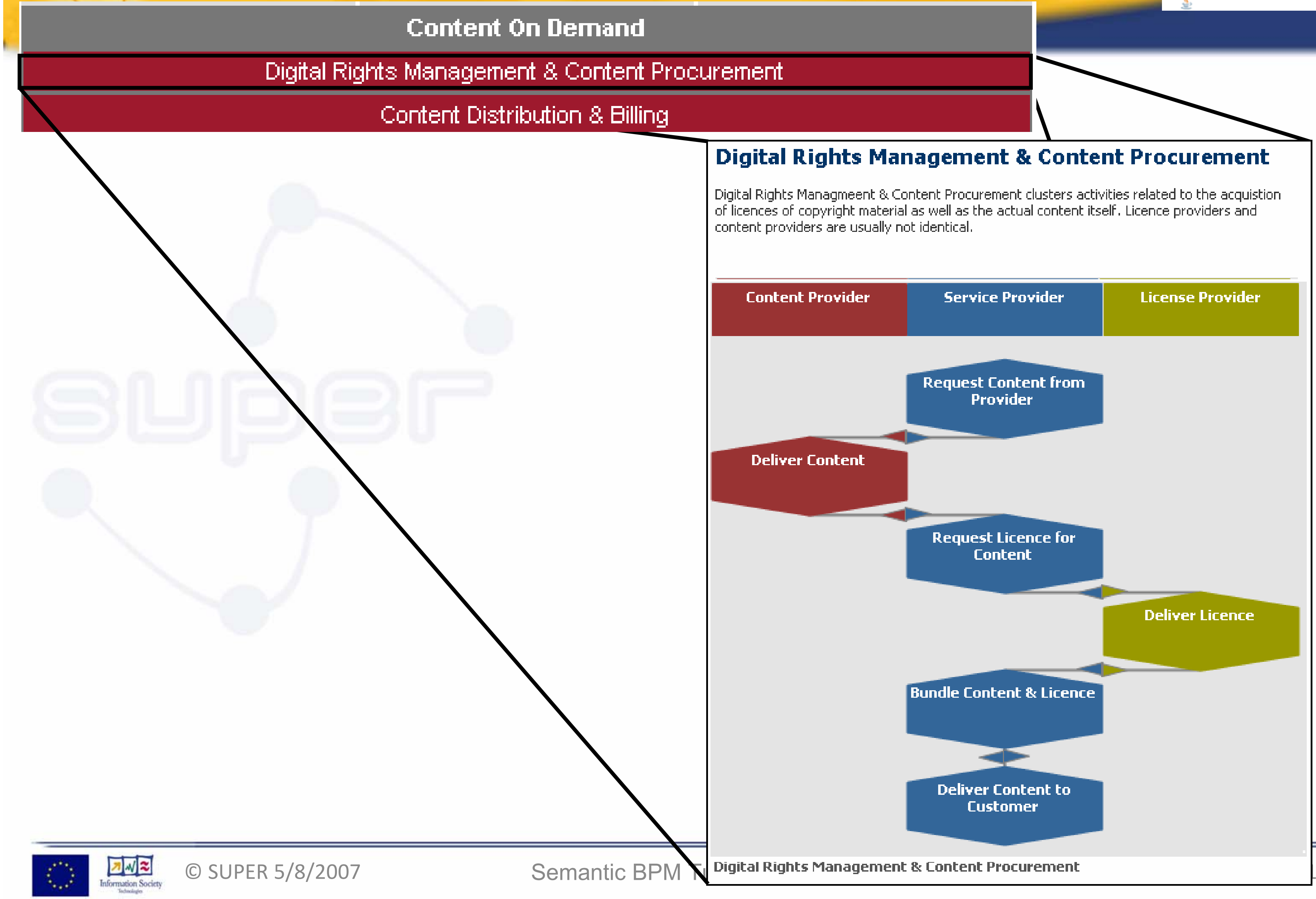
Content Distribution & Billing

Industry Value Chain
Suppliers & Partners
Design & Build Infrastructure
Operate Infrastructure
Develop & Promote Products & Services
Sell & Fulfill
Bill & Collect
Assist Customers
Customers & Channels
Network Life-Cycle Management
Demand and Supply Planning
Investment Management
Network Design and Build
Operations and Maintenance
Marketing Analytics & Product Management
Sales & Service Fulfillment
Lead and Opportunity Management
Sales and Order Management
Customer Field Service Management
Logistics Management
Dealer Management
Management
Complaints and Returns Management
Customer Field Service Management
Content On Demand
Digital Rights Management & Content Procurement
Content Distribution & Billing
Enterprise Management & Support

Digital Rights Management & Content Procurement



• Communication tool • What is it all about?	• Mind maps • Ad-hoc modelling techniques •
• Focus: Business Problem • Who does what, when, how and why? • Usually multiple layers	• Event-driven process chains • Flowchart techniques • BPMN • ...
• Process execution specification • Formal, clearly specified grammar • Focus: Implementation • Which component is called when, how, by whom with which data?	• BPEL • ...
• Web service encapsulation • Focus: Implementation • Which components can and should be exposed how as services?	• WS* • ...
• Implementation of components	• Programming languages • ...


Content On Demand
Digital Rights Management & Content Procurement
Content Distribution & Billing

Digital Rights Management & Content Procurement

Digital Rights Managmeent & Content Procurement clusters activities related to the acquistion of licences of copyright material as well as the actual content itself. Licence providers and content providers are usually not identical.

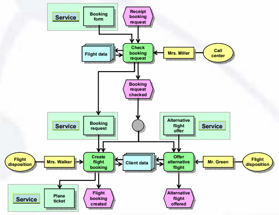
Content Provider
Service Provider
License Provider
Request Content from Provider
Deliver Content
Request Licence for Content
Deliver Licence
Bundle Content & Licence
Deliver Content to Customer

Digital Rights Management & Content Procurement

Modelling Stack

- Making sense of a domain\problem
- Communication tool
- What is it all about?

- Solution maps
- Mind maps
- Ad-hoc modelling techniques
- ...



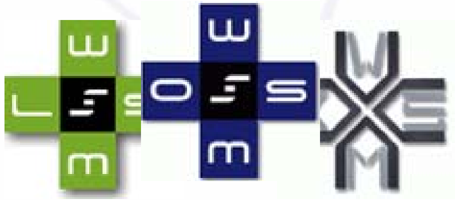
- Visualizing\specifying business process
- Focus: Business Problem
- Who does what, when, how and why?
- Usually multiple layers

- Business Scenario Maps
- Event-driven process chains
- Flowchart techniques
- BPMN
- ...

```
<process name="Mediation Example - Ordering BPEL Snippet - 1" suppressJoinFailure="yes" targetNamespace="...">
  <sequence>
    <receive name="Initial_Receive" createInstance="true"/>
    <invoke name="Invoke_Check_Order_Consistency"/>
    <switch>
      <case condition="bpws:getVariableData('consistency', '') = 'OK'">
        <flow>
          <invoke name="Invoke_Update_Provisioning_Systems_Subprocess"/>
          <invoke name="Invoke_CustomerReply_Confirmation_Note"/>
        </flow>
      </case>
      <otherwise>
        <invoke name="Invoke_CustomerReply_Rejection_Note"/>
      </otherwise>
    </switch>
    <reply name="Final_Reply"/>
  </sequence>
</process>
```

- Process execution specification
- Formal, clearly specified grammar
- Focus: Implementation
- Which component is called when, how, by whom with which data?

- BPEL
- ...



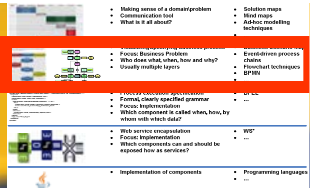
- Web service encapsulation
- Focus: Implementation
- Which components can and should be exposed how as services?

- WS*
- ...

- Implementation of components

- Programming languages
- ...

Integration in the ARIS House



- Making sense of a domain\problem
- Communication tool
- What is it all about?

- Solution maps
- Mind maps
- Ad-hoc modelling techniques
- ...

- Visualizing\specifying business process
- Focus: Business Problem
- Who does what, when, how and why?
- Usually multiple layers

- Business Scenario Ma
- Event-driven process chains
- Flowchart techniques
- BPMN
- ...

- Process execution specification
- Formal, clearly specified grammar
- Focus: Implementation
- Which component is called when, how, by whom with which data?

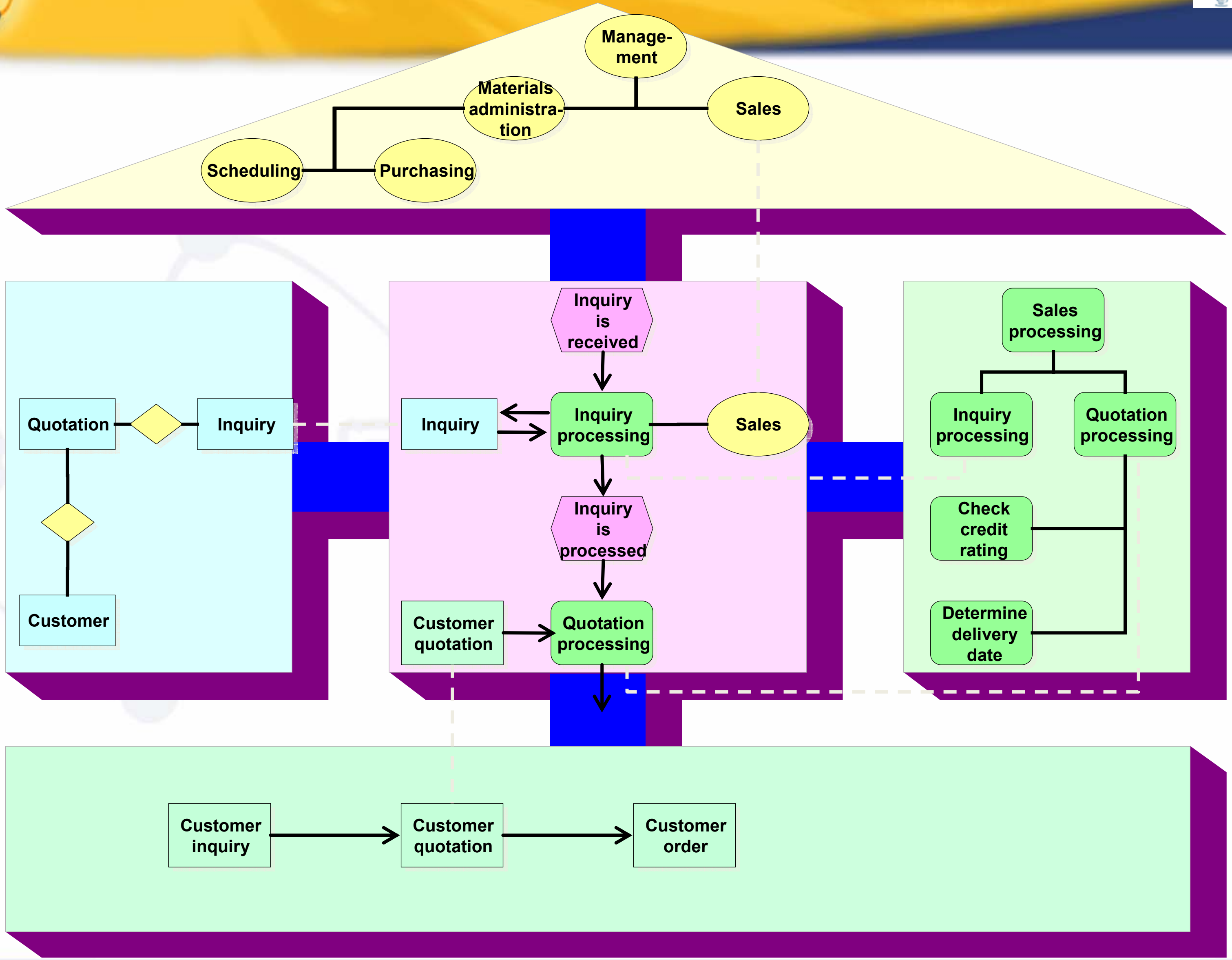
- BPEL
- ...

- Web service encapsulation
- Focus: Implementation
- Which components can and should be exposed how as services?

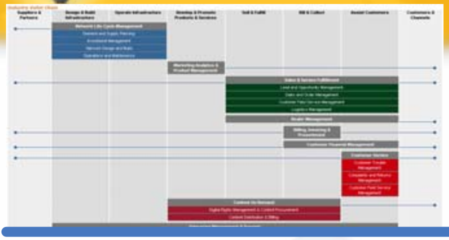
- WS*
- ...

- Implementation of components

- Programming languages
- ...

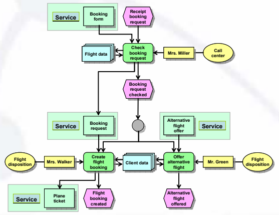


Modelling Stack



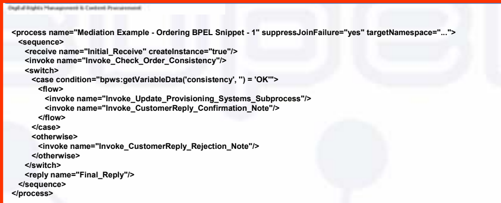
- Making sense of a domain\problem
- Communication tool
- What is it all about?

- Solution maps
- Mind maps
- Ad-hoc modelling techniques
- ...



- Visualizing\specifying business process
- Focus: Business Problem
- Who does what, when, how and why?
- Usually multiple layers

- Business Scenario Maps
- Event-driven process chains
- Flowchart techniques
- BPMN
- ...



Digital Rights Management & Content Procurement

```
<process name="Mediation Example - Ordering BPEL Snippet - 1" suppressJoinFailure="yes" targetNamespace="...">
  <sequence>
    <receive name="Initial_Receive" createInstance="true"/>
    <invoke name="Invoke_Check_Order_Consistency"/>
    <switch>
      <case condition="bpws:getVariableData('consistency', '') = 'OK'">
        <flow>
          <invoke name="Invoke_Update_Provisioning_Systems_Subprocess"/>
          <invoke name="Invoke_CustomerReply_Confirmation_Note"/>
        </flow>
      </case>
      <otherwise>
        <invoke name="Invoke_CustomerReply_Rejection_Note"/>
      </otherwise>
    </switch>
    <reply name="Final_Reply"/>
  </sequence>
</process>
```

- Process execution specification
- Formal, clearly specified grammar
- Focus: Implementation
- Which component is called when, how, by whom with which data?

- BPEL
- ...

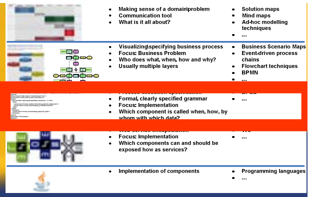
- Web service encapsulation
- Focus: Implementation
- Which components can and should be exposed how as services?

- WS*
- ...

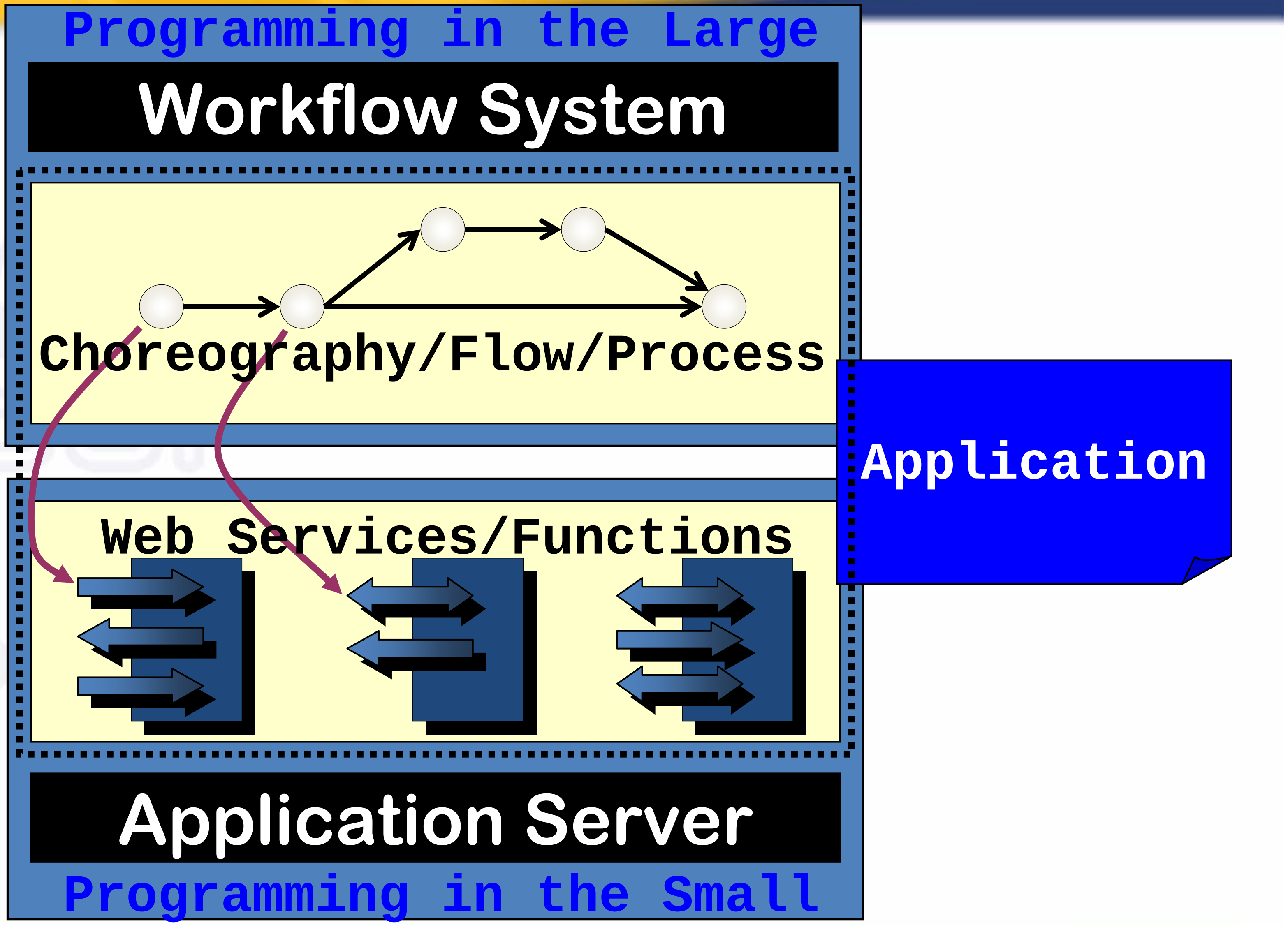
- Implementation of components

- Programming languages
- ...

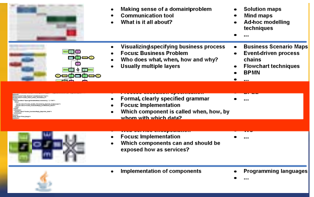
Programming Model



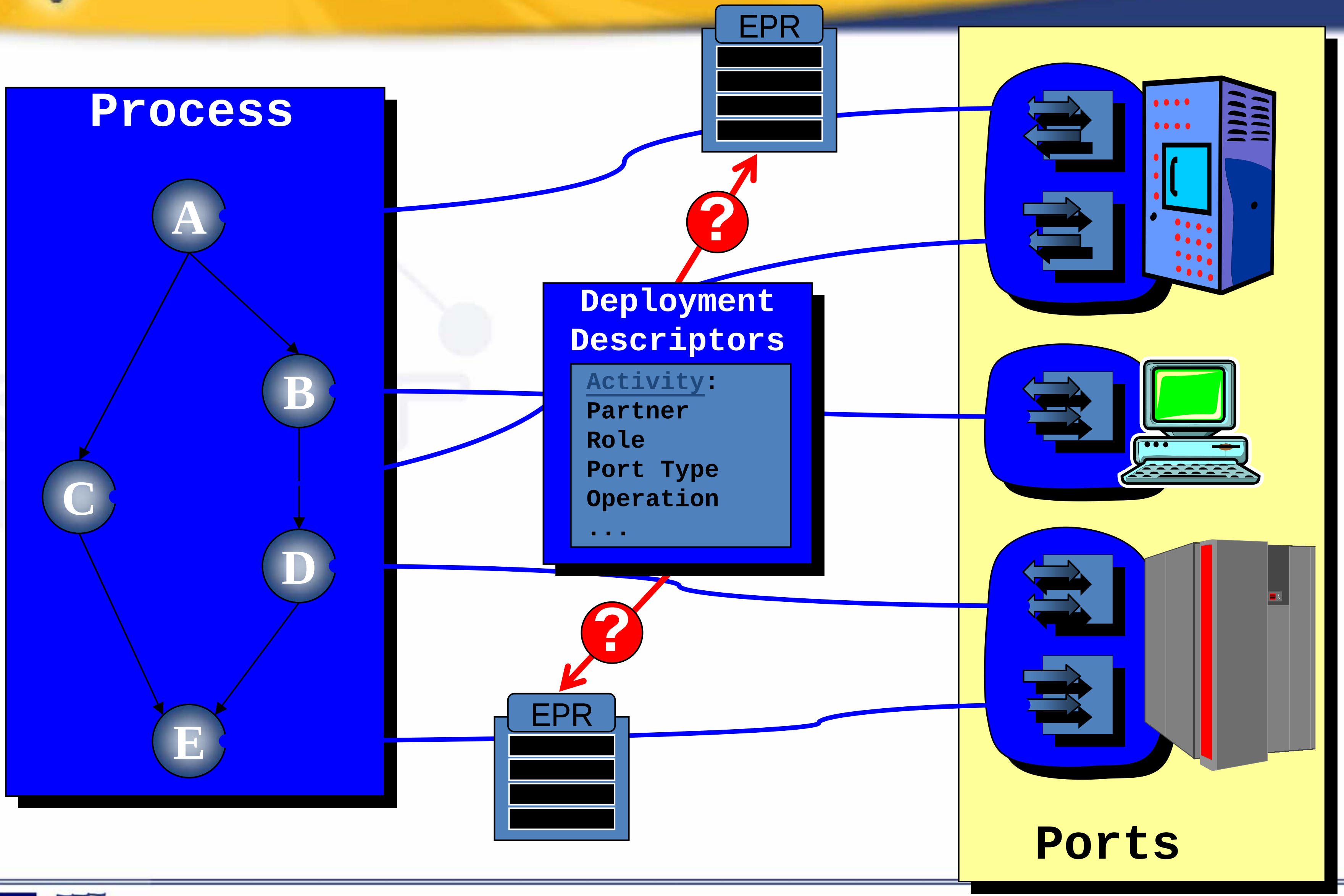
• Making sense of a domain\problem • Communication tool • What is it all about?	• Solution maps • Mind maps • Ad-hoc modelling techniques • ...
• Visualizing\specifying business process • Focus: Business Problem • Who does what, when, how and why? • Usually multiple layers	• Business Scenario Maps • Event-driven process chains • Flowchart techniques • BPMN • ...
• Process execution specification • Formal, clearly specified grammar • Focus: Implementation • Which component is called when, how, by whom with which data?	• BPEL • ...
• Web service encapsulation • Focus: Implementation • Which components can and should be exposed how as services?	• WS • ...
• Implementation of components	• Programming languages • ...



Deploying Applications



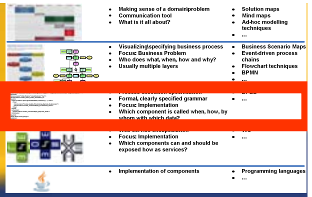
	- Making sense of a domain\problem - Communication tool - What is it all about?	- Solution maps - Mind maps - Ad-hoc modelling techniques - ...
	- Visualizing\specifying business process - Focus: Business Problem - Who does what, when, how and why? - Usually multiple layers	- Business Scenario Maps - Event-driven process chains - Flowchart techniques - BPMN - ...
	- Process execution specification - Formal, clearly specified grammar - Focus: Implementation - Which component is called when, how, by whom with which data?	- BPEL - ...
	- Web service encapsulation - Focus: Implementation - Which components can and should be exposed how as services?	- WS - ...
	- Implementation of components	- Programming languages - ...



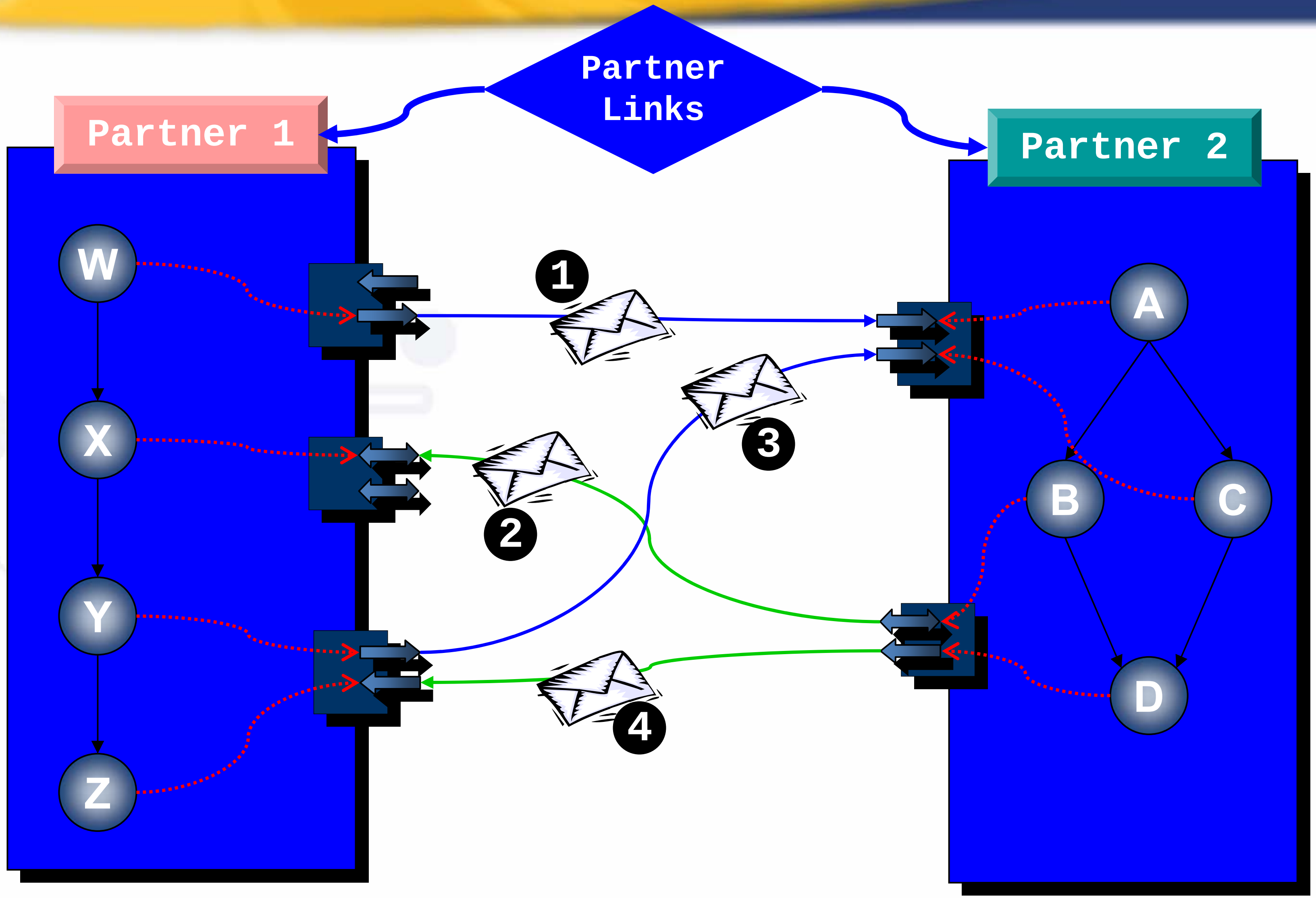
Deployment Descriptors

Activity:
Partner
Role
Port Type
Operation
...

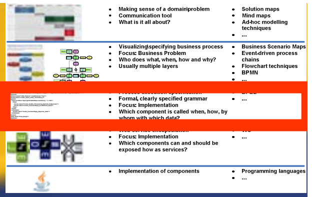
Business Protocols



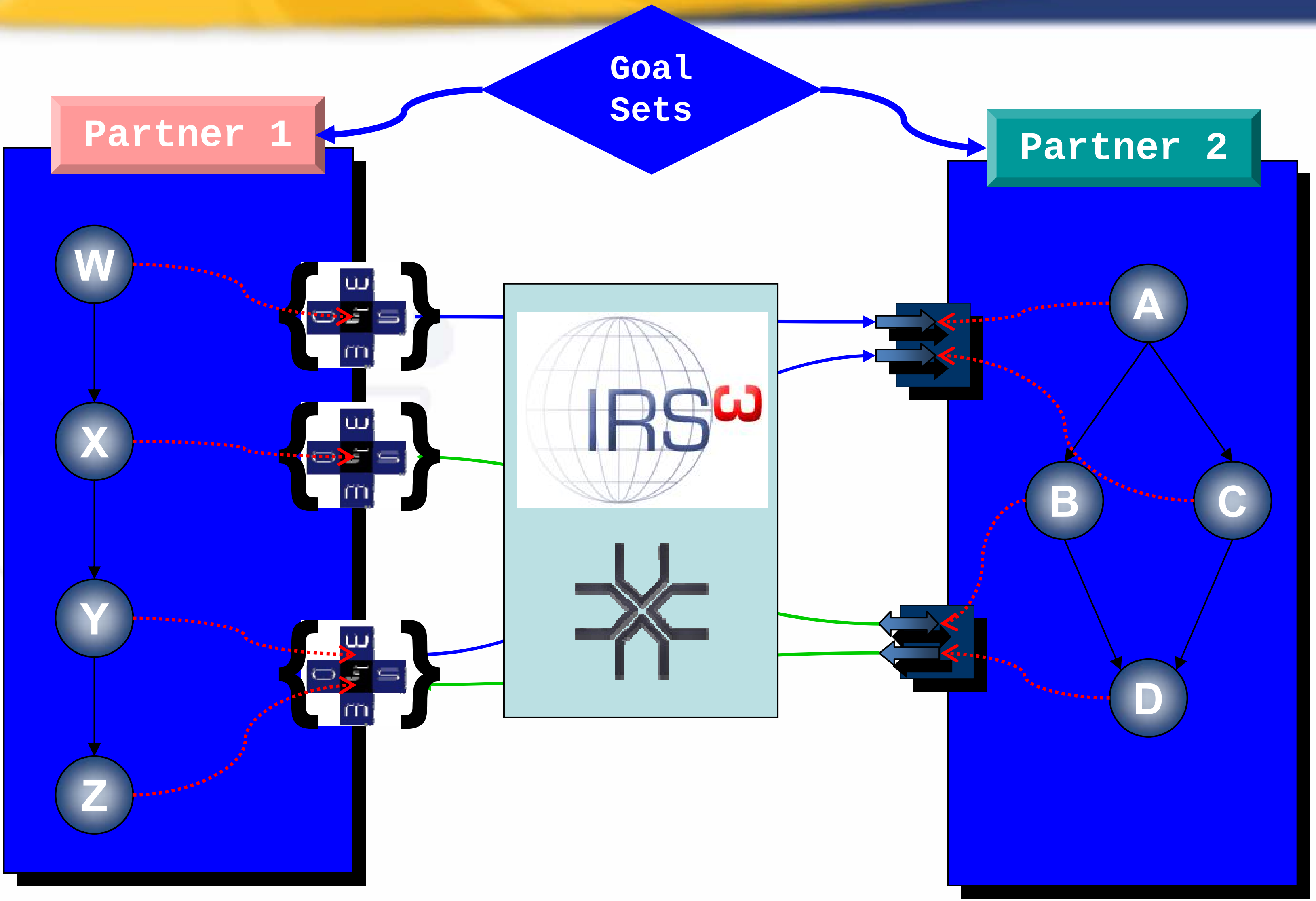
- Making sense of a domain\problem - Communication tool - What is it all about?	- Solution maps - Mind maps - Ad-hoc modelling techniques - ...
- Visualizing\specifying business process - Focus: Business Problem - Who does what, when, how and why? - Usually multiple layers	- Business Scenario Maps - Event-driven process chains - Flowchart techniques - BPMN - ...
- Formal, clearly specified grammar - Focus: Implementation - Which component is called when, how, by whom with which data?	- ...
- Focus: Implementation - Which components can and should be exposed how as services?	- ...
- Implementation of components	- Programming languages - ...



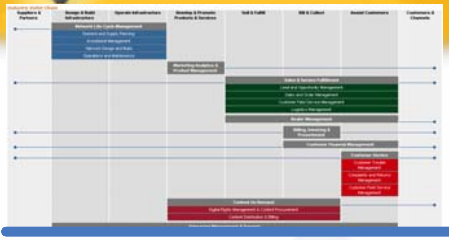
Business Protocols



	• Making sense of a domain\problem • Communication tool • What is it all about?	• Solution maps • Mind maps • Ad-hoc modelling techniques • ...
	• Visualizing\specifying business process • Focus: Business Problem • Who does what, when, how and why? • Usually multiple layers	• Business Scenario Maps • Event-driven process chains • Flowchart techniques • BPMN • ...
	• Process execution specification • Formal, clearly specified grammar • Focus: Implementation • Which component is called when, how, by whom with which data?	• BPEL • ...
	• Web service encapsulation • Focus: Implementation • Which components can and should be exposed how as services?	• WS • ...
	• Implementation of components	• Programming languages • ...

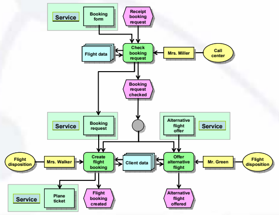


Modelling Stack



- Making sense of a domain\problem
- Communication tool
- What is it all about?

- Solution maps
- Mind maps
- Ad-hoc modelling techniques
- ...



- Visualizing\specifying business process
- Focus: Business Problem
- Who does what, when, how and why?
- Usually multiple layers

- Business Scenario Maps
- Event-driven process chains
- Flowchart techniques
- BPMN
- ...

```
<process name="Mediation Example - Ordering BPEL Snippet - 1" suppressJoinFailure="yes" targetNamespace="...">
  <sequence>
    <receive name="Initial_Receive" createInstance="true"/>
    <invoke name="Invoke_Check_Order_Consistency"/>
    <switch>
      <case condition="bpws:getVariableData('consistency', '') = 'OK'">
        <flow>
          <invoke name="Invoke_Update_Provisioning_Systems_Subprocess"/>
          <invoke name="Invoke_CustomerReply_Confirmation_Note"/>
        </flow>
      </case>
      <otherwise>
        <invoke name="Invoke_CustomerReply_Rejection_Note"/>
      </otherwise>
    </switch>
    <reply name="Final_Reply"/>
  </sequence>
</process>
```

- Process execution specification
- Formal, clearly specified grammar
- Focus: Implementation
- Which component is called when, how, by whom with which data?

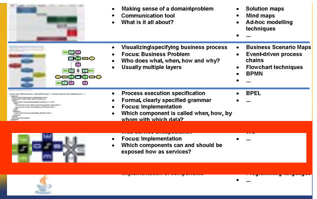
- BPEL
- ...

- Web service encapsulation
- Focus: Implementation
- Which components can and should be exposed how as services?

- WS*
- ...

- Implementation of components

- Programming languages
- ...

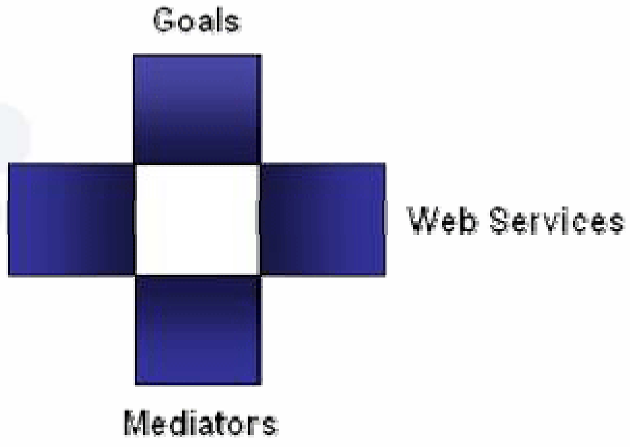


	- Making sense of a domain\problem - Communication tool - What is it all about?	- Solution maps - Mind maps - Ad-hoc modelling techniques - ...
	- Visualizing\specifying business process - Focus: Business Problem - Who does what, when, how and why? - Usually multiple layers	- Business Scenario Maps - Event-driven process chains - Flowchart techniques - BPMN - ...
	- Process execution specification - Formal, clearly specified grammar - Focus: Implementation - Which component is called when, how, by whom with which data?	- BPEL - ...
	- Focus: Implementation - Which components can and should be exposed how as services?	- ...
		- ...

WSMO Top Level Notions

Objectives that a client wants to achieve by using Web Services

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



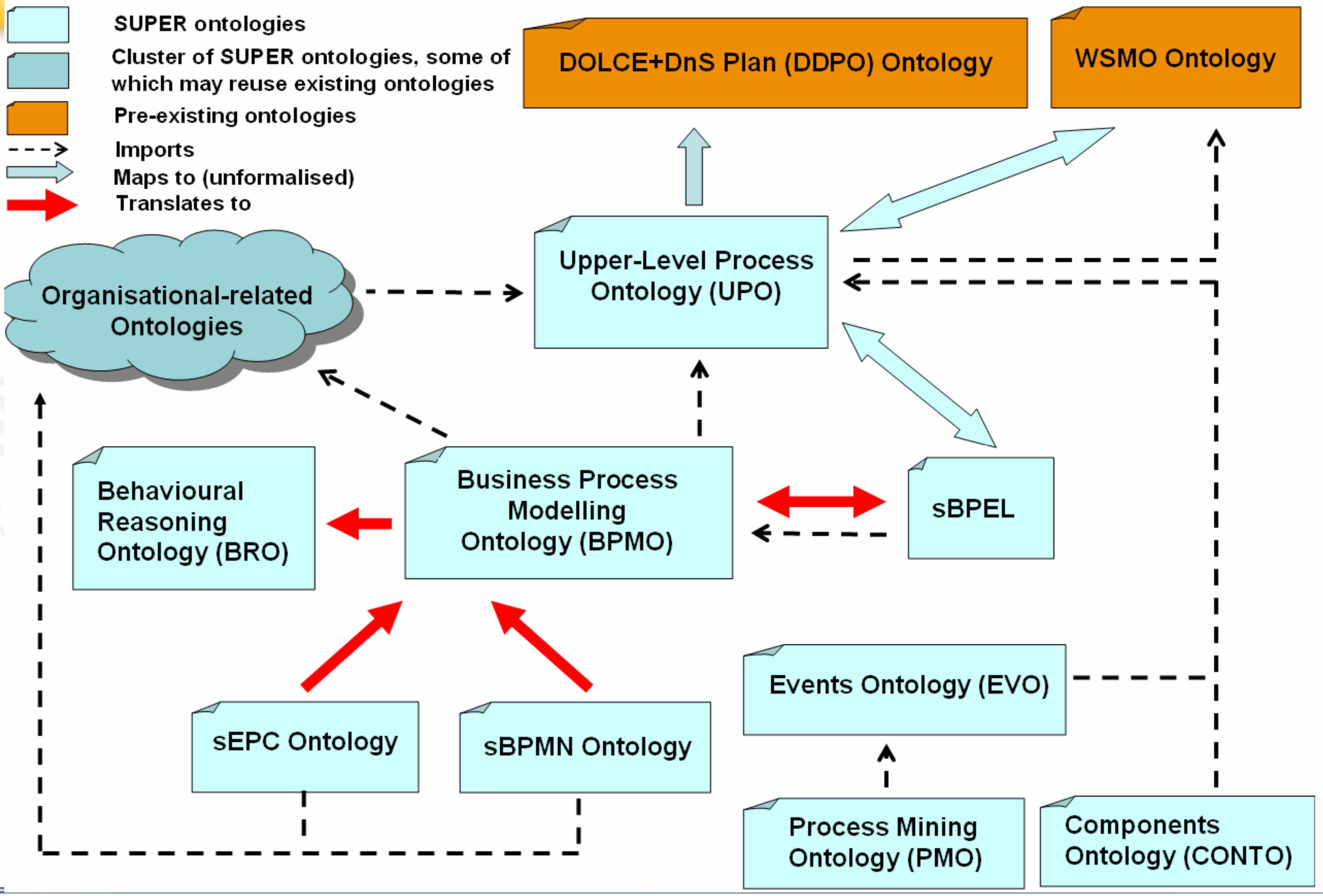
Semantic description of Web Services:

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

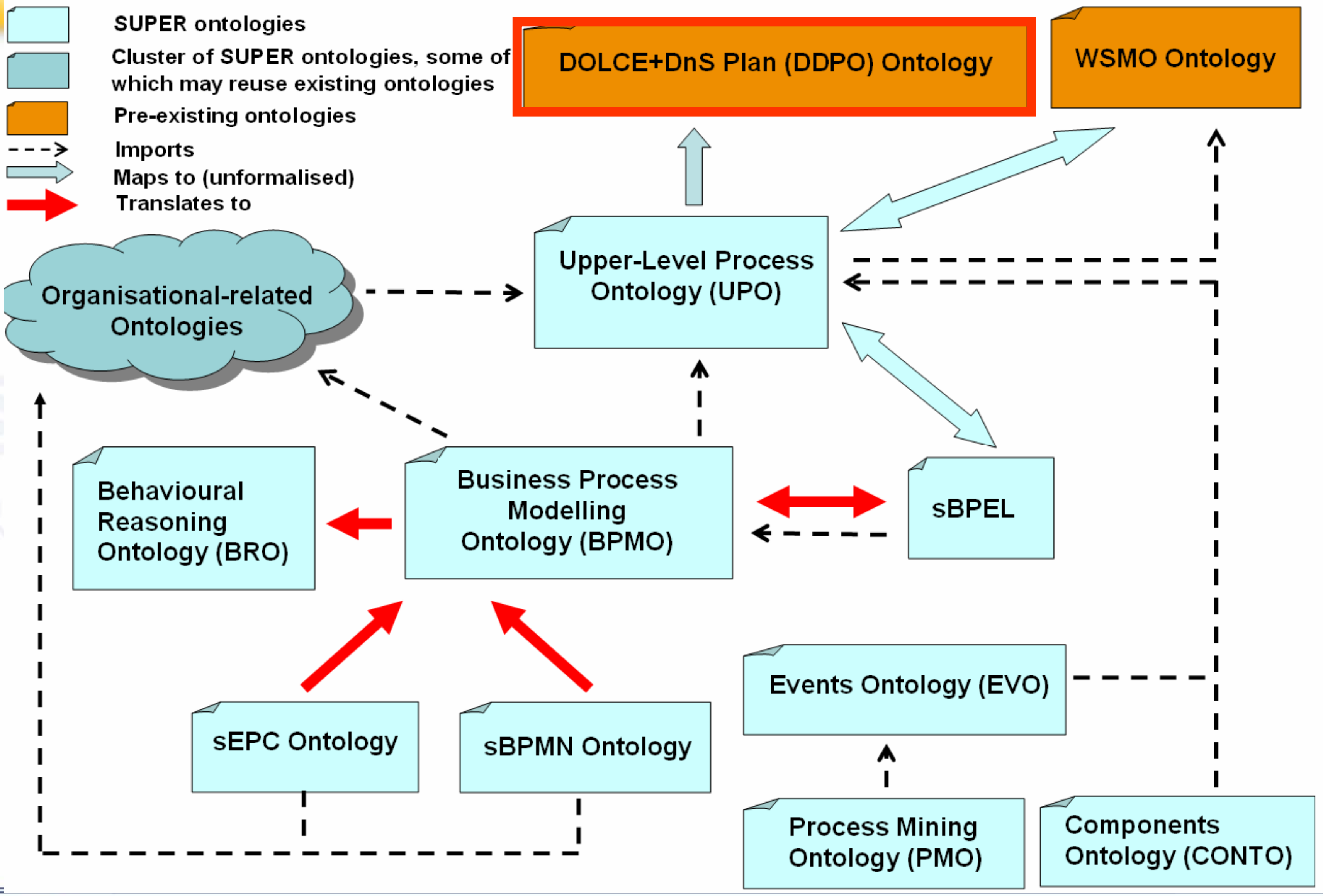
Connectors between components with mediation facilities for handling heterogeneities

SUPER Ontology Stack

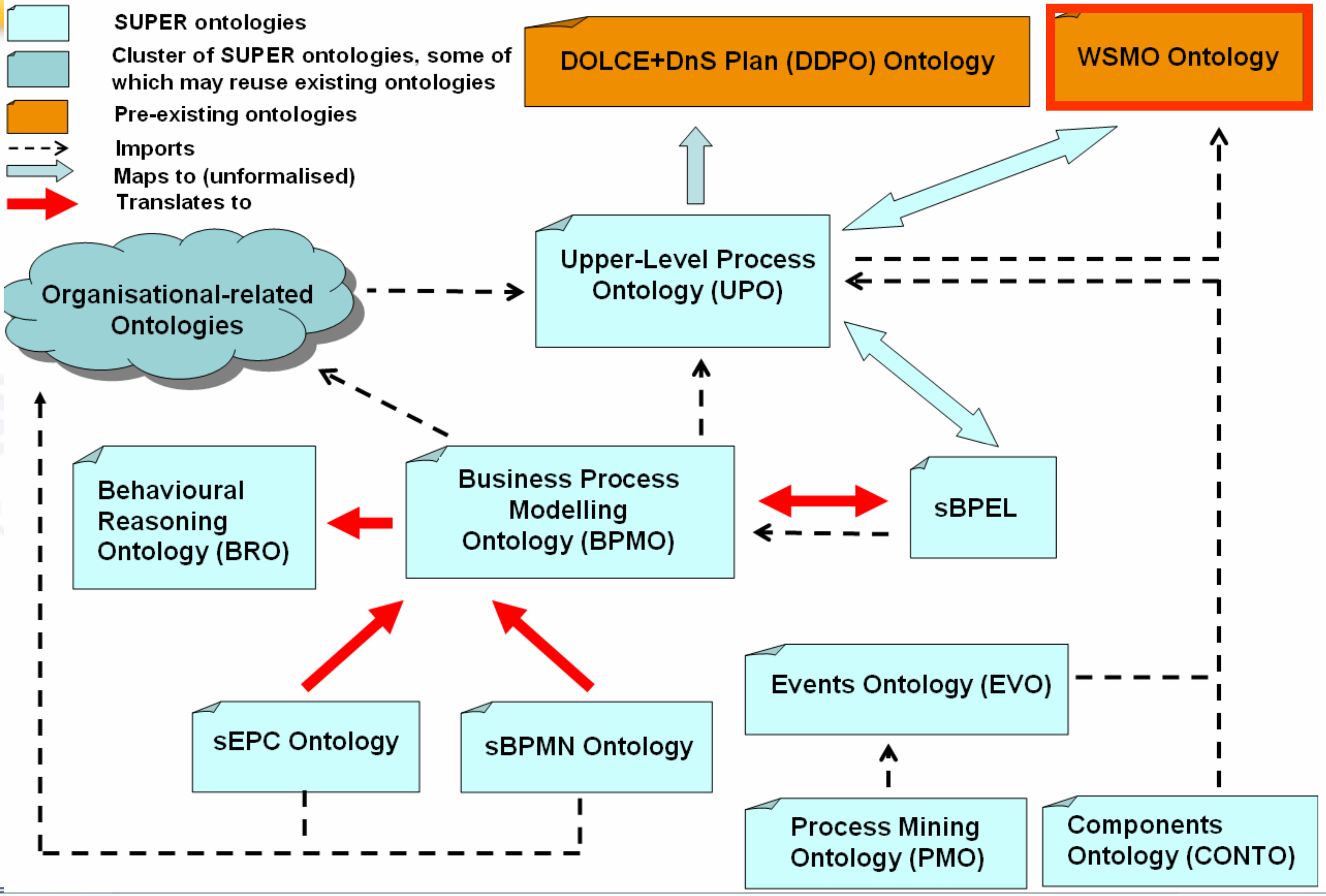
Ontologies overview


SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Upper-Level Process Ontology (UPO)
Organisational-related Ontologies
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
Events Ontology (EVO)
sEPC Ontology
sBPMN Ontology
Process Mining Ontology (PMO)
Components Ontology (CONTO)

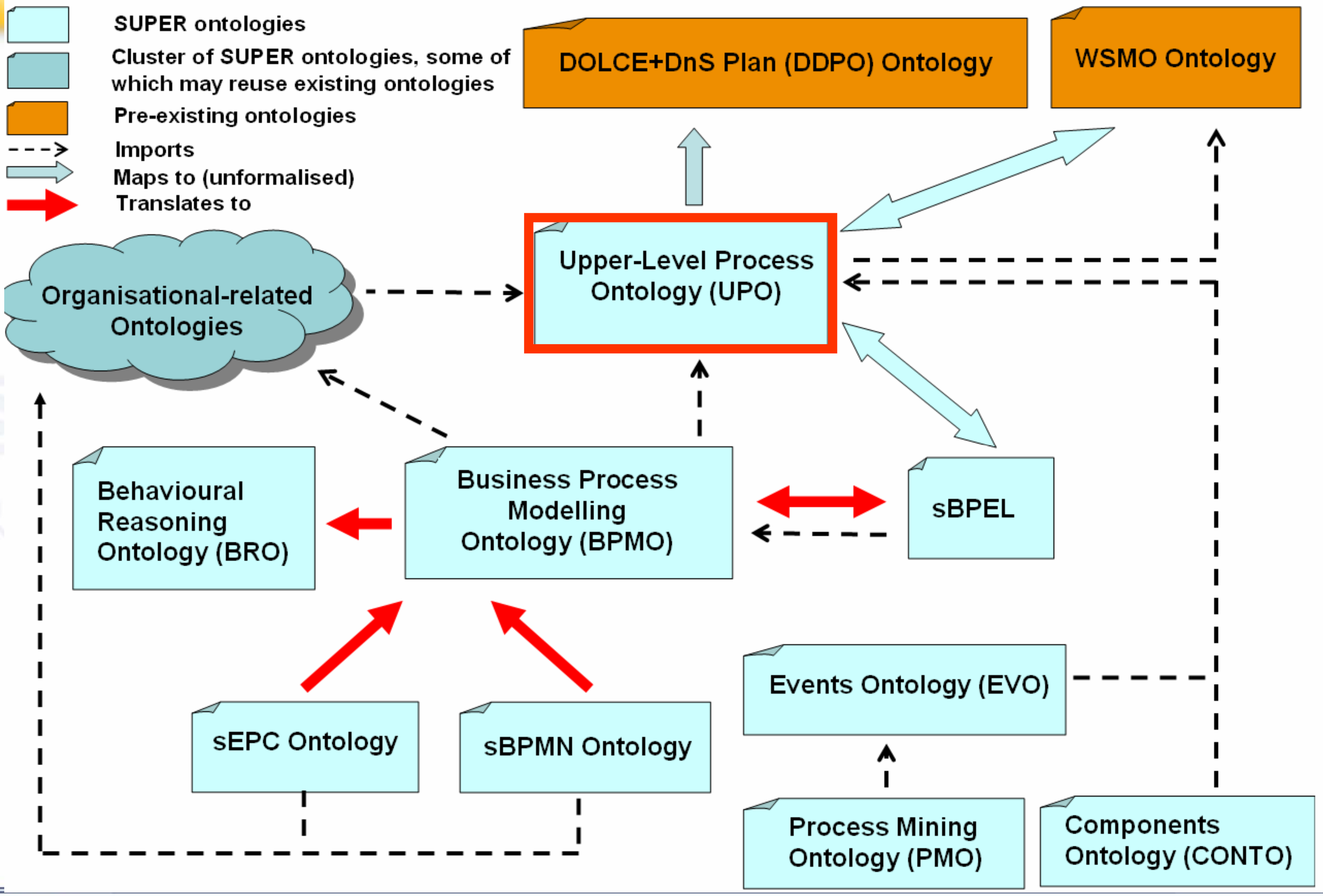
Ontologies overview


SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Upper-Level Process Ontology (UPO)
Organisational-related Ontologies
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
Events Ontology (EVO)
sEPC Ontology
sBPMN Ontology
Process Mining Ontology (PMO)
Components Ontology (CONTO)

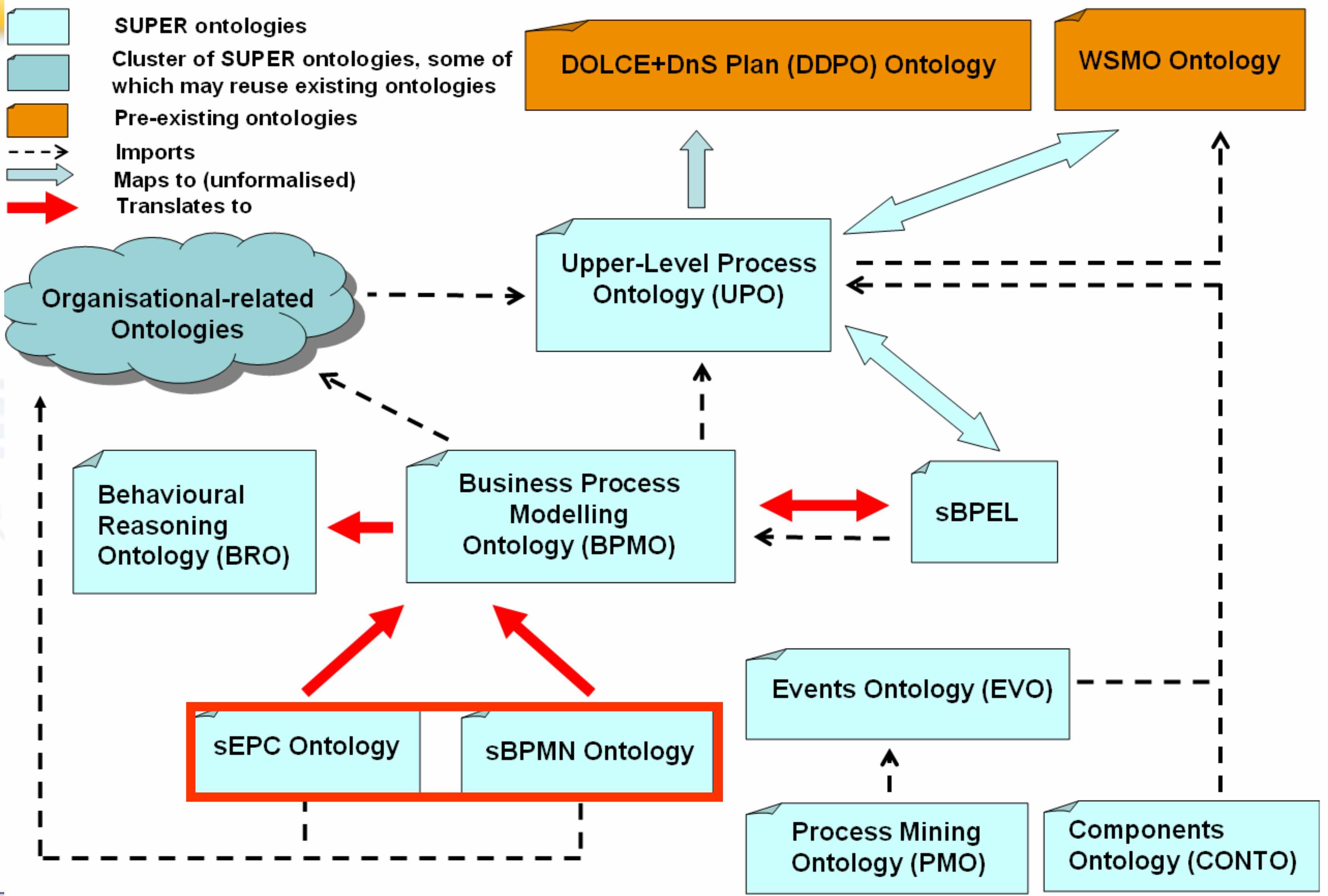
Ontologies overview


SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Upper-Level Process Ontology (UPO)
Organisational-related Ontologies
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
sEPC Ontology
sBPMN Ontology
Events Ontology (EVO)
Process Mining Ontology (PMO)
Components Ontology (CONTO)

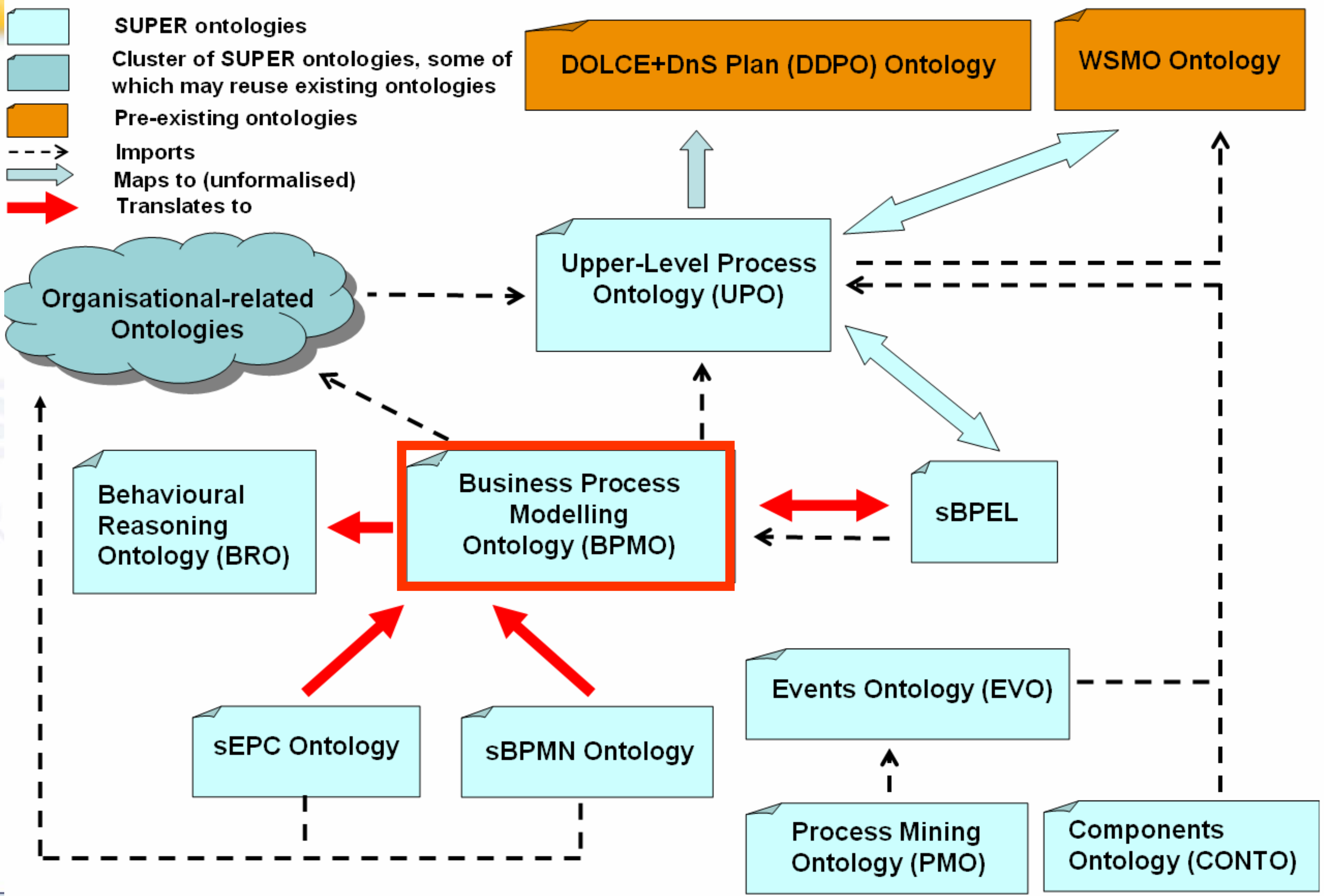
Ontologies overview


SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Organisational-related Ontologies
Upper-Level Process Ontology (UPO)
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
Events Ontology (EVO)
sEPC Ontology
sBPMN Ontology
Process Mining Ontology (PMO)
Components Ontology (CONTO)

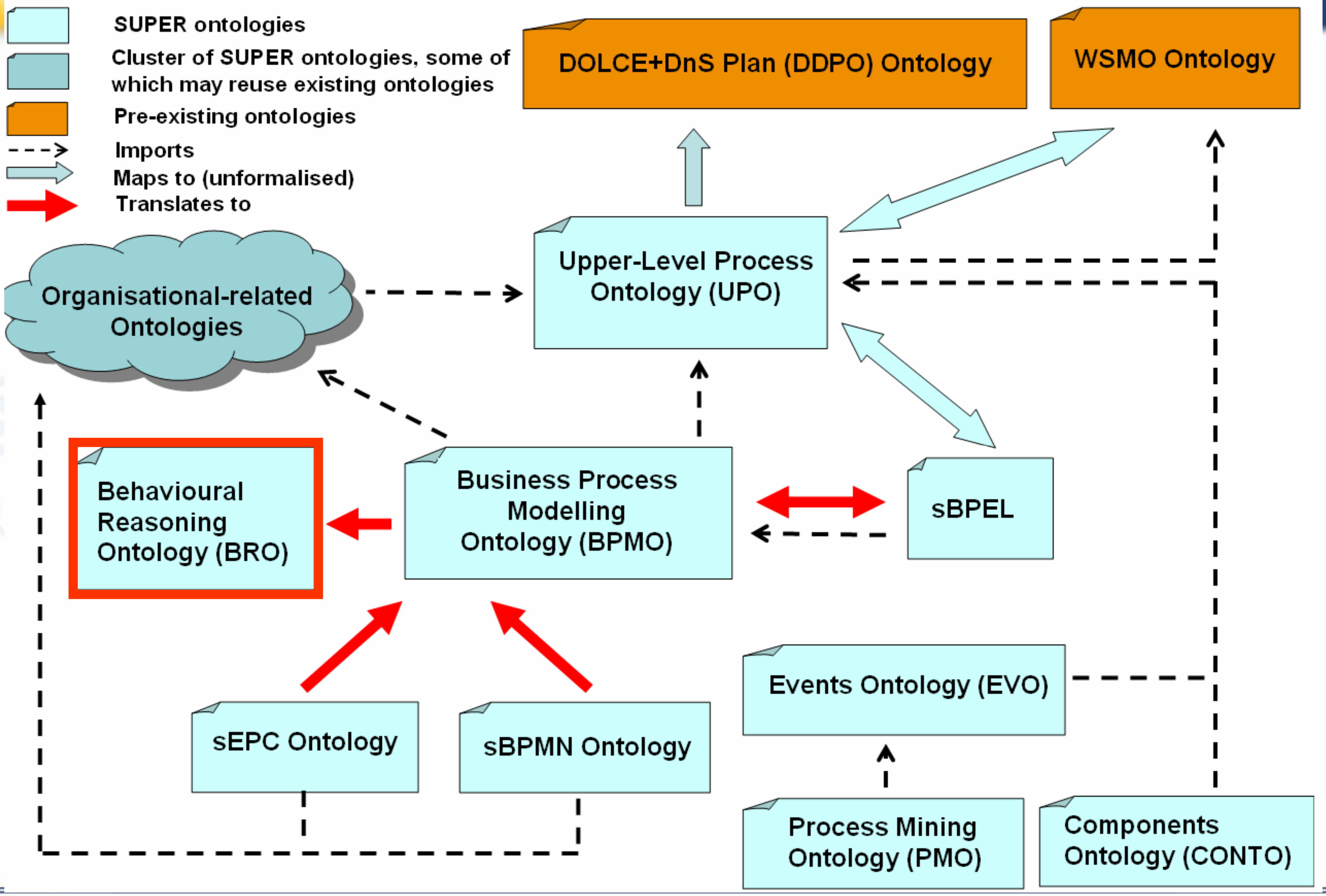
Ontologies overview


SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Upper-Level Process Ontology (UPO)
Organisational-related Ontologies
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
Events Ontology (EVO)
sEPC Ontology
sBPMN Ontology
Process Mining Ontology (PMO)
Components Ontology (CONTO)

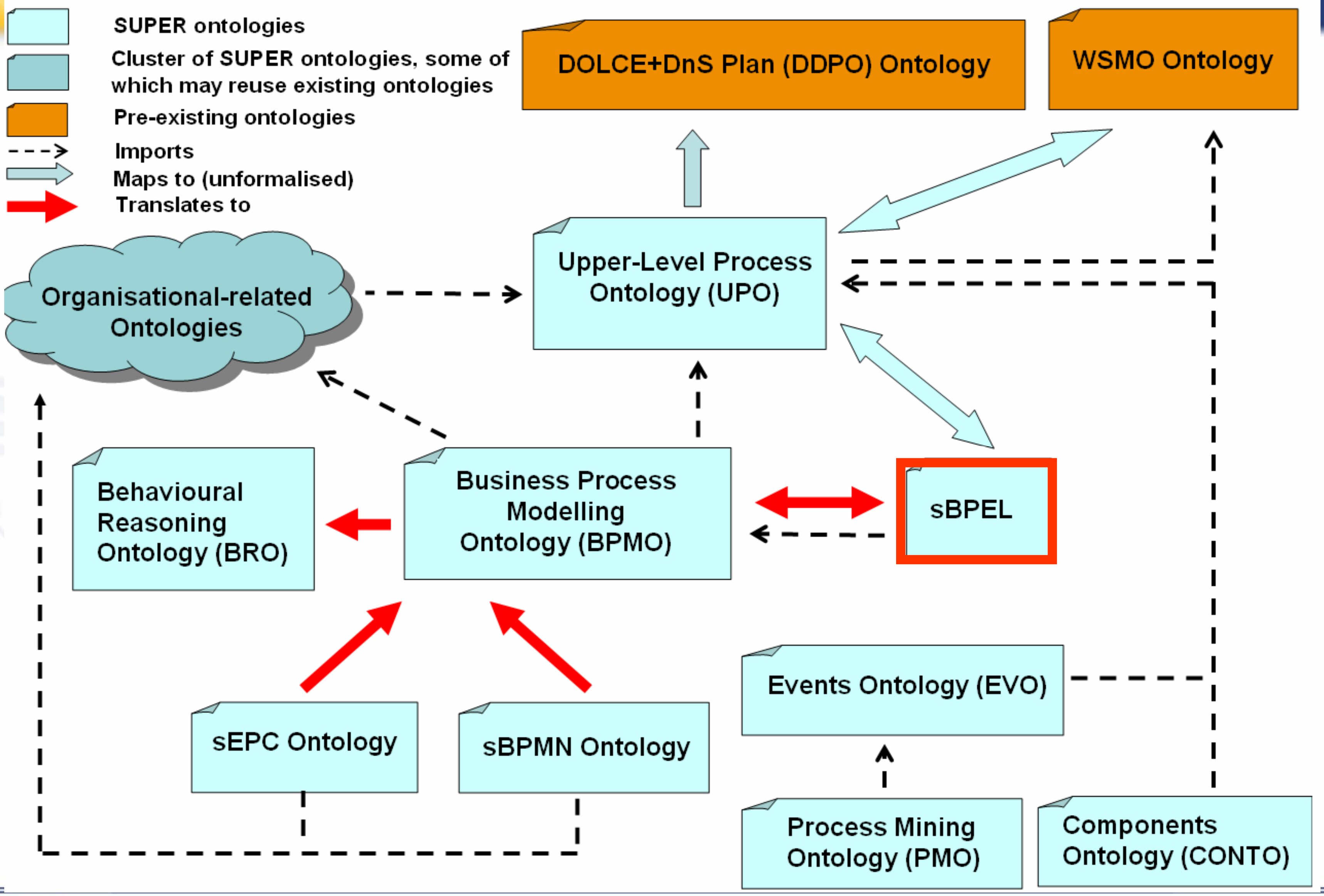
Ontologies overview


SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Upper-Level Process Ontology (UPO)
Organisational-related Ontologies
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
Events Ontology (EVO)
sEPC Ontology
sBPMN Ontology
Process Mining Ontology (PMO)
Components Ontology (CONTO)

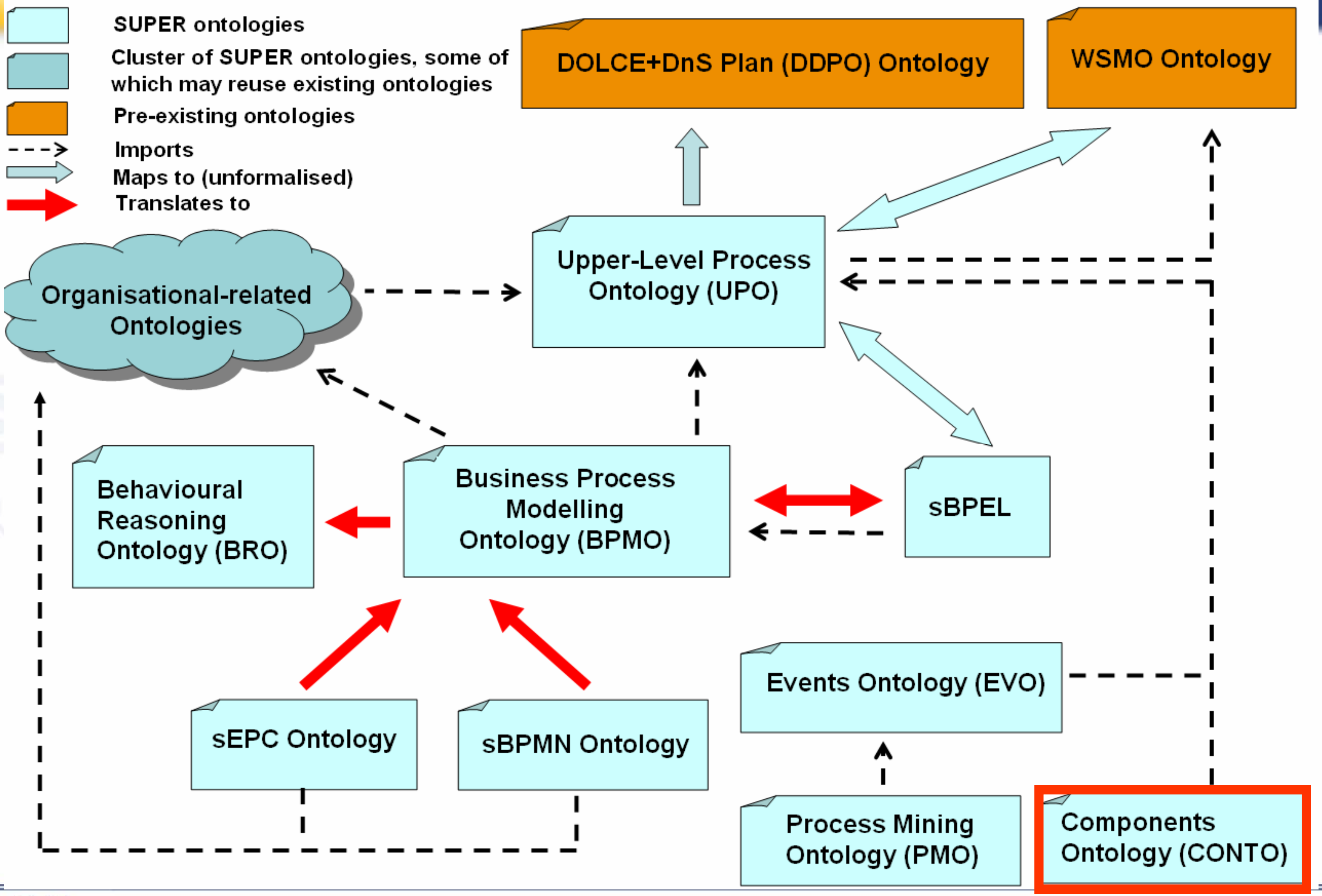
Ontologies overview


SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Organisational-related Ontologies
Upper-Level Process Ontology (UPO)
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
sEPC Ontology
sBPMN Ontology
Events Ontology (EVO)
Process Mining Ontology (PMO)
Components Ontology (CONTO)

Ontologies overview

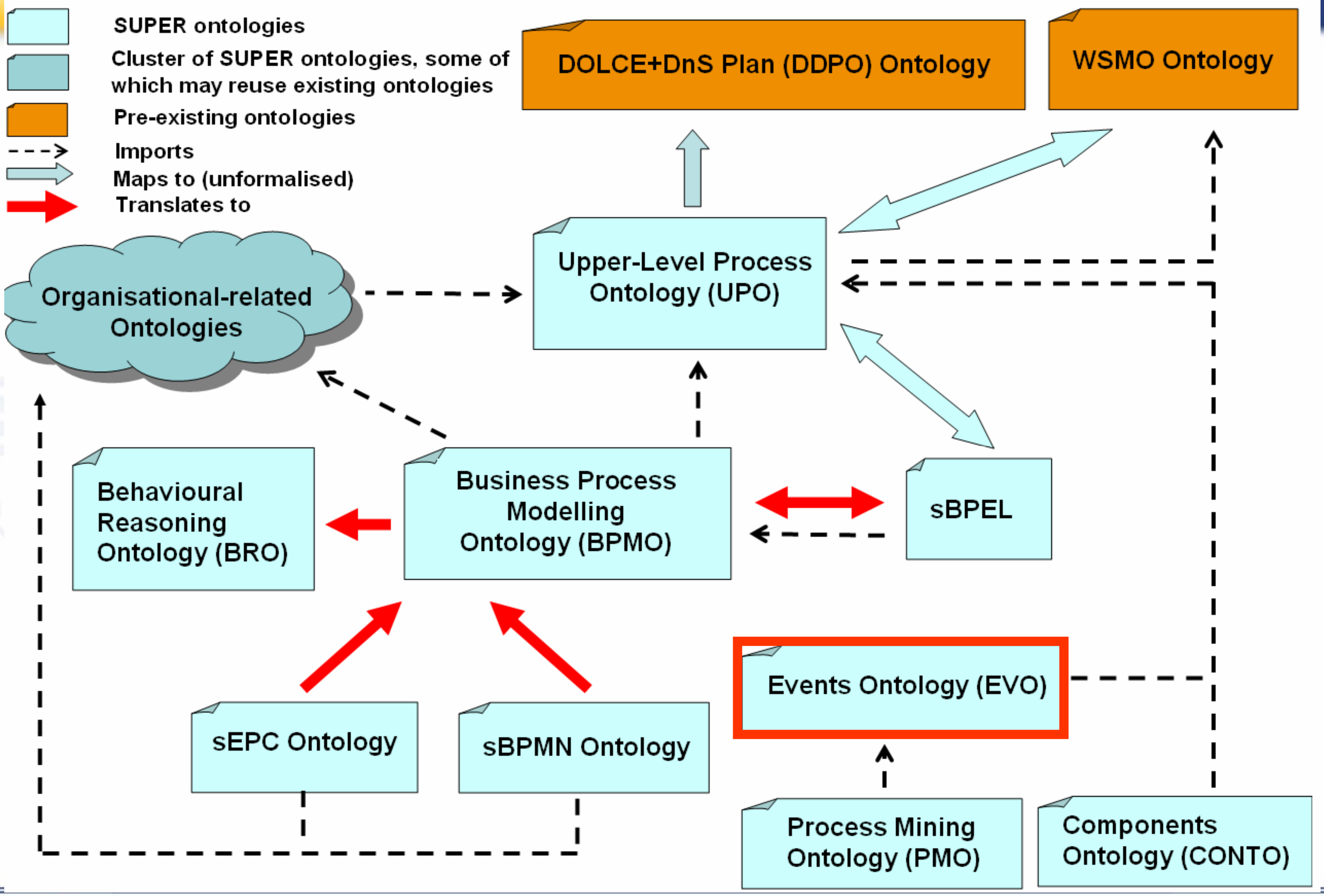

SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Upper-Level Process Ontology (UPO)
Organisational-related Ontologies
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
sEPC Ontology
sBPMN Ontology
Events Ontology (EVO)
Process Mining Ontology (PMO)
Components Ontology (CONTO)

Ontologies overview

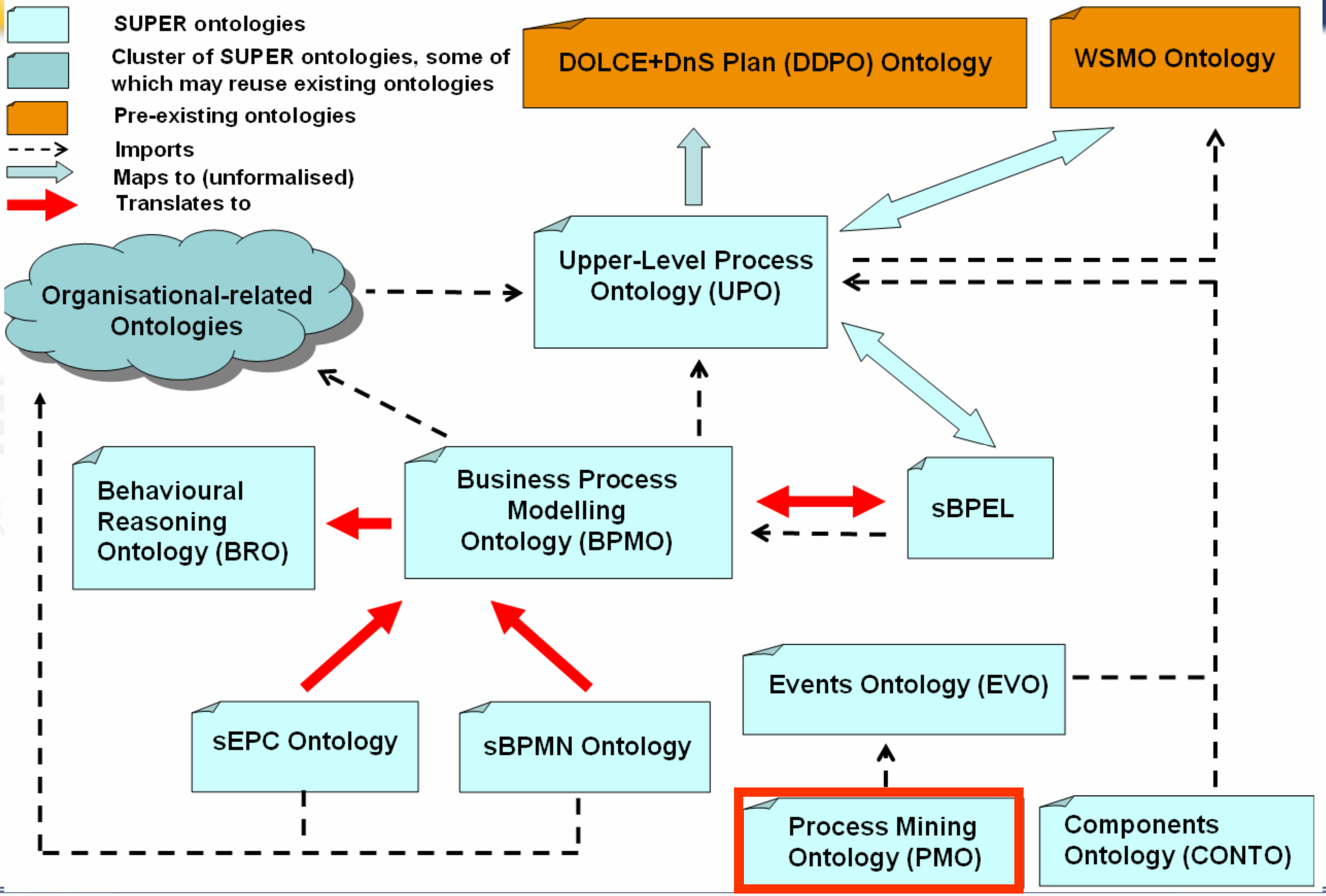


SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Organisational-related Ontologies
Upper-Level Process Ontology (UPO)
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
Events Ontology (EVO)
sEPC Ontology
sBPMN Ontology
Process Mining Ontology (PMO)
Components Ontology (CONTO)

Ontologies overview

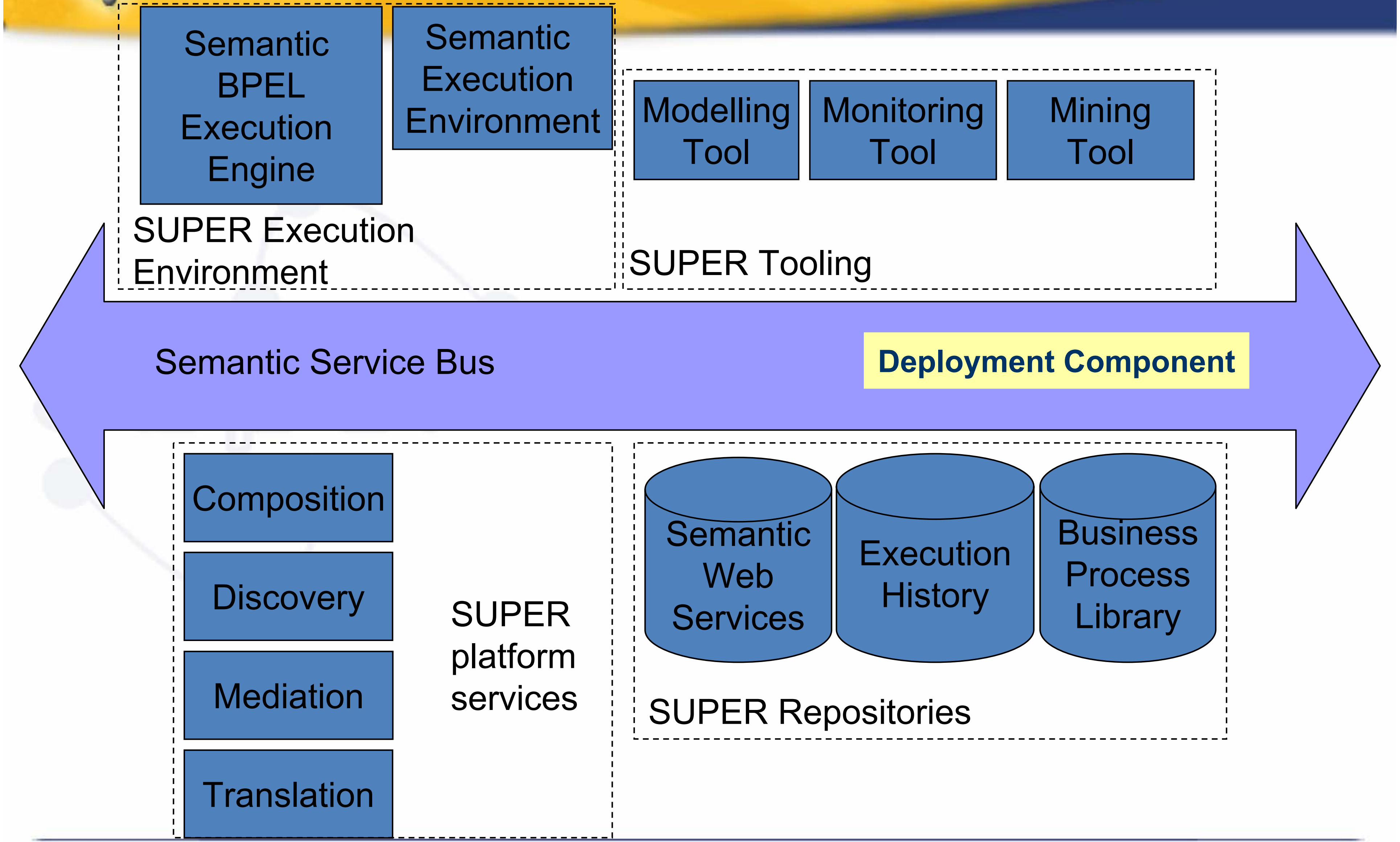

SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Upper-Level Process Ontology (UPO)
Organisational-related Ontologies
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
Events Ontology (EVO)
sEPC Ontology
sBPMN Ontology
Process Mining Ontology (PMO)
Components Ontology (CONTO)

Ontologies overview

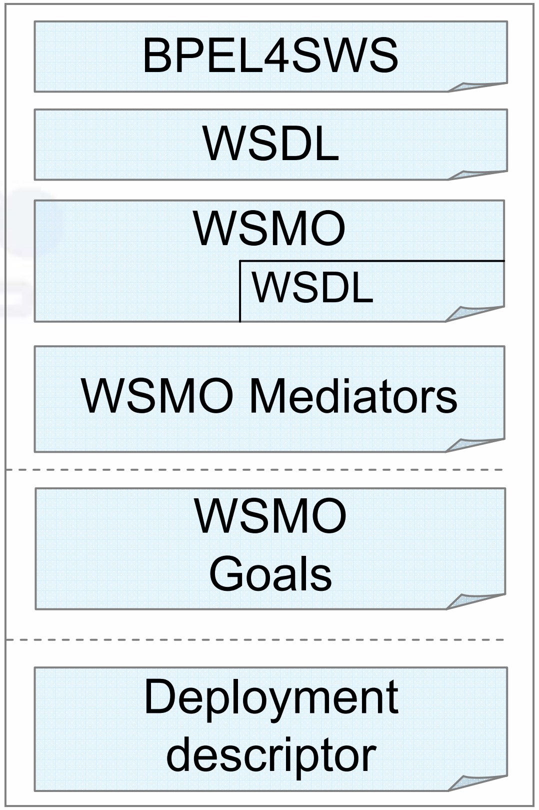

SUPER ontologies
Cluster of SUPER ontologies, some of which may reuse existing ontologies
Pre-existing ontologies
Imports
Maps to (unformalised)
Translates to
DOLCE+DnS Plan (DDPO) Ontology
WSMO Ontology
Organisational-related Ontologies
Upper-Level Process Ontology (UPO)
Behavioural Reasoning Ontology (BRO)
Business Process Modelling Ontology (BPMO)
sBPEL
sEPC Ontology
sBPMN Ontology
Events Ontology (EVO)
Process Mining Ontology (PMO)
Components Ontology (CONTO)

SUPER Architecture and Scenarios

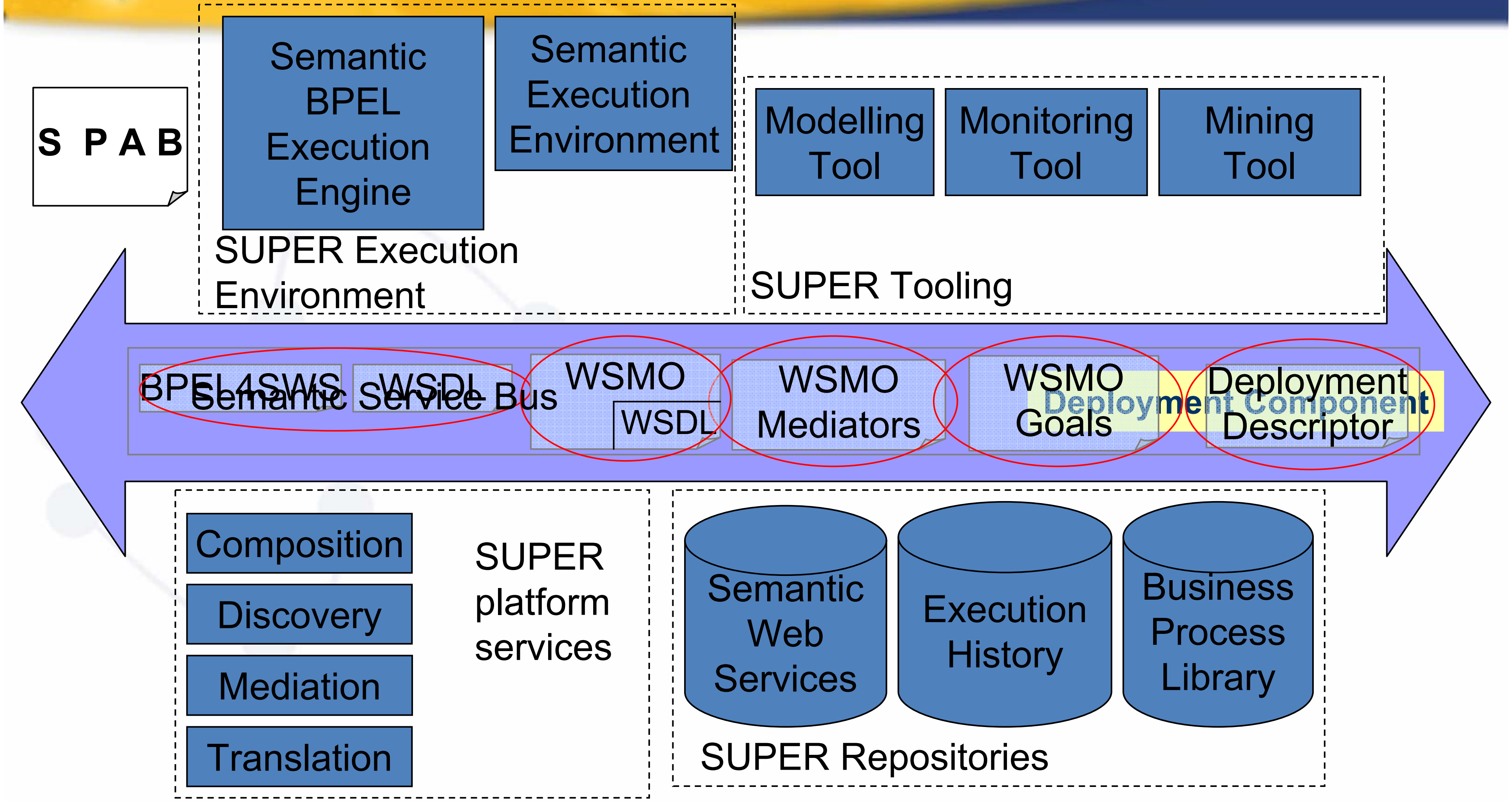
Architecture Static View



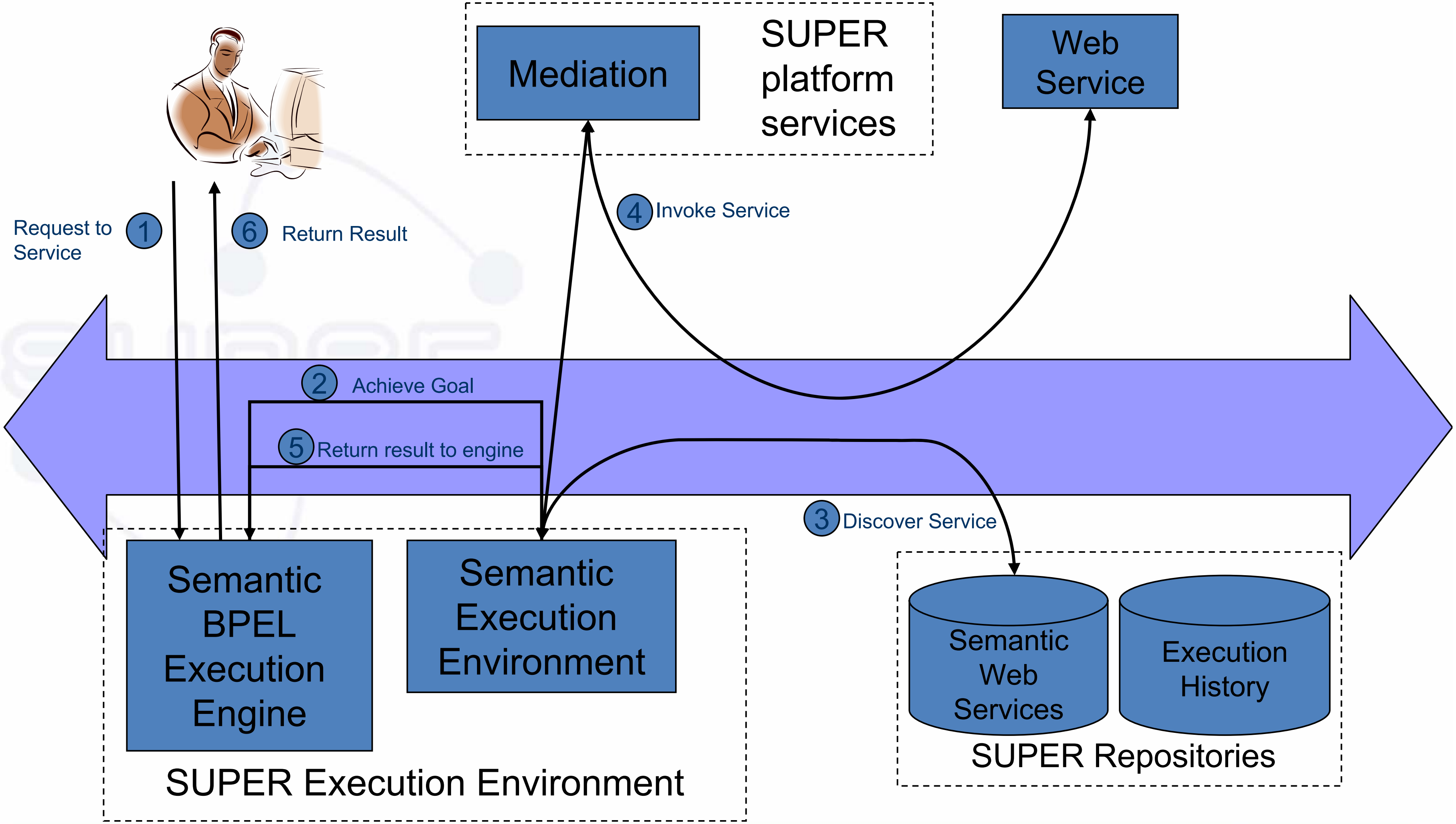
Deployment Process: Semantic Process Artefacts Bundle (SPAB)



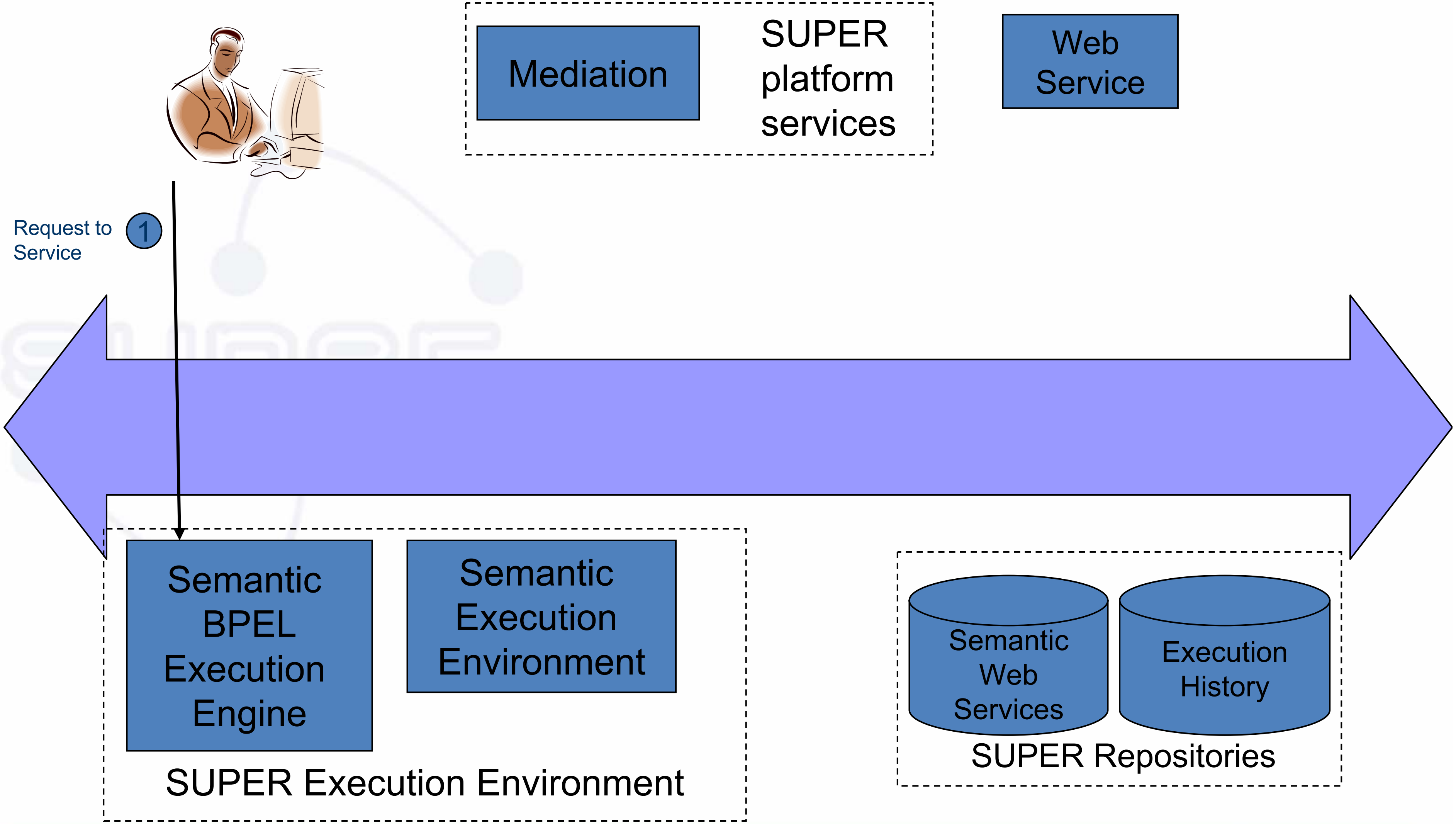
Deployment Process



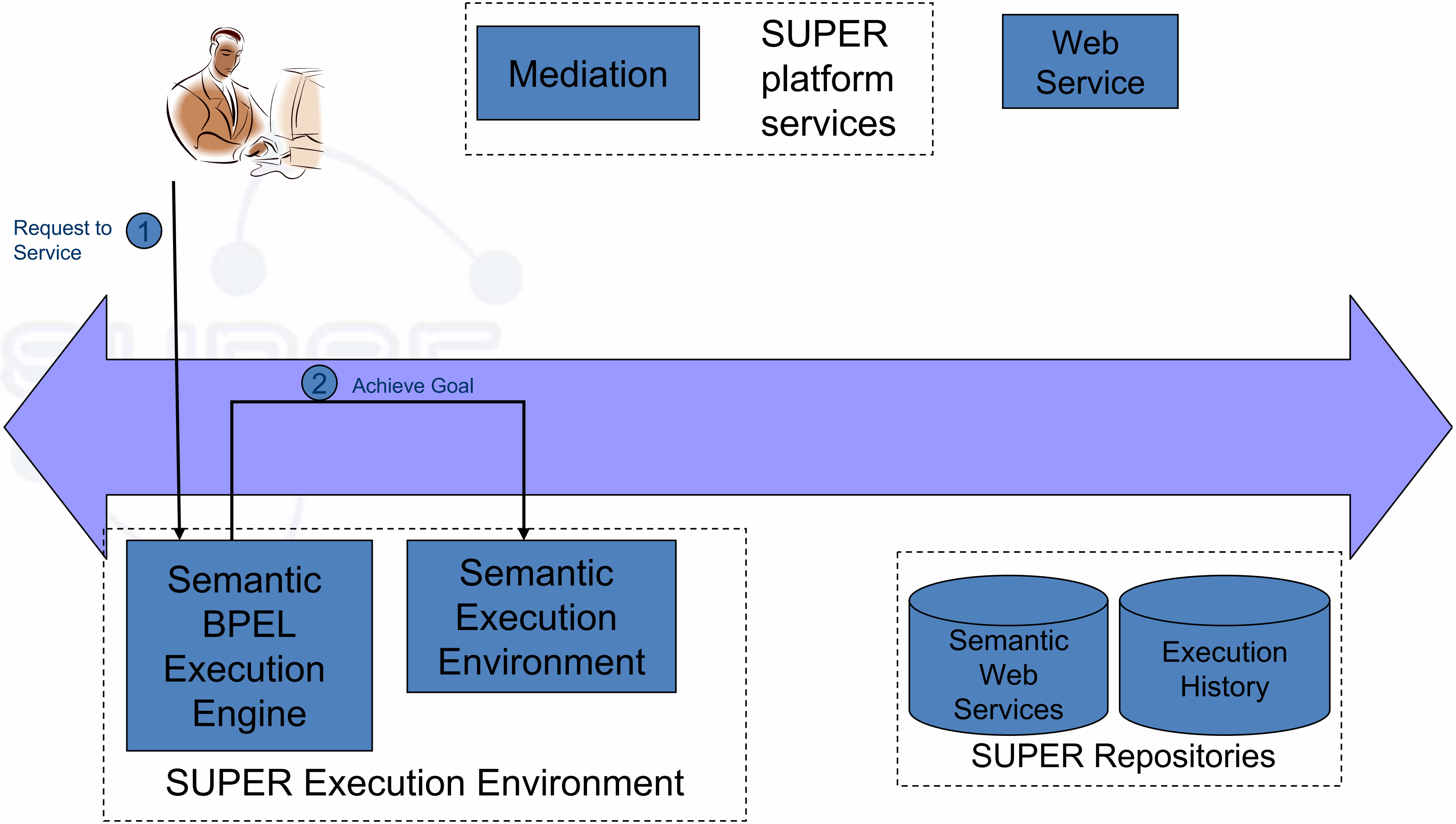
Semantic Business Process Execution



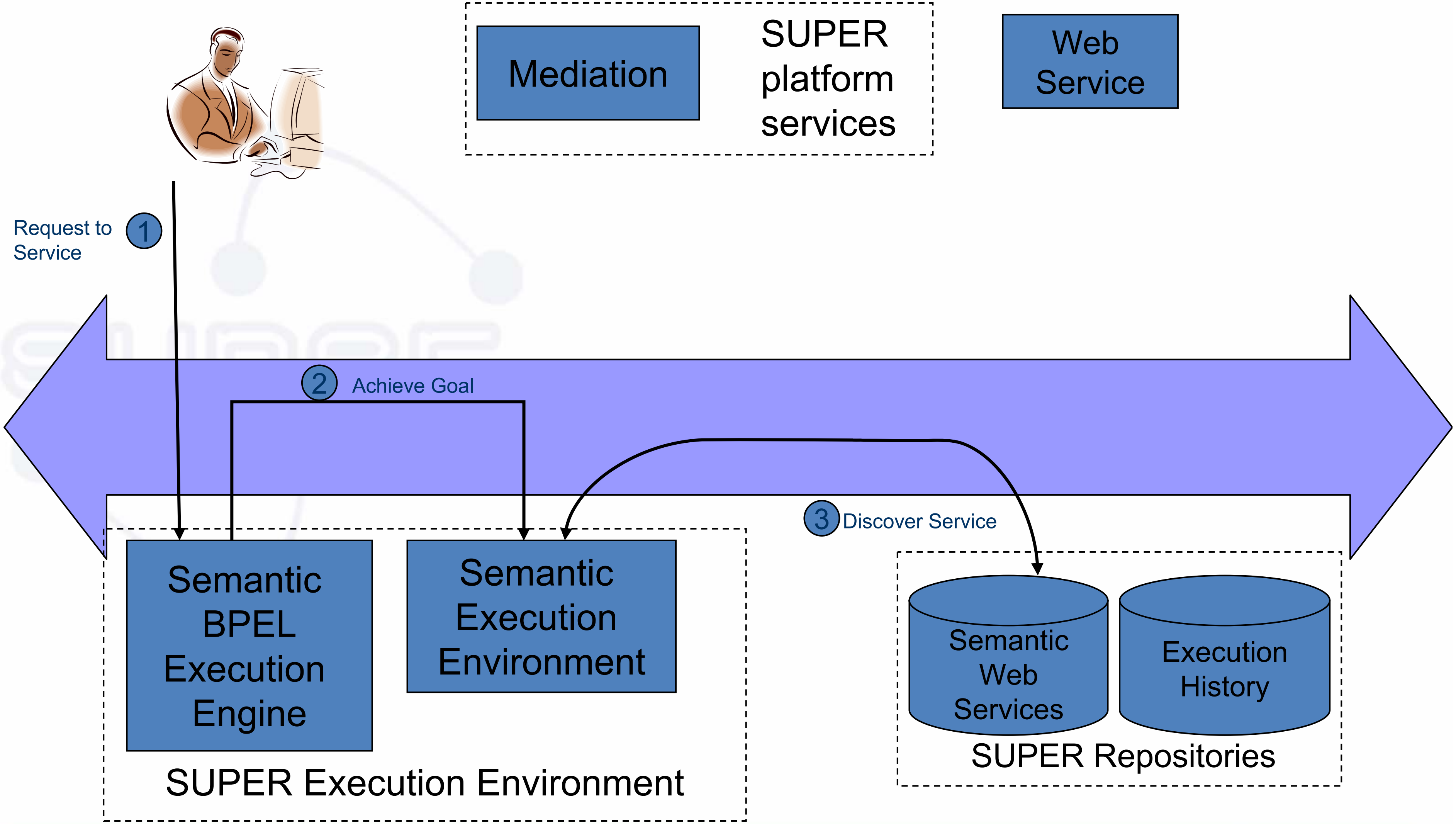
Semantic Business Process Execution



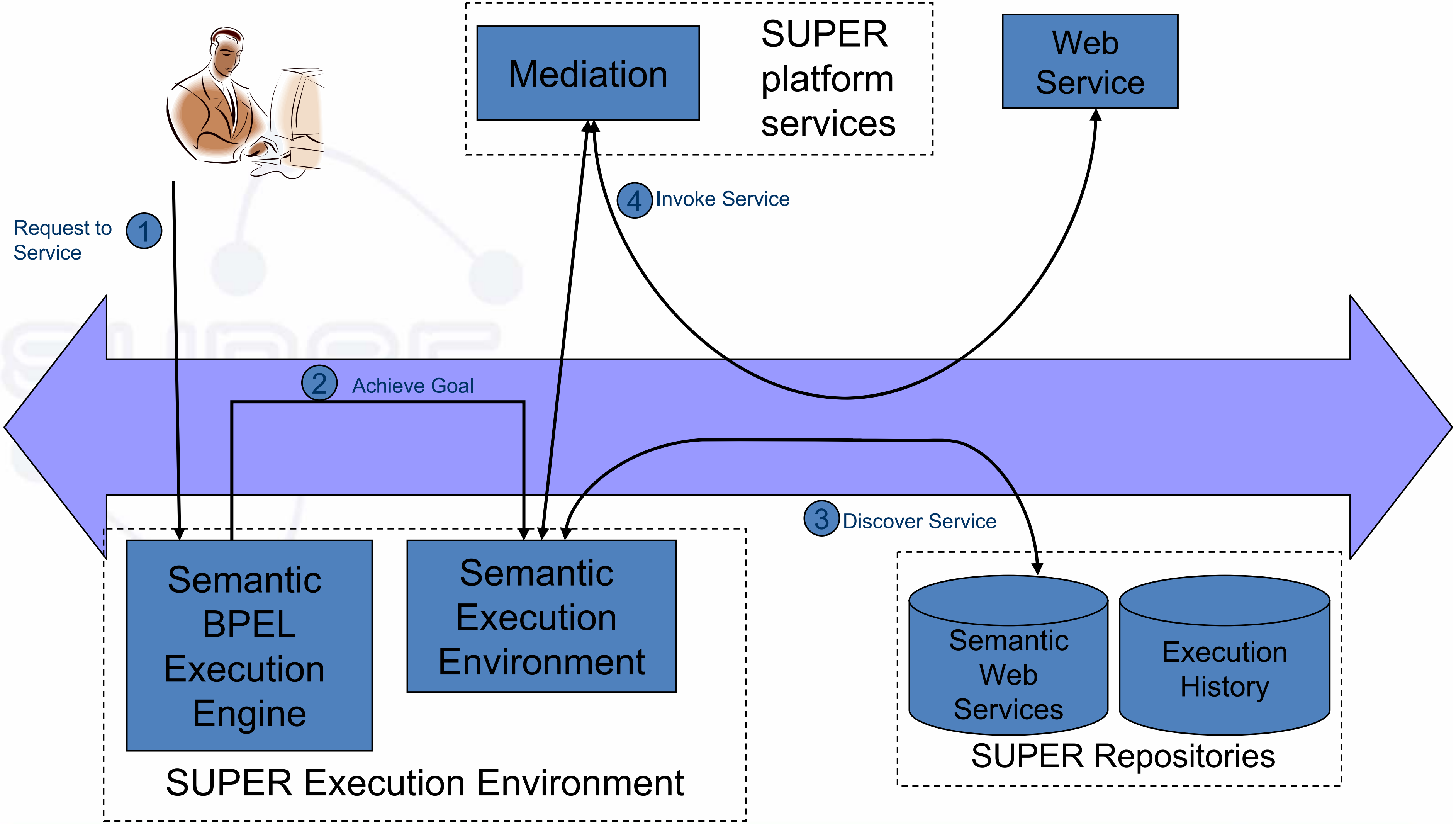
Semantic Business Process Execution



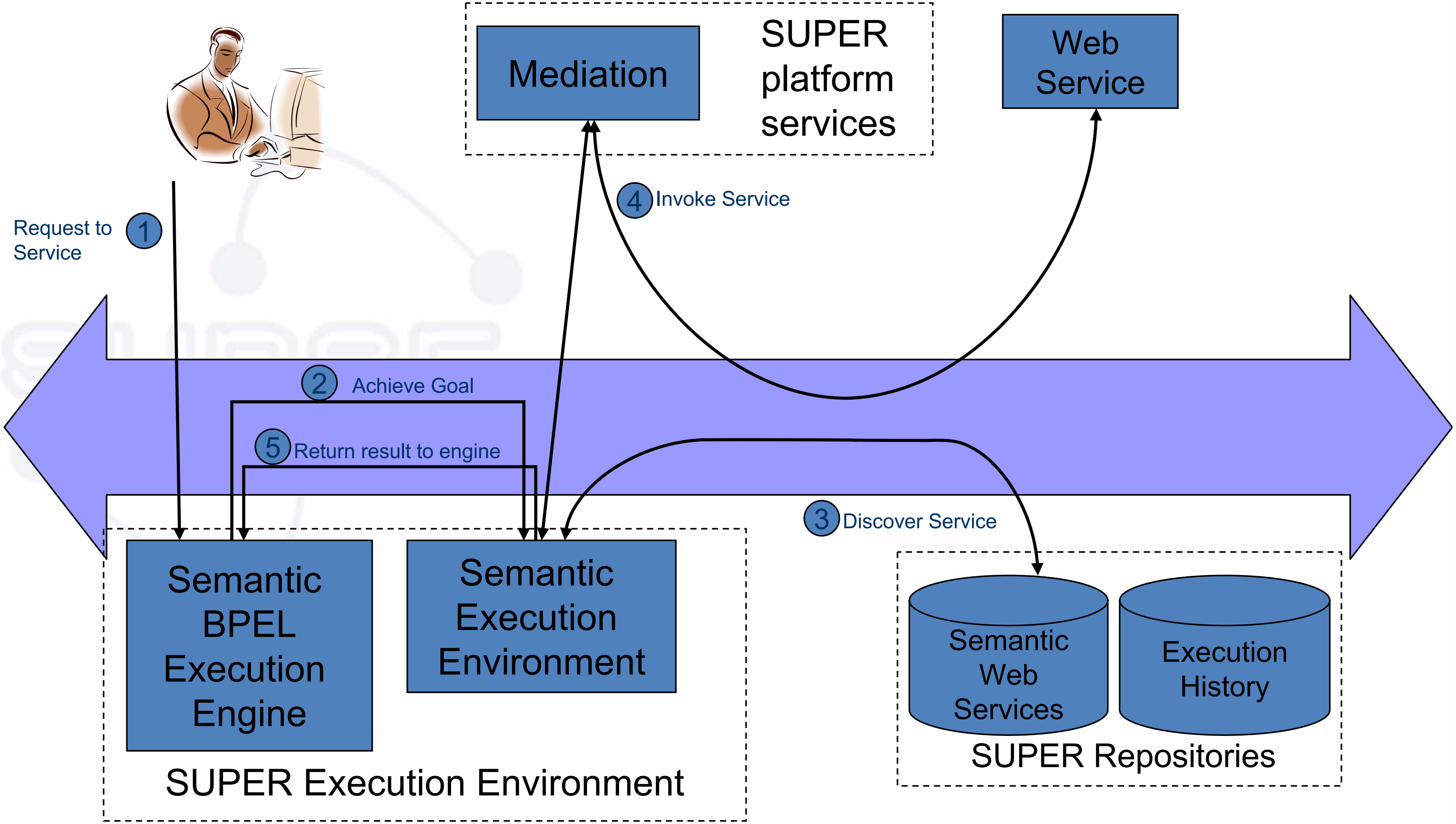
Semantic Business Process Execution



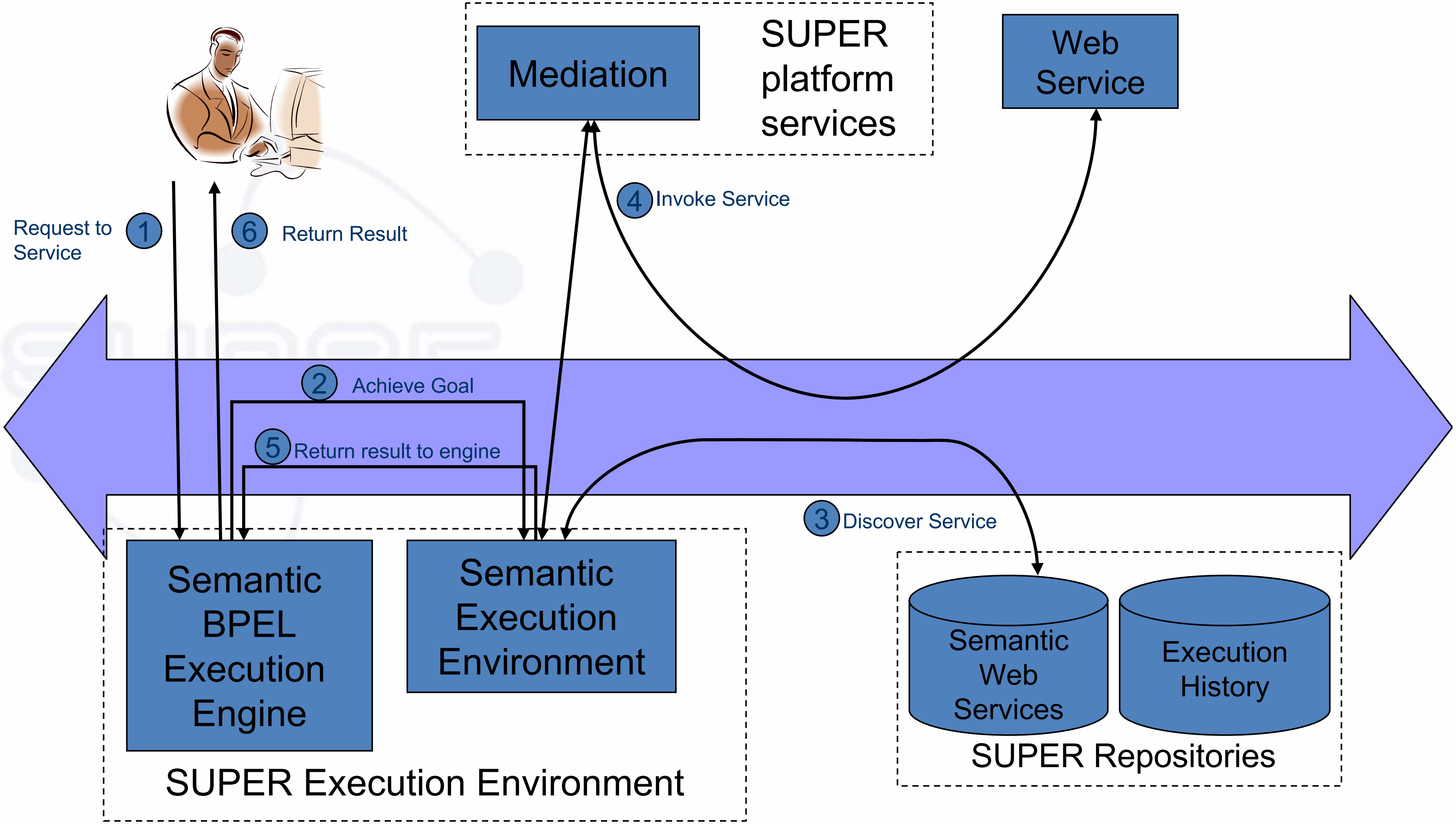
Semantic Business Process Execution



Semantic Business Process Execution



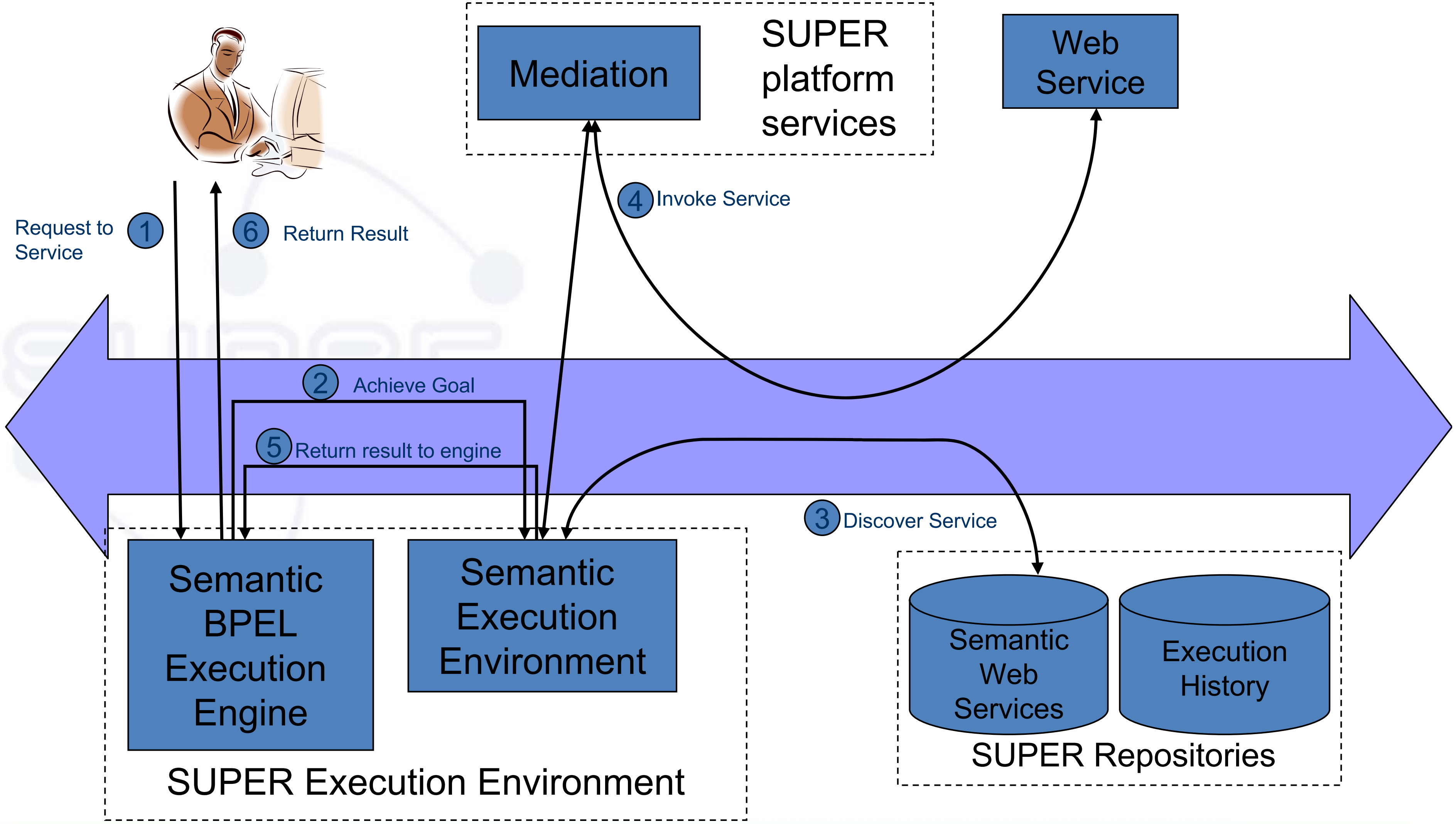
Semantic Business Process Execution



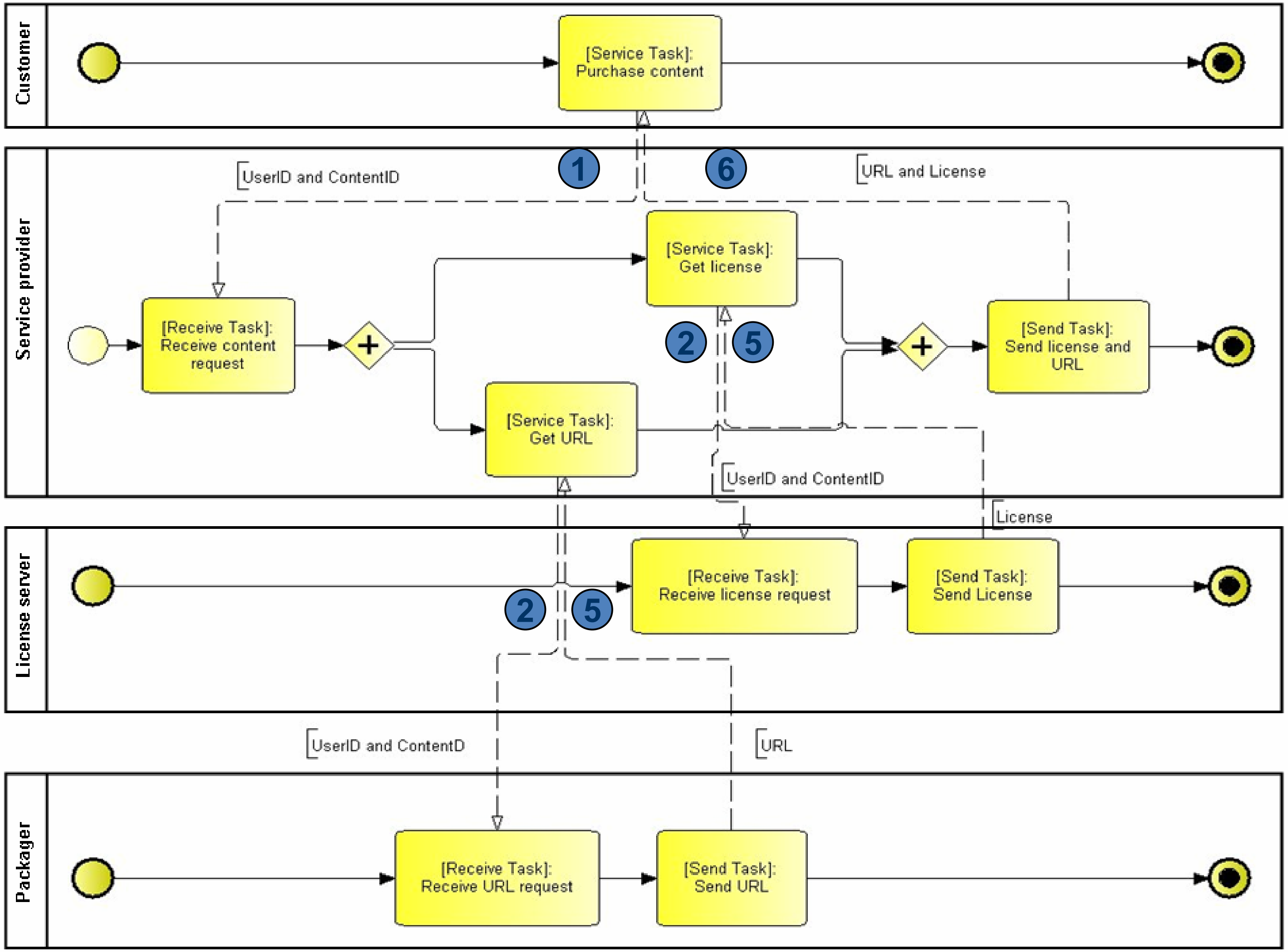
Semantic Business Process Execution on the SUPER Infrastructure

– DEMONSTRATION –

Semantic Business Process Execution



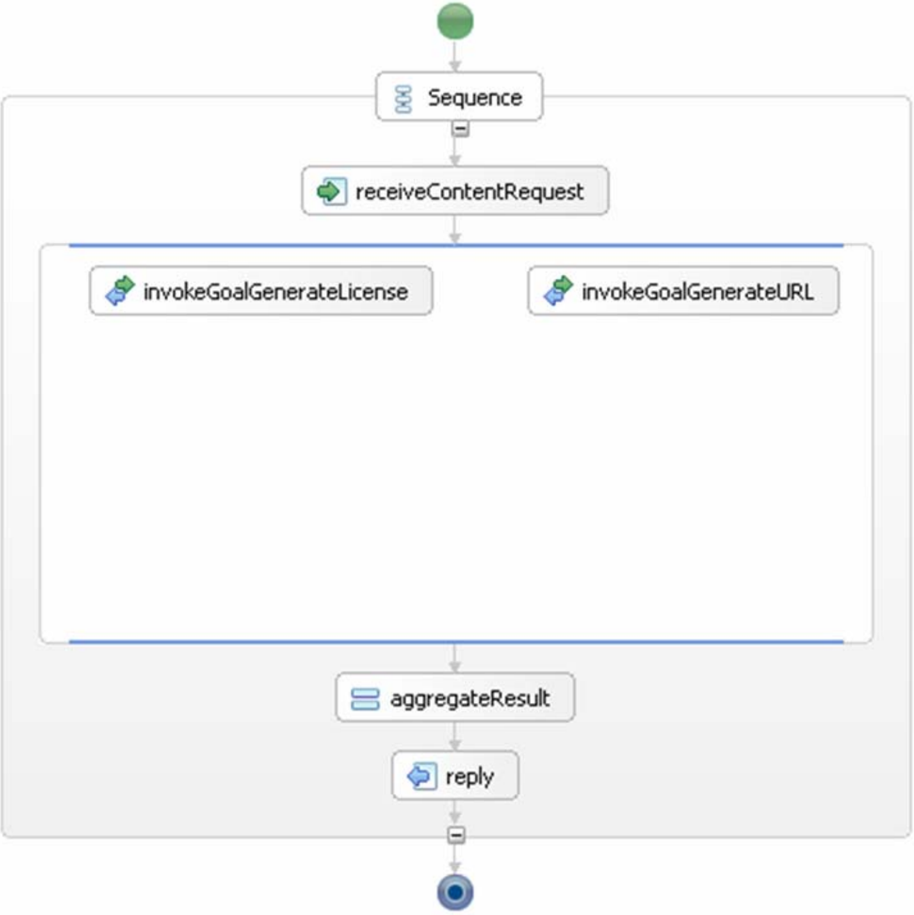
Purchase Content Process


Customer
Service provider
License server
Packager
[Service Task]: Purchase content
UserID and ContentID
URL and License
1
6
[Receive Task]: Receive content request
[Service Task]: Get license
[Service Task]: Get URL
[Send Task]: Send license and URL
2
5
UserID and ContentID
License
[Receive Task]: Receive license request
[Send Task]: Send License
2
5
UserID and ContentD
URL
[Receive Task]: Receive URL request
[Send Task]: Send URL

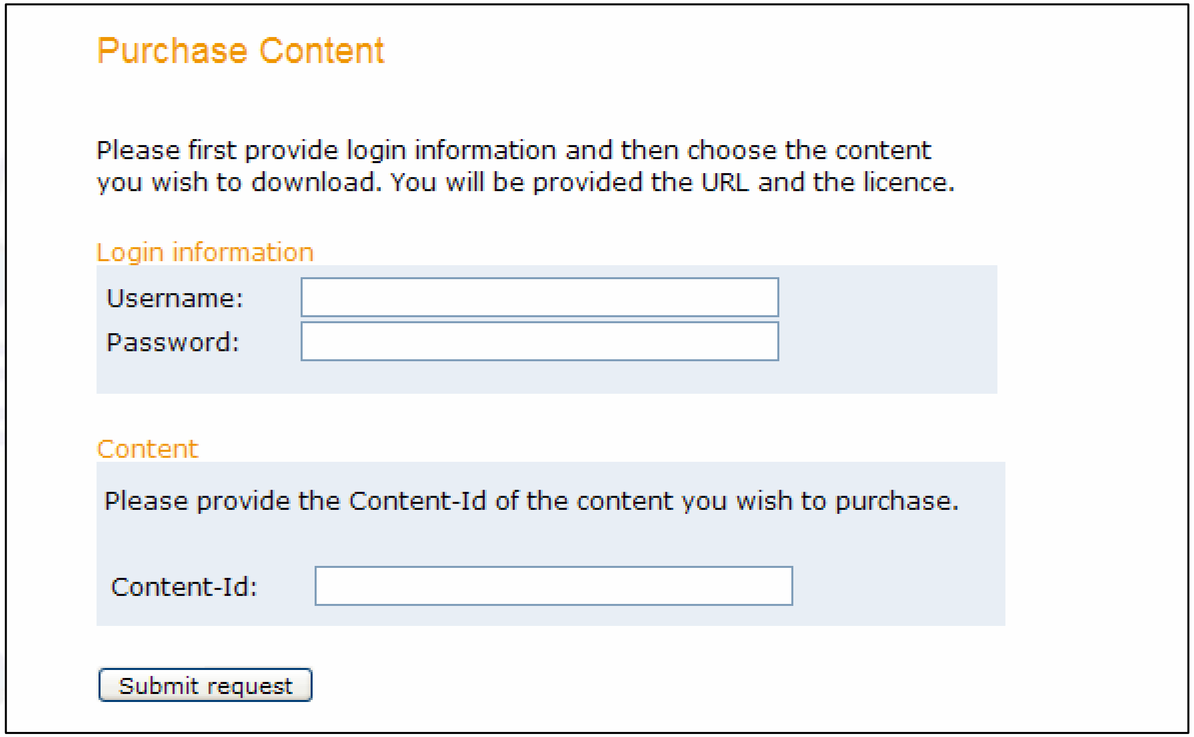
BPEL4SWS Process (2)

The Content

Purchase Process



Demonstration - Process Client



Purchase Content

Please first provide login information and then choose the content you wish to download. You will be provided the URL and the licence.

Login information

Username:

Password:

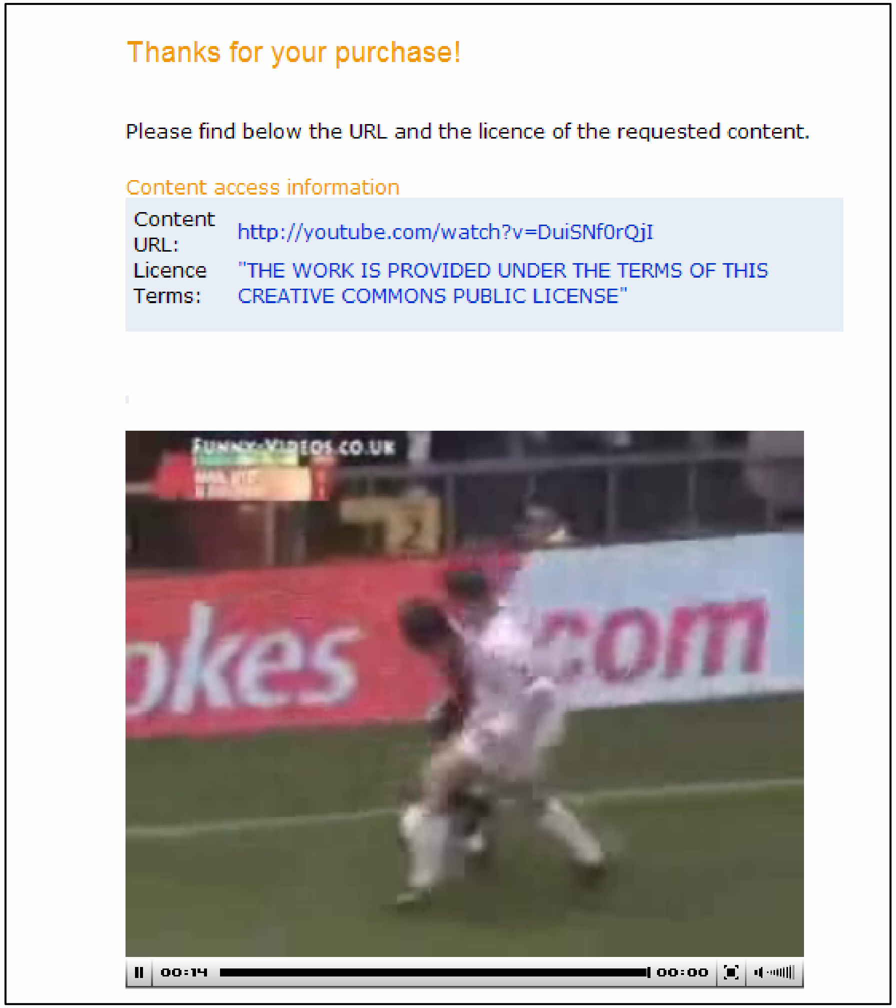
Content

Please provide the Content-Id of the content you wish to purchase.

Content-Id:

Demonstration – Process Response

Result of the Execution of the Purchase Content Process



Thanks for your purchase!

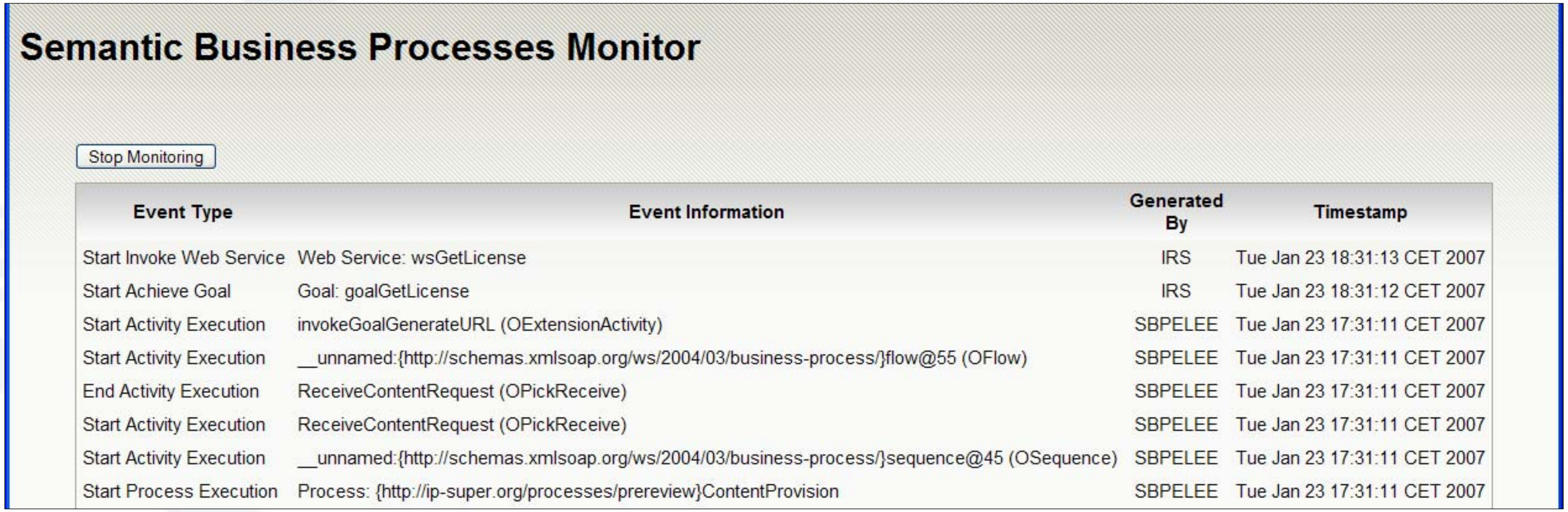
Please find below the URL and the licence of the requested content.

Content access information

Content URL:	http://youtube.com/watch?v=DuiSNf0rQjI
Licence Terms:	"THE WORK IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE"

Monitoring - Process Events

- Semantic Business Process Monitoring

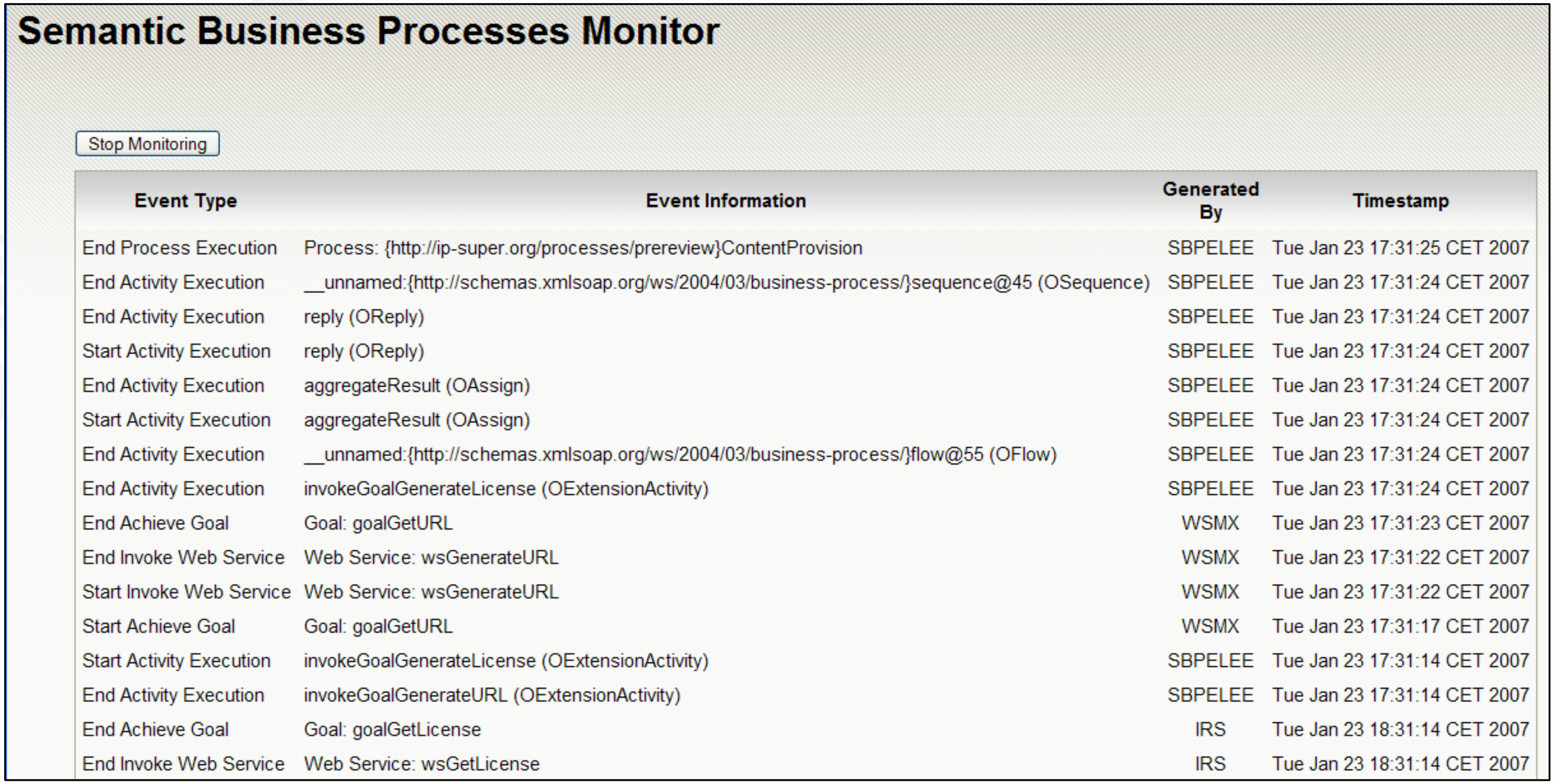


Semantic Business Processes Monitor

Event Type	Event Information	Generated By	Timestamp
Start Invoke Web Service	Web Service: wsGetLicense	IRS	Tue Jan 23 18:31:13 CET 2007
Start Achieve Goal	Goal: goalGetLicense	IRS	Tue Jan 23 18:31:12 CET 2007
Start Activity Execution	invokeGoalGenerateURL (OExtensionActivity)	SBPELEE	Tue Jan 23 17:31:11 CET 2007
Start Activity Execution	__unnamed:{http://schemas.xmlsoap.org/ws/2004/03/business-process/}flow@55 (OFlow)	SBPELEE	Tue Jan 23 17:31:11 CET 2007
End Activity Execution	ReceiveContentRequest (OPickReceive)	SBPELEE	Tue Jan 23 17:31:11 CET 2007
Start Activity Execution	ReceiveContentRequest (OPickReceive)	SBPELEE	Tue Jan 23 17:31:11 CET 2007
Start Activity Execution	__unnamed:{http://schemas.xmlsoap.org/ws/2004/03/business-process/}sequence@45 (OSequence)	SBPELEE	Tue Jan 23 17:31:11 CET 2007
Start Process Execution	Process: {http://ip-super.org/processes/prereview}ContentProvision	SBPELEE	Tue Jan 23 17:31:11 CET 2007

Monitoring - Process Events (2)

- Semantic Business Process Monitoring



Event Type	Event Information	Generated By	Timestamp
End Process Execution	Process: {http://ip-super.org/processes/prereview}ContentProvision	SBPELEE	Tue Jan 23 17:31:25 CET 2007
End Activity Execution	__unnamed:{http://schemas.xmlsoap.org/ws/2004/03/business-process/}sequence@45 (OSequence)	SBPELEE	Tue Jan 23 17:31:24 CET 2007
End Activity Execution	reply (OReply)	SBPELEE	Tue Jan 23 17:31:24 CET 2007
Start Activity Execution	reply (OReply)	SBPELEE	Tue Jan 23 17:31:24 CET 2007
End Activity Execution	aggregateResult (OAssign)	SBPELEE	Tue Jan 23 17:31:24 CET 2007
Start Activity Execution	aggregateResult (OAssign)	SBPELEE	Tue Jan 23 17:31:24 CET 2007
End Activity Execution	__unnamed:{http://schemas.xmlsoap.org/ws/2004/03/business-process/}flow@55 (OFlow)	SBPELEE	Tue Jan 23 17:31:24 CET 2007
End Activity Execution	invokeGoalGenerateLicense (OExtensionActivity)	SBPELEE	Tue Jan 23 17:31:24 CET 2007
End Achieve Goal	Goal: goalGetURL	WSMX	Tue Jan 23 17:31:23 CET 2007
End Invoke Web Service	Web Service: wsGenerateURL	WSMX	Tue Jan 23 17:31:22 CET 2007
Start Invoke Web Service	Web Service: wsGenerateURL	WSMX	Tue Jan 23 17:31:22 CET 2007
Start Achieve Goal	Goal: goalGetURL	WSMX	Tue Jan 23 17:31:17 CET 2007
Start Activity Execution	invokeGoalGenerateLicense (OExtensionActivity)	SBPELEE	Tue Jan 23 17:31:14 CET 2007
End Activity Execution	invokeGoalGenerateURL (OExtensionActivity)	SBPELEE	Tue Jan 23 17:31:14 CET 2007
End Achieve Goal	Goal: goalGetLicense	IRS	Tue Jan 23 18:31:14 CET 2007
End Invoke Web Service	Web Service: wsGetLicense	IRS	Tue Jan 23 18:31:14 CET 2007

References

References BPM Foundations

[Herzum and Sims 2000] Herzum, P.; Sims, O. (2000): Business Component Factory. John Wiley & Sons.

[McGovern et al. 2006] McGovern, J.; Sims, O.; Jain, A.; Little, M. (2006): Enterprise Service Oriented Architectures. Springer, Dordrecht, Netherlands.

[Scheer et al. 2005] Scheer, A.-W.; Thomas, O.; Adam, O. (2005): Process modelling using event-driven process chains. In: Dumas, M.; van der Aalst, W. M. P.; ter Hofstede, A. H. M. (Eds.): Process-Aware Information Systems. Wiley, Hoboken, New Jersey, USA, pp. 119–146.

[Smith and Fingar 2003] Smith, H.; Fingar, P. (2003): Business Process Management: The Third Wave. Meghan-Kiffer Press, Tampa, FL, USA, 1st edition.

[Specht et al, 2005] T. Specht, J. Drawehn, M. Thränert, and S. Kühne. Modeling cooperative business processes and transformation to a service oriented architecture. In 7th International IEEE Conference on E-Commerce Technology, 2005.

[Teboul 2006] Teboul, J. (2006): Service Is Front Stage: Positioning Services for Value Advantage. Palgrave Macmillan, New York, USA.

References BPEL & BPM Technologies

[BEPL 1.1] Andrews, T.; Curbera, F.; Dholakia, H.; Goland, Y.; Klein, J.; Leymann, F.; Liu, K.; Roller, D.; Smith, D.; Thatte, S.; Trickovic, I.; Weerawarana S.: *Business Process Execution Language for Web Services (BPEL4WS) 1.1*. Technical Report, OASIS, 2003.

[OASIS SOA] MacKenzie, C. M.; Laskey, K.; McCabe, F.; Brown, P. F.; Metz, R. : *Reference Model for Service Oriented Architecture 1.0*. Technical Report, OASIS. 2006.

[Aalst et al., 2003] Wil M. P. van der Aalst, A. H. M. ter Hofstede, B. Kiepuszewski, and A. P. Barros: Workflow Patterns. In *Distributed and Parallel Databases*, 14(3):5–51, July 2003.

[Mendling and Nüttgens, 2006] J. Mendling and M. Nüttgens. EPC Markup Language (EPML) - an xml-based interchange format for event-driven process chains (EPC). In *International Journal Information Systems and e-Business Management (ISeB)*, 4(3):245–263, July 2006.

[Wohed et al, 2002] P. Wohed, Wil M. P. van der Aalst, M. Dumas, and A. H. M. ter Hofstede. *Pattern-based analysis of BPEL4WS*. Technical Report FIT-TR-2002-04, Queensland University of Technology, Brisbane, Australia, 2002.

[Kopp et al, 2006] Oliver Kopp, Tobias Unger, and Frank Leymann. Nautilus event-driven process chains: Syntax, semantics, and their mapping to bpel. In *Proceedings of the 5th GI Workshop on Event-Driven Process Chains (EPK 2006)*, pages 85–104, Vienna, Austria, 2006.

[Manddell and McIllraith, 2003] Daniel J. Mandell and Sheila A. McIlraith. Adapting BPEL4WS for the Semantic Web: The Bottom-Up Approach to Web Service Interoperation. In Proceedings of the Second International Semantic Web Conference (ISWC2003),

References SWS Foundations

[Alonso et al., 2004] Alonso, G., Casati, F., Kuno, H., and Machiraju, V. (2004). Web Services: Concepts, Architectures and Applications. Data-Centric Systems and Applications. Springer, Berlin, Heidelberg.

[Berners-Lee, 1999] Berners-Lee, T. (1999). Weaving the Web. Harper, San Francisco, USA.

[Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The Semantic Web. Scientific American, 284(5):34-43.

[Bussler, 2003] Bussler, C. (2003). B2B Integration: Concepts and Architecture. Springer, Berlin, Heidelberg.

[Fensel, 2003] Fensel, D. (2003). Ontologies: A Silver Bullet for Knowledge Management and E-Commerce. Springer, Berlin, Heidelberg, 2 edition.

[Fensel et al., 2006] Fensel, D. et al: Enabling Semantic Web Services. The Web Service Modeling Ontology WSMO. Springer, Berlin, Heidelberg, 2006.

[Goméz-Pérez et al., 2003] Goméz-Pérez, A., Corcho, O., and Fernandez-Lopez, M. (2003). Ontological Engineering. With Examples from the Areas of Knowledge Management, E-Commerce and Semantic Web. Series of Advanced Information and Knowledge Processing. Springer, Berlin, Heidelberg.

[Gruber, 1993] Gruber, T. R. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition, 5:199-220.

References Semantic Web Services

[de Bruijn et al., 2006] de Bruijn, J., Fensel, D., Lausen, H., Polleres, A., Roman, D., and Stollberg, M. (2006). Enabling Semantic Web Services. The Web Service Modeling Ontology. Springer.

[Fensel and Bussler, 2002] Fensel, D. and Bussler, C. (2002). The Web Service Modeling Framework WSMF. Electronic Commerce Research and Applications, 1(2).

[McIlraith et al., 2001] McIlraith, S., Cao Son, T., and Zeng, H. (2001). Semantic Web Services. IEEE Intelligent Systems, Special Issue on the Semantic Web, 16(2):46-53.

[Preist, 2004] Preist, C. (2004). A Conceptual Architecture for Semantic Web Services. In Proc. of the Int. Semantic Web Conf. (ISWC 2004).

[Roman et al., 2005] Roman, D., Keller, U., Lausen, H., de Bruijn, J., Lara, R., Stollberg, M., Polleres, A., Feier, C., Bussler, C., and Fensel, D. (2005). Web Service Modeling Ontology. Applied Ontology, 1(1):77-106.

[Stollberg et al., 2006] Stollberg, M., Feier, C., Roman, D., and Fensel, D. (2006). Semantic Web Services - Concepts and Technology. In Ide, N., Cristea, D., and Tufis, D. (editors), Language Technology, Ontologies, and the Semantic Web. Kluwer Publishers.

[Sycara et al. 2003] Katia Sycara, Massimo Paolucci, Anupriya Ankolekar and Naveen Srinivasan, "Automated Discovery, Interaction and Composition of Semantic Web services," Journal of Web Semantics, Volume 1, Issue 1, September 2003, pp. 27-46

References SWS: W3C Submissions

OWL-S

[Martin, 2004] Martin, D. (2004). OWL-S: Semantic Markup for Web Services. W3C Member Submission 22 November 2004. online: http://www.w3.org/Submission/OWL-S/.

WSMO [see also www.wsmo.org]

[Lausen et al., 2005] Lausen, H., Polleres, A., and Roman (eds.), D. (2005). Web Service Modeling Ontology (WSMO). W3C Member Submission 3 June 2005. online: http://www.w3.org/Submission/WSMO/.

SWSF

[Battle et al., 2005] Battle, S., Bernstein, A., Boley, H., Grosof, B., Gruninger, M., Hull, R., Kifer, M., D., M., S., M., McGuinness, D., Su, J., and Tabet, S. (2005). Semantic Web Services Framework (SWSF). W3C Member Submission 9 September 2005. online: http://www.w3.org/Submission/SWSF/.

WSDL-S

[Akkiraju et al., 2005] Akkiraju, R., Farrell, J., Miller, J., Nagarajan, M., Schmidt, M.- T., Sheth, A., and Verma, K. (2005). Web Service Semantics - WSDL-S. W3C Member Submission 7 November 2005. online: http://www.w3.org/Submission/WSDL-S/.

References Discovery

Stollberg, M. Stollberg, M.; Keller, U.; Lausen, H. and Heymans, S.: **Two-phase Web Service Discovery based on Rich Functional Descriptions.** In Proc. of the 4th European Semantic Web Conference (ESWC 2007), Innsbruck, Austria.

B. Benatallah, M. Hacid, C. Rey, F. Toumani **Towards Semantic Reasoning for Web Services Discovery**,. *In Proc. of the International Semantic Web Conference (ISWC 2003), 2003*

Keller, U.; Lara, R.; Lausen, H.; Polleres, A.; Fensel, D.: **Automatic Location of Services**. In Proc. of the 2nd European Semantic Web Symposium (ESWS2005), Heraklion, Crete, 2005.

M. Kifer, R. Lara, A. Polleres, C. Zhao, U. Keller, H. Lausen and D. Fensel: **A Logical Framework for Web Service Discovery.** Proc. 1st. Intl. Workshop SWS'2004 at ISWC 2004,Hiroshima, Japan, November 8, 2004, CEUR Workshop Proceedings, ISSN 1613-0073

Lei Li and Ian Horrocks. **A software framework for matchmaking based on semantic web technology**. In *Proc. of the Twelfth International World Wide Web Conference (WWW 2003), 2003.*

Massimo Paolucci, Takahiro Kawamura, Terry R. Payne, Katia Sycara; **Semantic Matching of Web Services Capabilities**. In Proceedings of the 1st International Semantic Web Conference (ISWC2002), 2002

References Composition

[Berardi et al., 2003] Berardi, D., Calvanese, D., Giacomo, G. D., Lenzerini, M., and Mecella, M. (2003). Automatic Composition of e-Services that Export their Behavior. In Proc. of First Int. Conference on Service Oriented Computing (ICSOC).

[Martens, 2003] Martens, A. (2003). On Compatibility of Web Services. Petri Net Newletter, 65:12-20.

[Sirin et al., 2004] Sirin, E., Parsia, B., Wu, D., Hendler, J., and Nau, D. (2004). HTN Planning for Web Service Composition Using SHOP2. Journal of Web Semantics, 1(4):377-396.

[Pistore and Traverso, 2006] Pistore, M. and Traverso, P. (2006). Theoretical Integration of Discovery and Composition. Deliverable D2.4.6, Knowledge Web.

[Stollberg, 2005] Stollberg, M. (2005). Reasoning Tasks and Mediation on Choreography and Orchestration in WSMO. In Proceedings of the 2nd International WSMO Implementation Workshop (WIW 2005), Innsbruck, Austria.

[Traverso and Pistore, 2004] Traverso, P. and Pistore, M. (2004). Automatic Composition of Semantic Web Services into Executable Processes. In Proc. 3rd International Semantic Web Conference (ISWC 2004), Hiroshima, Japan.

References Mediation

[Cimpian and Mocan, 2005] Cimpian, E. and Mocan, A. (2005). WSMX Process Mediation Based on Choreographies. In Proceedings of the 1st International Workshop on Web Service Choreography and Orchestration for Business Process Management at the BPM 2005, Nancy, France.

[Mocan (ed.), 2005] Mocan (ed.), A. (2005). WSMX Data Mediation. WSMX Working Draft D13.3. available at: http://www.wsmo.org/TR/d13/d13.3/v0.2/.

[Mocan et al., 2005] Mocan, A., Cimpian, E., Stollberg, M., Scharffe, F., and Scicluna, J. (2005). WSMO Mediators. WSMO deliverable D29 ¯nal draft 21 Dec 2005. available at: http://www.wsmo.org/TR/d29/.

[Scharffe and de Bruijn, 2005] Scharffe, F. and de Bruijn, J. (2005). A language to specify mappings between ontologies. In Proc. of the Internet Based Systems IEEE Conference (SITIS05).

[Stollberg et al., 2006] Stollberg, M., Cimpian, E., Mocan, A., and Fensel, D. (2006). A Semantic Web Mediation Architecture. In Proceedings of the 1st Canadian Semantic Web Working Symposium (CSWWS 2006), Quebec, Canada.

[Wiederhold, 1994] Wiederhold, G. (1994). Mediators in the Architecture of the Future Information Systems. Computer, 25(3):38-49.

References WSMO

- The central location where WSMO work and papers can be found is WSMO Working Group: http://www.wsmo.org

- WSMO languages – WSML Working Group: http://www.wsml.org

- WSMO implementation
  - WSMX working group : http://www.wsmx.org
  - WSMX open source can be found at: https://sourceforge.net/projects/wsmx/

References IRS III

J. Domingue, L. Cabral, F. Hakimpour, D. Sell and E. Motta: IRS-III: A Platform and Infrastructure for Creating WSMO-based Semantic Web Services. Proceedings of the Workshop on WSMO Implementations (WIW 2004) Frankfurt, Germany, September 29-30, 2004, CEUR Workshop Proceedings, ISSN 1613-0073, online http://CEUR-WS.org/Vol-113/paper3.pdf.

J. Domingue and S. Galizia: Towards a Choreography for IRS-III.

Proceedings of the Workshop on WSMO Implementations (WIW 2004) Frankfurt, Germany, September 29-30, 2004, CEUR Workshop Proceedings, ISSN 1613-0073, online http://CEUR-WS.org/Vol-113/paper7.pdf.

Cabral, L., Domingue, J., Motta, E., Payne, T. and Hakimpour, F. (2004).

Approaches to Semantic Web Services: An Overview and Comparisons. In proceedings of the First European Semantic Web Symposium (ESWS2004);

10-12 May 2004, Heraklion, Crete, Greece.

Motta, E., Domingue, J., Cabral, L. and Gaspari, M. (2003) IRS-II: A Framework and Infrastructure for Semantic Web Services. In proceedings of the 2nd International Semantic Web Conference (ISWC2003) 20-23 October 2003, Sundial Resort, Sanibel Island, Florida, USA.

These papers and software downloads can be found at: http://kmi.open.ac.uk/projects/irs