

Web Service Modeling Ontology (WSMO) - An Ontology for Semantic Web Services

**John Domingue, Dumitru Roman, and Michael Stollberg
(On behalf of the WSMO Working Group)**

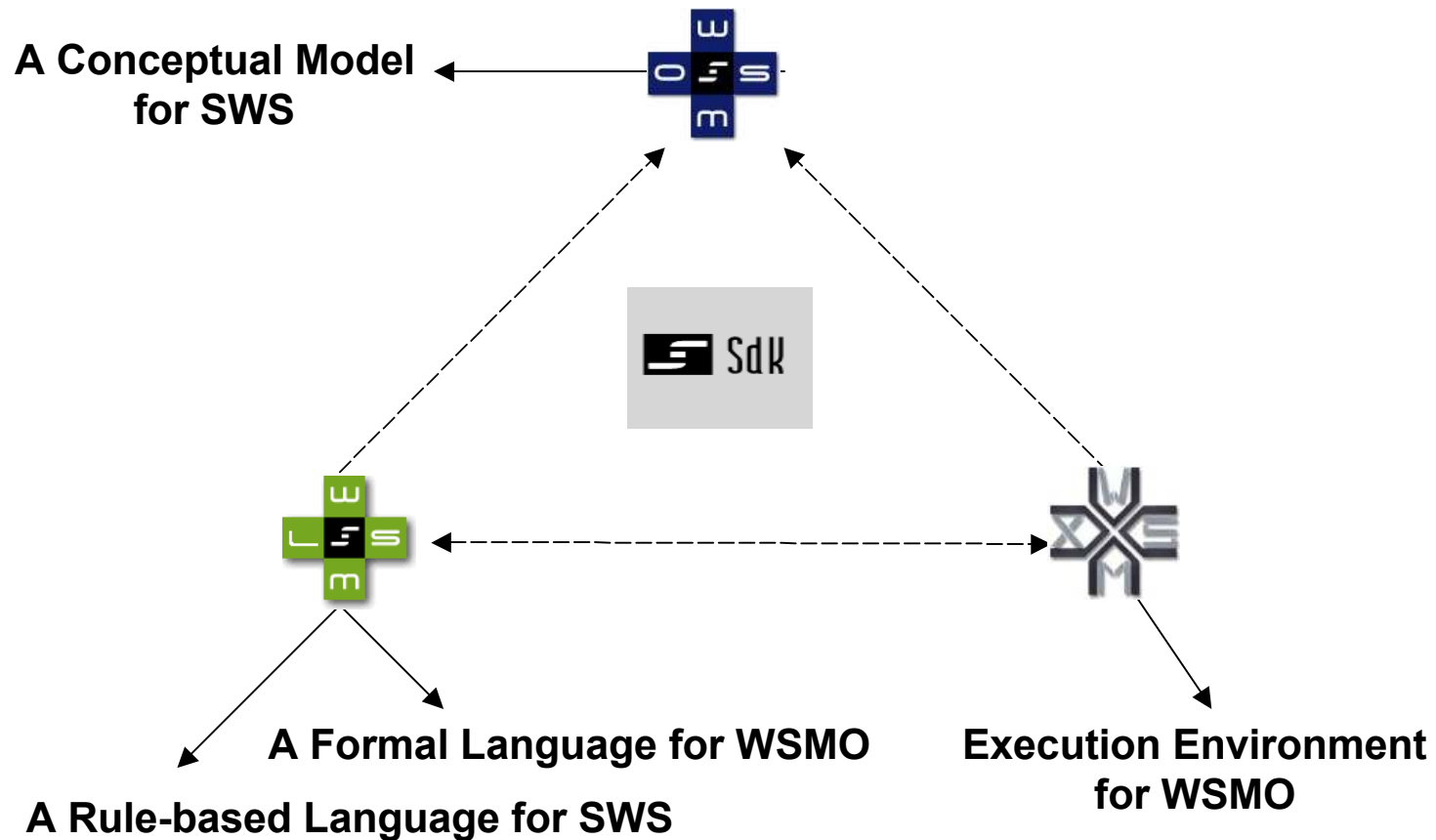
**The WSMO Working Group Co-Chairs are:
Christoph Bussler, John Domingue, and Dieter Fensel**

Contents

- Thanks to W3C
- Background
- WSMO overview
- W3C comments and our replies
- Demo
- Summary



WSMO Working Groups



Underlying Principles

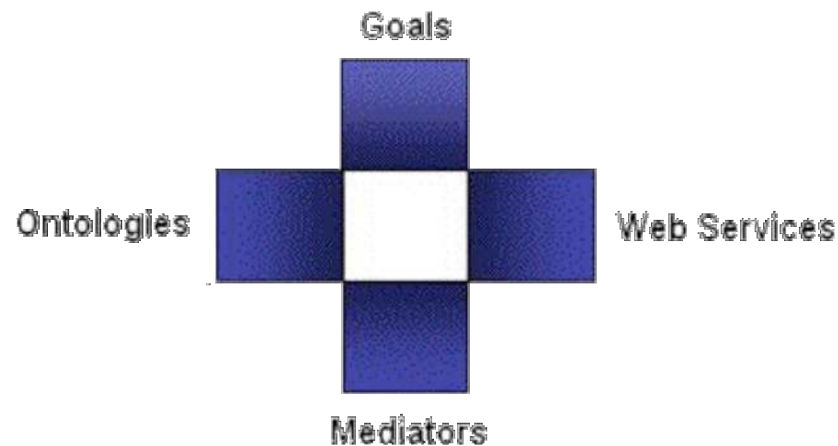
- Web Compliance
- Ontology based
- Strict decoupling
- Centrality of mediation
- Ontological role separation
- Description vs Implementation
- Execution Semantics
- Service vs Web Service



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



Non-Functional Properties

- Dublin Core Metadata Set:
 - complete item description
 - used for resource management
- Quality of Service Information
 - availability, stability
- Other
 - Versioning, Owner, financial



Non-Functional Properties List

Dublin Core Metadata

Contributor
Coverage
Creator
Description
Format
Identifier
Language
Publisher
Relation
Rights
Source
Subject
Title
Type

Quality of Service

Accuracy
NetworkRelatedQoS
Performance
Reliability
Robustness
Scalability
Security
Transactional
Trust

Other

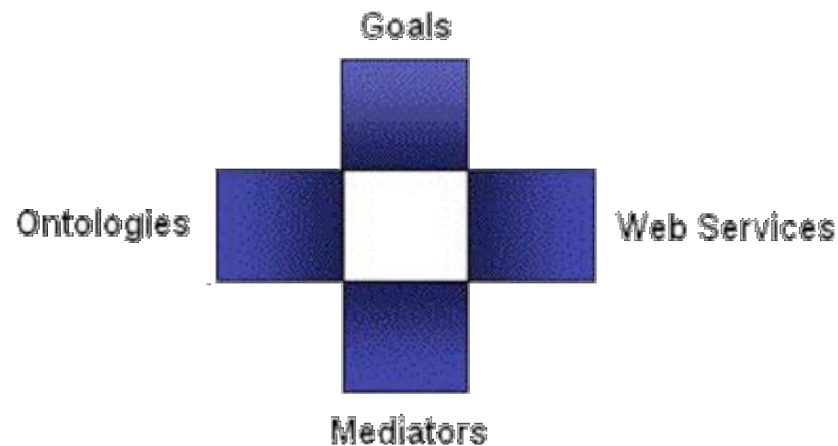
Financial
Owner
TypeOfMatch
Version



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



Ontology Specification

- **Non functional properties** (see before)
- **Imported Ontologies** importing existing ontologies where no heterogeneities arise
- **Used mediators** OO Mediators (ontology import with terminology mismatch handling)

Ontology Elements:

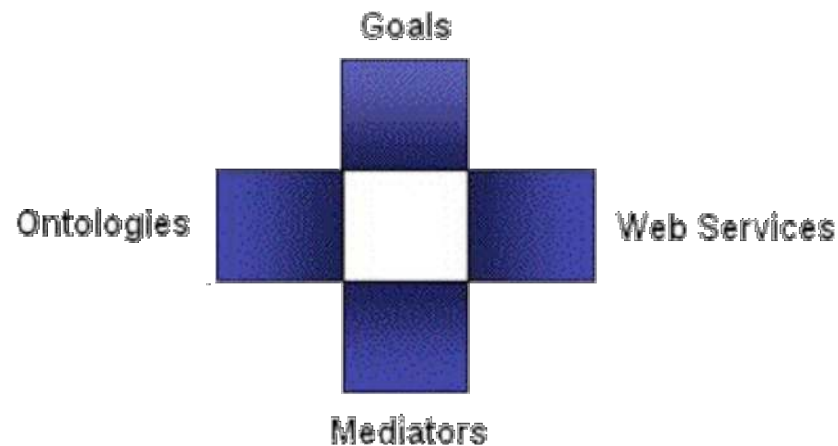
- Concepts** set of concepts that belong to the ontology, incl.
- Attributes** set of attributes that belong to a concept
- Relations** define interrelations between several concepts
- Functions** special type of relation (unary range = return value)
- Instances** set of instances that belong to the represented ontology
- Axioms** axiomatic expressions in ontology (logical statement)



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



WSMO Web Service Description

- complete item description
- quality aspects
- Web Service Management

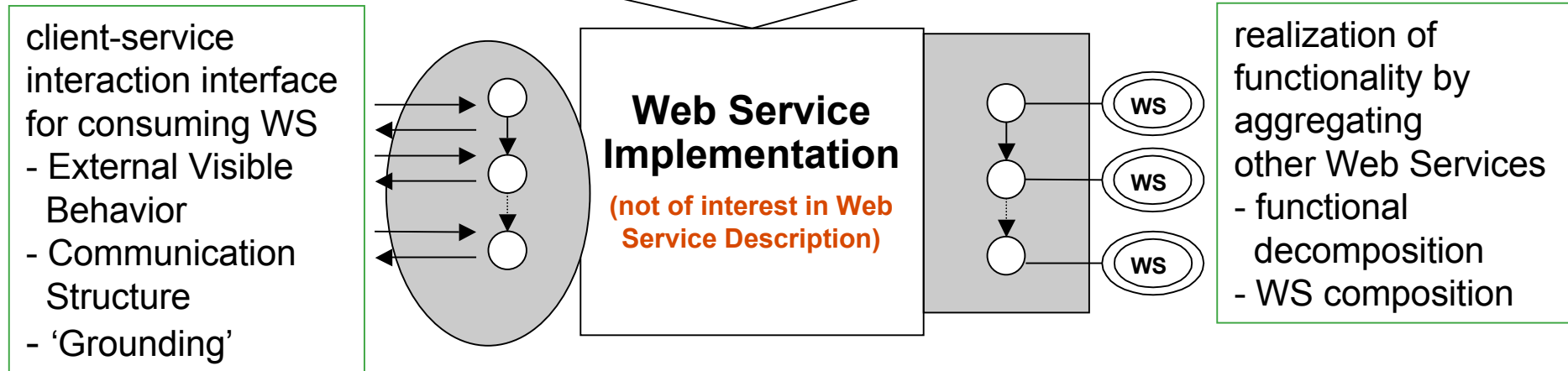
- Advertising of Web Service
- Support for WS Discovery

Non-functional Properties

Capability

DC + QoS + Version + financial

functional description



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

WSMO Working Group



Capability Specification

- **Non functional properties**
- **Imported Ontologies**
- **Used mediators**
 - *OO Mediator*: importing ontologies with mismatch resolution
 - *WG Mediator*: link to a Goal wherefore service is not usable a priori
- **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**

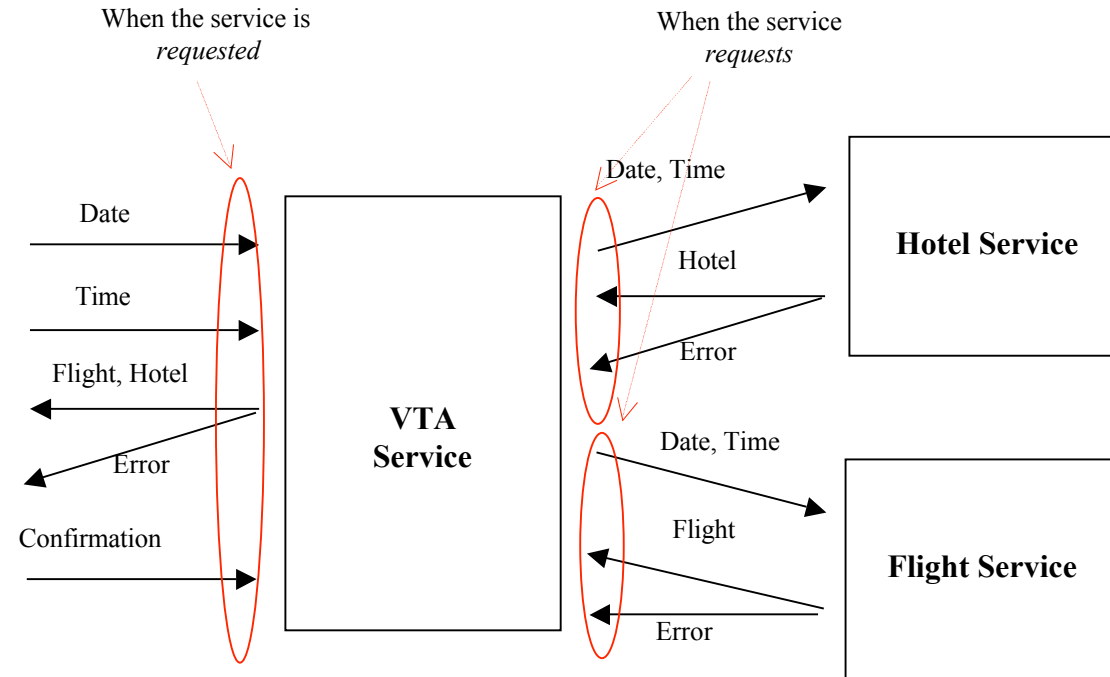
describes the result of the Web Service in relation to the input, and conditions on it
- **Effects**

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

- VTA example:



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



Service Interface Description Model

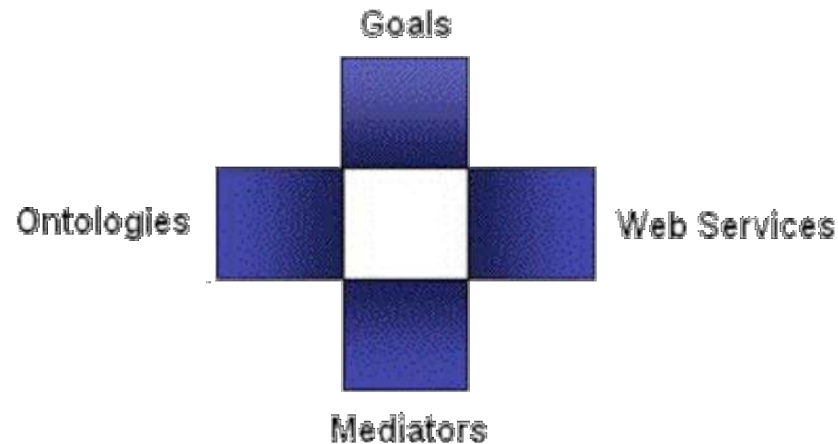
- Vocabulary Ω :
 - ontology schema(s) used in service interface description
 - usage for information interchange: in, out, shared, controlled
- States $_(_)$:
 - a stable status in the information space
 - defined by attribute values of ontology instances
- Guarded Transition $GT(_)$:
 - state transition
 - general structure: **if** (condition) **then** (action)
 - different for Choreography and Orchestration



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



Goals

- **Ontological De-coupling of Requester and Provider**
- **Derived from task / problem solving methods/domain model**
- **Structure and reuse of requests**
 - **Search**
 - **Diagnose**
 - **Classify**
 - **Personalise**
 - **Book a holiday**
- **Requests may in principle not be satisfiable**
- **Ontological relationships & mediators used to link goals to web services**



Goal Specification

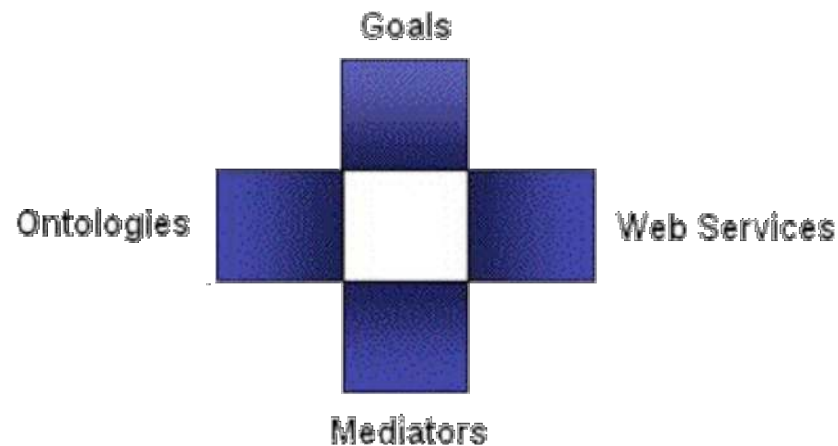
- **Non functional properties**
- **Imported Ontologies**
- **Used mediators**
 - *OO Mediators*: importing ontologies with heterogeneity resolution
 - *GG Mediator*:
 - Goal definition by reusing an already existing goal
 - allows definition of **Goal Ontologies**
- **Requested Capability**
 - describes service functionality expected to resolve the objective
 - defined as capability description from the requester perspective
- **Requested Interface**
 - describes communication behaviour supported by the requester for consuming a Web Service (Choreography)
 - Restrictions / preferences on orchestrations of acceptable Web 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



Mediation

- **Heterogeneity ...**

- For 1\$ on programming, \$5 - \$9 on integration
- Mismatches on structural / semantic / conceptual / level
- Assume (nearly) always necessary

- **Description of role**

- Components that resolve mismatches
- Declarative description of arbitrary web service

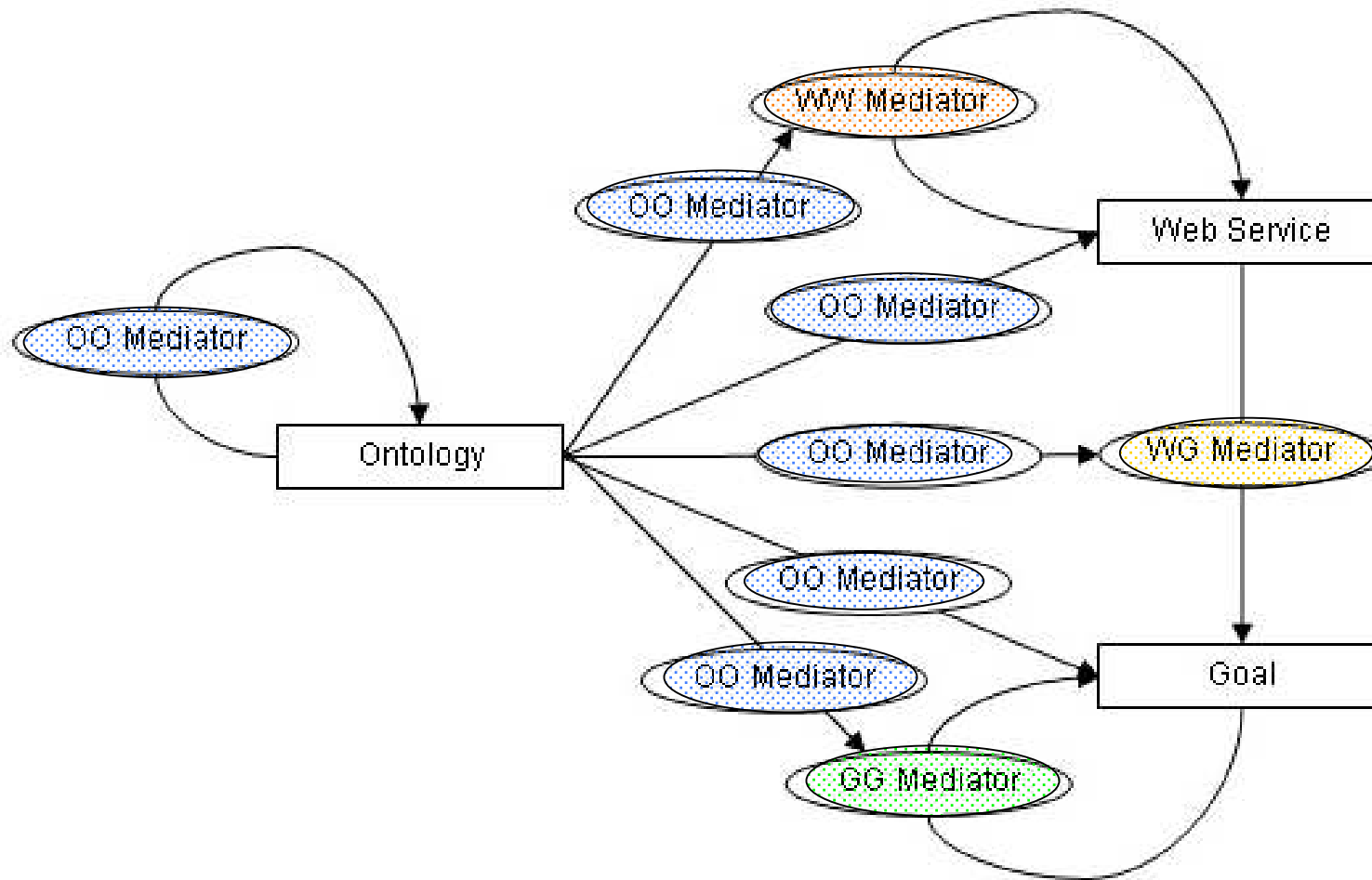
- **Types of Mediation within Semantic Web Services**

(WSMF):

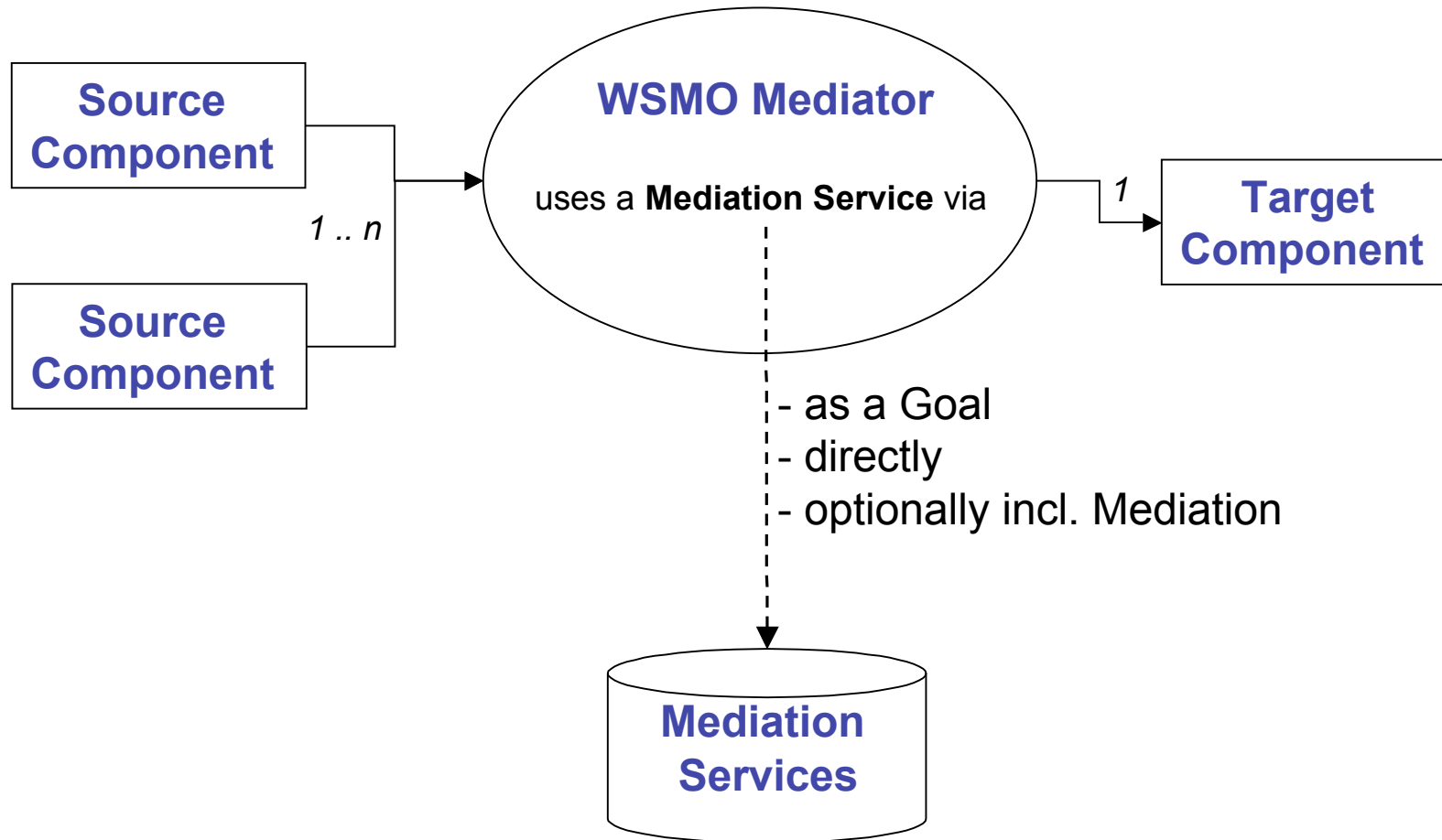
- (1) **Data:** mediate heterogeneous Data Sources
- (2) **Protocol:** mediate heterogeneous Communication Patterns
- (3) **Process:** mediate heterogeneous Business Processes



WSMO Mediators Overview



Mediator Structure

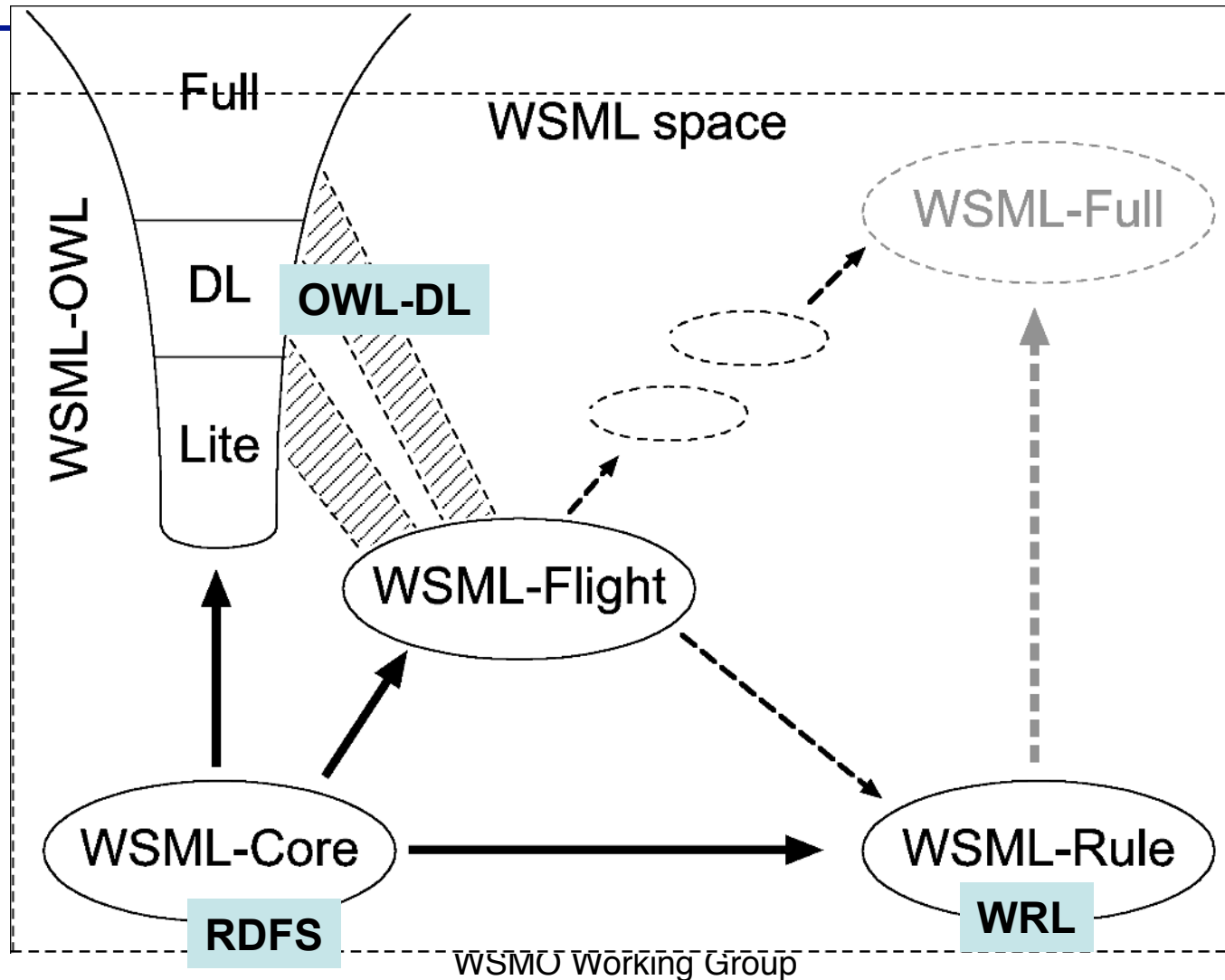


Relationship to standards

- Web Service
 - Choreography in web services
 - "the interactions of services with their users. Any user of a Web service, automated or otherwise, is a client of that service. These users may, in turn, be other Web Services, applications or human beings."
 - WSDL
 - WS-CDL
- Semantic Web
 - RDF
 - OWL



Variants of WSML

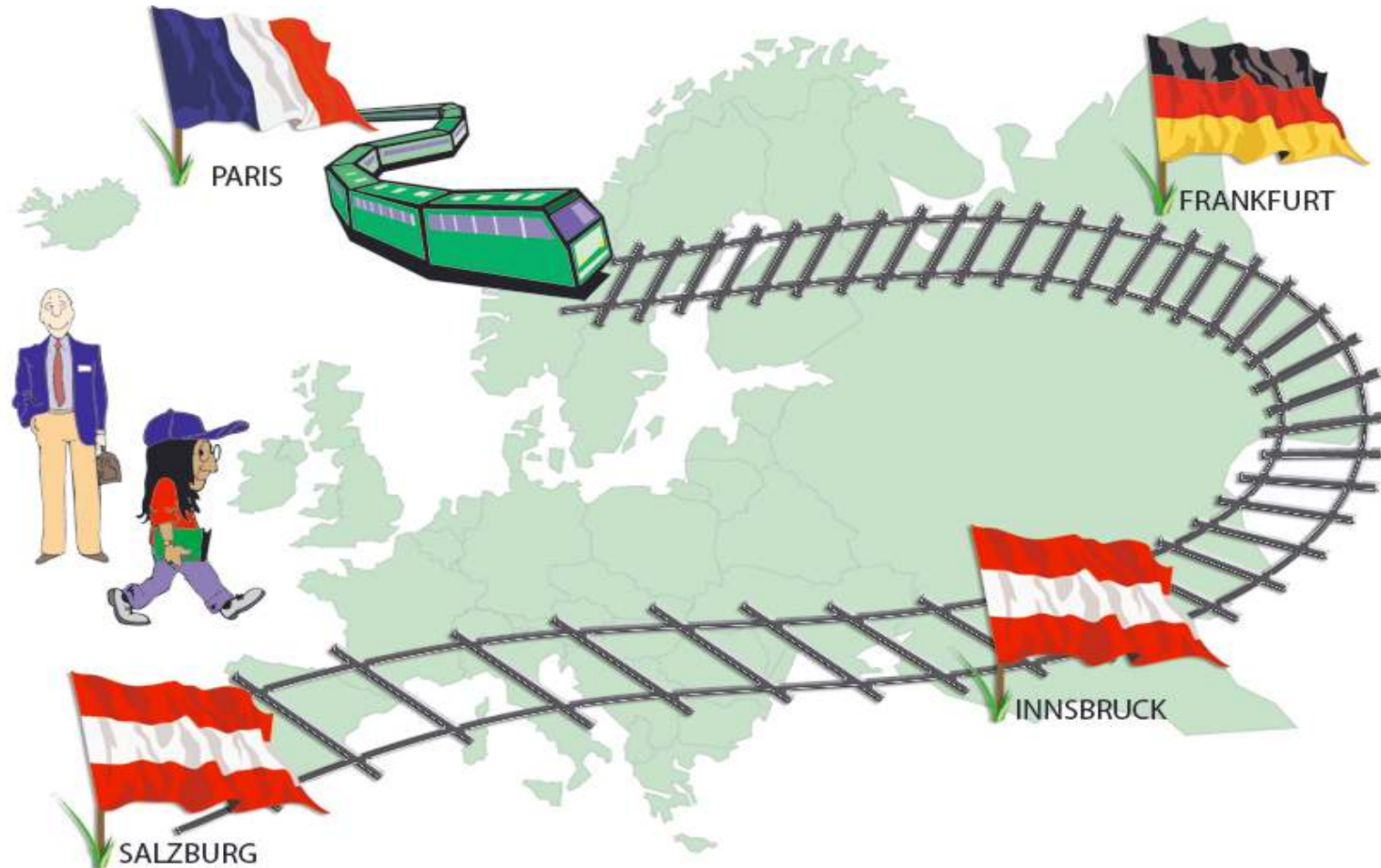


Use Cases

- WSMO use cases
- WSMX use cases
- 3 within DIP
 - eGovernment
 - B2B Integration in Telecoms
 - eBanking
- Other EU projects
 - Cocoon project
 - ASG
 - KnowledgeWeb



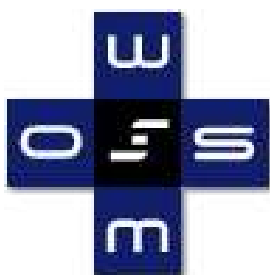
European Train Travel Demo



WSMO Summary

- Web compliant
- Represented in WSML family
- Goal
 - User as (ontological & distinct) “First class citizen”
- Web Service
 - Capability
 - Interface
 - Choreography
 - Orchestration
- Mediator
 - Source, target, mediation service
 - Brought to fore because of importance





<http://www.wsmo.org>