

## Semantic Web Services

**Exam Preparation**  
Lecture XIV – 2<sup>nd</sup> July 2009  
Dieter Fensel



### Where are we?

| #  | Date                   | Title                                |
|----|------------------------|--------------------------------------|
| 1  | 5 <sup>th</sup> March  | Introduction                         |
| 2  | 12 <sup>th</sup> March | Web Science                          |
| 3  | 19 <sup>th</sup> March | Service Science                      |
| 4  | 26 <sup>th</sup> March | Web Services (WSDL, SOAP, UDDI, XML) |
| 5  | 2 <sup>nd</sup> April  | Web 2.0 and RESTful services         |
| 6  | 23 <sup>rd</sup> April | WSMO                                 |
| 7  | 30 <sup>th</sup> April | WSML                                 |
| 8  | 7 <sup>th</sup> May    | WSMX                                 |
| 9  | 14 <sup>th</sup> May   | OWL-S and others                     |
| 10 | 28 <sup>th</sup> May   | WSMO-Lite, MicroWSMO                 |
| 11 | 4 <sup>th</sup> June   | SWS Use Cases                        |
| 12 | 18 <sup>th</sup> June  | seekda: the business point of view   |
| 13 | 25 <sup>th</sup> June  | Mobile services                      |
| 14 | 2 <sup>nd</sup> July   | Exam Preparation                     |



### In this lecture

- Goal of this lecture is to
  - Recapitulate main points of the course
  - Leave space for discussion

### Semantic Web Services Introduction

- Vision of the Semantic Web
  - Layer cake
- Ontologies as the basic building block
  - Definition
  - Languages
- Web Services
  - Definition
  - Distinction between a Service and a Web Service
  - Deficiencies of the current WS technologies
- Semantic Web Services
  - Vision and challenges

- What is the Web?
  - Definition
  - Structural and semantic components
- What is the Web science?
  - Definitions and endorsements
  - Multi-disciplinary approach
  - Goals
  - Process and methodology
  - Challenges

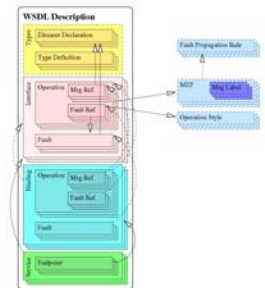
- What is service?
  - Service vs. Web Service
- Service properties
  - Functional, behavioral and non-functional
- What is Service Science?
  - Goals
- What is SOA?
  - Main actors, principles, lifecycle, self-\* properties
- What is SESA?
  - Layers

- SOAP
  - Message Structure
  - Processing Model
  - Protocol Bindings
  - Message Exchange Patterns

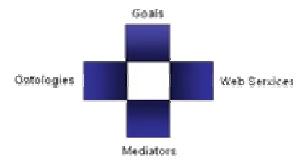
```

<env:Envelope
  xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header>
    <n:alertcontrol xmlns:n="http://example.org/alertcontrol">
      <n:priority>1</n:priority>
      <n:expires>2001-08-22T14:00:00-05:00</n:expires>
    </n:alertcontrol>
  </env:Header>
  <env:Body>
    <m:alert xmlns:m="http://example.org/alert">
      <m:msg>Pick up Mary at school at 2pm</m:msg>
    </m:alert>
  </env:Body>
</env:Envelope>
    
```

- WSDL
  - Interface
  - Message Exchange patterns
    - Invocation
    - Messaging
  - Binding
  - Service

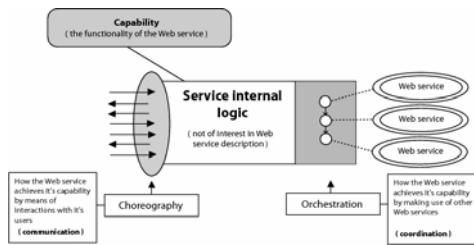


- REST Concepts
  - Architectural style
  - HTTP the main implementation of REST
  - REST Ingredients
    - Client-Server, Layering, Stateless Communication, Uniform Interface, Caching, Code-on-Demand
- Definition of a RESTful Web Service
- RESTful Models
  - Hypermedia
  - Operations

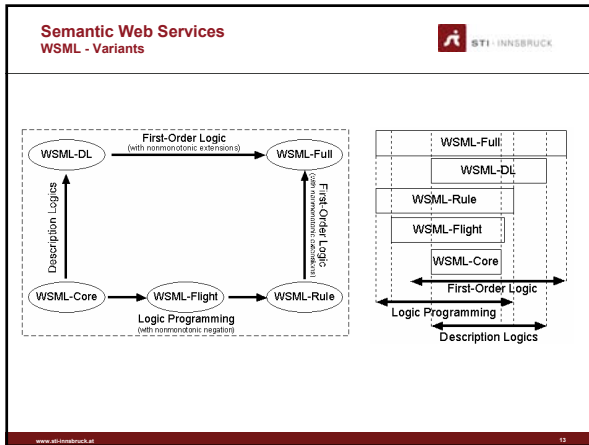


- Ontologies
  - Specification
  - Ontology elements: Concepts, Attributes, Relations, Functions, Instances, Axioms

- Distinction between the Web Service and Goal



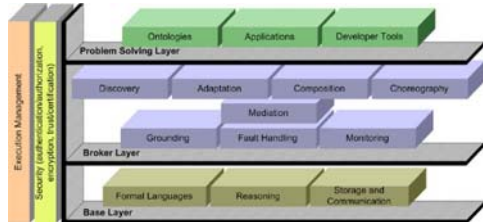
- Mediation
  - Data Level
  - Protocol Level
  - Process Level
- Four different types of mediators in WSMO
  - ggMediators
  - ooMediators
  - wgMediators
  - wwMediators



- Semantic Web Services**  
WSML - Syntax
- The WSML Full syntax consists of two major parts: the **conceptual syntax** and the **logical expression syntax**
    - The conceptual syntax is used for the modeling of ontologies, goals, web services and mediators; these are the elements of the WSMO conceptual model
    - Logical expressions are used to refine these definitions using a logical language
  - The other language variants impose restrictions on the general syntax (WSML Full syntax)

- Semantic Web Services**  
WSML - Syntax
- A WSML specification contains all information about a class and its attributes, a relation and its parameters and an instance and its attribute values in one large syntactic construct, instead of being divided into a number of atomic chunks
  - Attributes are defined locally to a class
  - A WSML specification is separated into two parts
    - Meta information part, and
    - Specification of concepts, attributes, instances, relations, axioms, interfaces, ...
  - WSML adopts the **namespace** mechanism of RDF; a namespace can be seen as part of an IRI
  - An **identifier** in WSML is either a *data value*, an *IRI*, an *anonymous ID*, or a *variable*
  - WSML has direct support for different types of concrete data corresponding to XML Schema primitive datatypes

- Semantic Web Services**  
WSML - Syntax
- WSML Prologue contains all those elements that are in common between all types of WSML specifications and all WSML variants
  - WSML Header
    - Nonfunctional properties
    - Importing ontologies
    - Mediator usages
  - WSML Ontologies
  - WSML Capabilities
  - WSML Logical Expressions
  - Restrictions of WSML variants



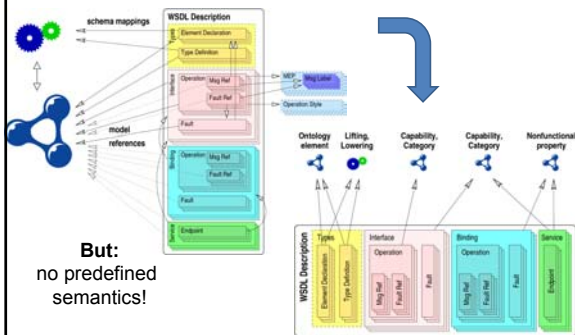
- Design Principles
  - Service-oriented principle
  - Semantic principle
  - Problem-solving principle
  - Distributed principle
- Lifecycle
  - Discovery > Composition > Selection > Mediation > Choreography > Invocation

- Discovery
  - Key word vs. semantic matchmaking
- Selection and ranking
- Data mediation
  - Design vs. run-time phase
- Process mediation
- Choreography
- Invocation
  - Role of grounding
- Retrieve WSMO artifacts

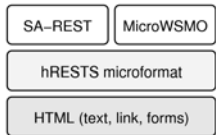
- Mandatory execution semantics
  - Goal-Based Web Service Discovery
  - Web Service Invocation
  - Goal-Based Service Execution

- IRS III
- OWL S
- METEOR S
- SWSF
  - Conceptual model
  - Relations and differences to WSMO/L/X

- Service model (expressed in RDF(S))
- Service semantics
  - Functional
  - Behavioral
  - Nonfunctional
  - Information model



- HTML for RESTful Service Description
  - Introduces the service model structure
    - service (+ label)
    - operations (+ address, method)
    - input, output
  - Could also be in RDFa
  - Basis for extensions:
    - MicroWSMO adds semantic annotations
- MicroWSMO Extends hRESTS
    - model for model references
    - lifting, lowering
  - Applies same semantics as WSMO-Lite



## Semantic Web Services Use Cases



- DIP
  - developing and extending Semantic Web and Web Service technologies in order to produce a new technology infrastructure for Semantic Web Services (SWS)
  - B2B in telecommunication, Contract Catalogue Case Study, GIS Emergency Planning, eBanking
- SUPER
  - to raise Business Process Management (BPM) to the business level, where it belongs, from the IT level where it mostly resides now.
  - Benefits and business opportunities
- SWING
  - develop an open, easy-to-use Semantic Web Service framework of suitable ontologies and inference tools for annotation, discovery, composition, and invocation of geospatial web services
  - Mineral Resources Management
- SOA4All
  - will facilitate a Service Web of billions of services revolutionizing the access and usage of software
  - Public sector, BT Web21c, C2C Service e-Commerce

## Semantic Web Services seekda



- Search Engine for Web Services
  - fully automated focused crawling process
  - aggregating information from multiple sources into a semantic model
  - efficient means for finding services
  - community features enabling understanding and selecting right services
- Focused Crawling
- Data analysis
- Web Service Marketplace
- Software as a Service (SaaS)

## Semantic Web Services Mobile Services



- Enabling mobile technologies
- End-User Empowerment in Converging Service Platforms
- Enabling User-Driven Semantics
  - SPICE Project
- User-Generated Policies
  - PAT tool
- User-Generated Mobile Microservices
  - M:Ciudad project

## Next Lecture



| S  | Date                   | Title                                |
|----|------------------------|--------------------------------------|
| 1  | 6 <sup>th</sup> March  | Introduction                         |
| 2  | 12 <sup>th</sup> March | Web Science                          |
| 3  | 19 <sup>th</sup> March | Service Science                      |
| 4  | 26 <sup>th</sup> March | Web-Services (WSDL, SOAP, UDDI, XML) |
| 5  | 2 <sup>nd</sup> April  | Web 2.0 services/ restful services   |
| 6  | 23 <sup>rd</sup> April | WSMO                                 |
| 7  | 30 <sup>th</sup> April | WSML                                 |
| 8  | 7 <sup>th</sup> May    | WSMX                                 |
| 9  | 14 <sup>th</sup> May   | OWL-S and others                     |
| 10 | 28 <sup>th</sup> May   | SA-WSDL, WSMO-Lite, MicroWSMO        |
| 11 | 4 <sup>th</sup> June   | SWS are good for what                |
| 12 | 18 <sup>th</sup> June  | seekda: the business point of view   |
| 13 | 25 <sup>th</sup> June  | Mobile services                      |
| 14 | 2 <sup>nd</sup> July   | Exam                                 |

Questions?

