Semantic Web Technologies II

SS 2009

01.07.2009

Semantic Web Services

Dr. Sudhir Agarwal
Dr. Stephan Grimm
Dr. Peter Haase
PD Dr. Pascal Hitzler
Denny Vrandecic
Martin Junghans

Agenda

- Web Services
  - Definition
  - Standard technologies
- Semantic Web Services
  - Rationale
  - Application scenarios
  - Discovery, ranking, composition
- KASWS – Karlsruhe Semantic Web Services

Web Services

- So far, we focused on the description of the content (static data) of Web pages
  - Information consumption by humans
  - Communication between humans

- Here, the focus is
  - Information consumption by machines
  - Data exchange between machines

What the W3C says

A Web service is a **software system** designed to support **interoperable machine-to-machine** interaction over a network. It has an **interface** described in a **machine-processable** format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its **description** using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related **standards**.

© W3C, 2004. Web Services Architecture
http://www.w3.org/TR/ws-arch/#whatis

Web Services

- Abstract from
  - Hardware, location
  - Communication protocols
  - Software / implementation

  as they rely on standards

- One possibility to implement service oriented architectures (SOA)

- Technology for application integration
  - Functionalities offered via the Web

Further Advantages

- Interoperability within and among enterprises
- Large-scale distributed computing
- Improves evolvability

- Drawback:
  - Careful engineering required

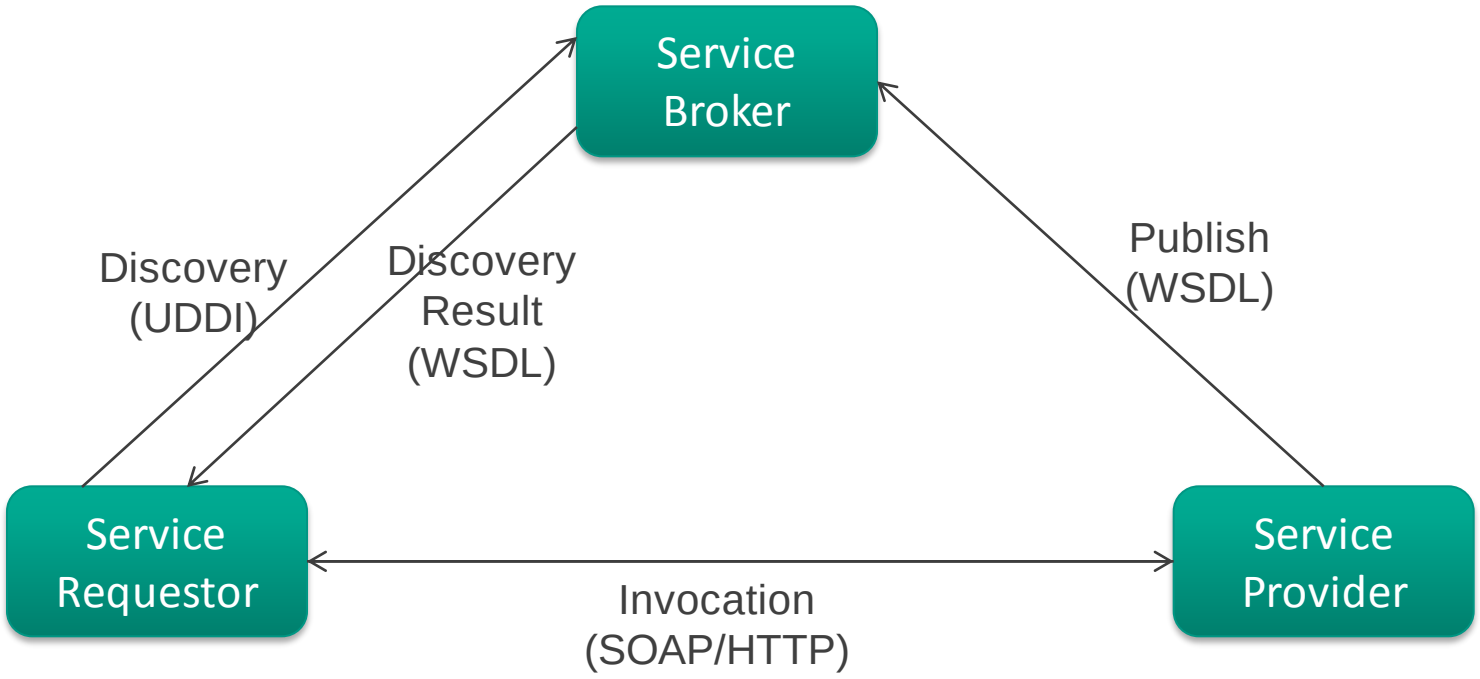
Agenda

- Web Services
  - Definition
  - Standard technologies
- Semantic Web Services
  - Rationale
  - Application scenarios
  - Discovery, ranking, composition
- KASWS – Karlsruhe Semantic Web Services

Web Service Standards

- Web service architecture based on standards for
  - Connection
  - Communication
  - Description
  - Discovery

Architectural Model



eXtensible Markup Language (XML)

- Information exchange between Participants
- Message format specification
  - SOAP messages
  - WSDL service description files

Simple Object Access Protocol (SOAP)

- Exchange structured data (XML)
- Defines XML message patterns, i.e., syntax only
- Relies on other protocols
  - Often Remote Procedure Call (RPC) and HTTP
- Independent of transport protocols
  - Often HTTP
- SOAP messages represent cross platform remote calls

SOAP Request: Stock Quote

```
<?xml version="1.0"?>
<soap:Envelope
 xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
 soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">

   <soap:Body xmlns:tns="http://www.example.org/stockquote.wsdl">
     <tns:GetStockPriceRequest>
       <tns:stockSymbol>IBM</tns:stockSymbol>
       <tns:time>2009-07-01T21:42:23Z</tns:time>
     </tns:GetStockPriceRequest>
   </soap:Body>

</soap:Envelope>
```

SOAP Response: Stock Quote

```
<?xml version="1.0"?>
<soap:Envelope
 xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
 soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">

   <soap:Body xmlns:tns="http://www.example.org/stockquote.wsdl">
     <tns:GetStockPriceResponse>
       <tns:stockPrice>34.5</tns:stockPrice>
     </tns:GetStockPriceResponse>
   </soap:Body>

</soap:Envelope>
```

Web Service Description Language (WSDL)

- Format to offer services
- Application-specific interfaces
- Describes information required for invoking a Web service
  - Operations that a Web Service can perform
  - Messages it can process
  - Protocols it can support
  - Physical bindings to URIs and protocols

WSDL Structure

```
<definitions>
  <types>
    Data types used by the Web service
  </types>
  <message>
    Messages used by the Web service
  </message>
  <portType>
    Operations performed by the web service
  </portType>
  <binding>
    Communication protocols used by the Web service
  </binding>
</definitions>
```

WSDL Example: Stock Quote

```
<?xml version="1.0"?>
<definitions name="StockQuote"
 targetNamespace="http://example.com/stockquote.wsdl"
 xmlns:tns="http://example.com/stockquote.wsdl"
 xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
 xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
 xmlns="http://schemas.xmlsoap.org/wsdl/">

    <message name="GetStockPriceRequest">
        <part name="stockSymbol" element="xsd:string"/>
        <part name="time" element="xsd:timeInstant"/>
    </message>

    <message name="GetStockPriceResponse">
        <part name="stockPrice" type="xsd:float"/>
    </message>
 ...
</definitions>
```

Namespaces

Exchanged messages

WSDL Example: Stock Quote

```
<?xml version="1.0"?>
<definitions name="StockQuote"

 ...
    <portType name="StockQuotePortType">
        <operation name="GetStockPrice">
            <input message="tns:GetStockPriceRequest"/>
            <output message="tns:GetStockPriceResponse"/>
        </operation>
    </portType>
 ...

</definitions>
```

Abstract operations and messages

WSDL Example: Stock Quote

```
<?xml version="1.0"?>
<definitions name="StockQuote„ ...
 ...
  <binding name="StockQuoteSoapBinding" type="tns:StockQuotePortType">
    <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="GetStockPrice">
      <soap:operation soapAction="http://example.com/GetStockPrice"/>
      <input>
        <soap:body use="encoded" namespace="http://example.com/stockquote"
                   encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/>
      </input>
      <output>
        <soap:body use="encoded" namespace="http://example.com/stockquote"
                   encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/>
      </output>
    </operation>
  </binding>

 ...
</definitions>
```

Protocol

Message format

Message format

Universal Description, Discovery, and Integration (UDDI)

- Service registry API specification for businesses
- Provides WSDL documents
- Components of registry
  - White pages
    - Address, name, tax number
  - Yellow pages
    - Taxonomy based categorization
  - Green pages
    - Technical details (operating platform, billing requirements)

Web Service Standards

- Additional specifications available
  - WS-* stack comprises
    - Messaging
      - WS-Addressing, WS-Transfer
    - Metadata Exchange
      - WS-Discovery, WS-Policy
    - Security
      - WS-Security, WS-Trust
    - Business processes
      - WS-BPEL, WS-Choreography
    - …

Agenda

- Web Services
  - Definition
  - Standard technologies
- Semantic Web Services
  - Rationale
  - Application scenarios
  - Discovery, ranking, composition
- KASWS – Karlsruhe Semantic Web Services

Semantic Description of Web Services

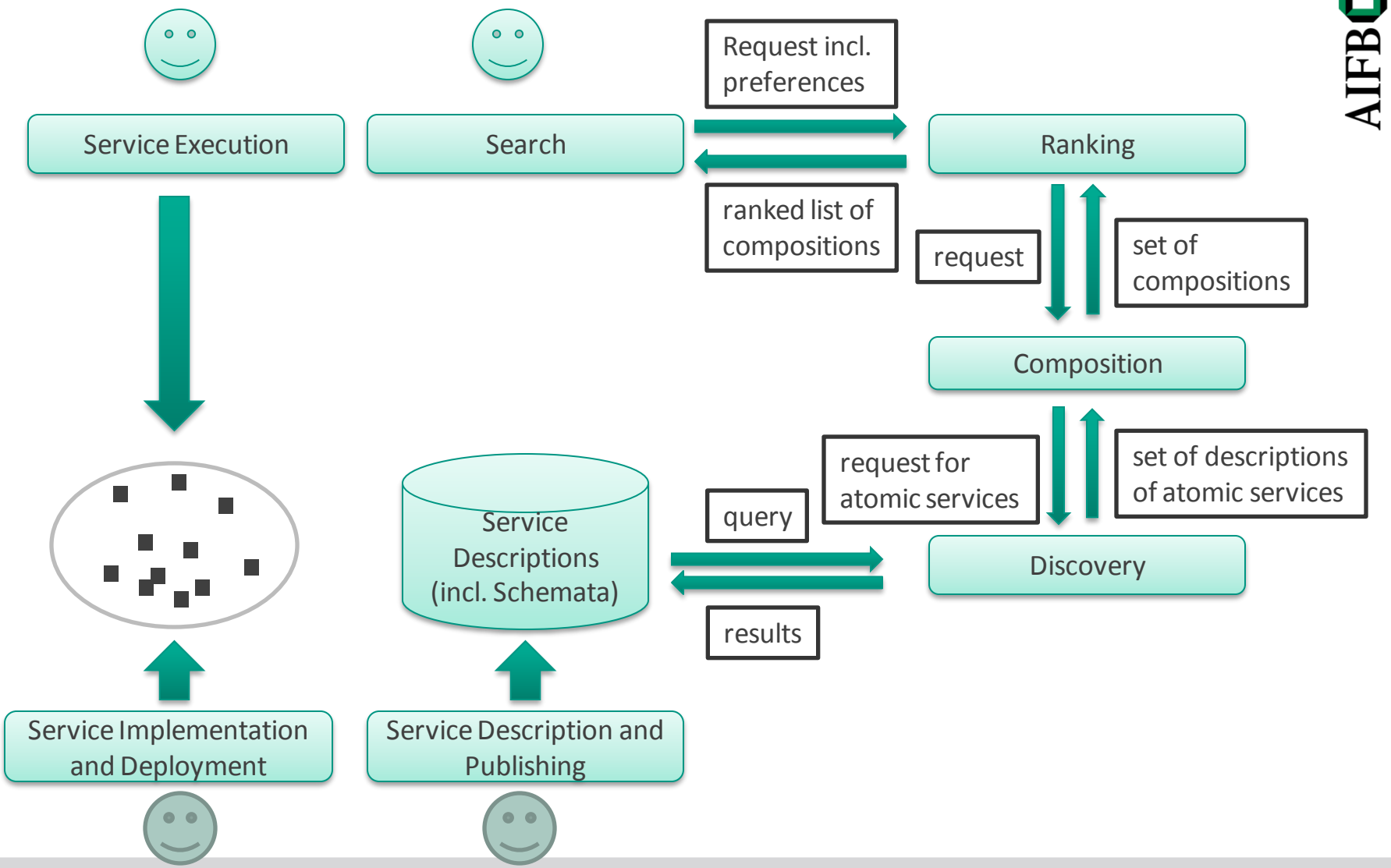
- Why do we need Semantics?
  - As described, Web services work fine
  - SOA and Web Services are widely adopted in practice

- Service Web
  - Services composed out of other services
  - Mashups, Processes
- SOAP, WSDL only define syntax (XML Schema)
  - Sufficient for service invocation
- Plenty of scenarios need automation
  - Discovery, ranking, composition, execution, etc.

Semantic Description of Web Services

- XML Schema
  - Describes syntax of messages only
    - Different ways for expressing an artifact
      Interpretation by a human necessary
    - Interoperability of data difficult,
      since schemata can not be aligned easily
  - Allows to validate XML documents
  - Formal semantics given by interpretation as a tree.
    However, meaning of an edge can not be modeled explicitly.

A Simple Scenario



Motivation of Semantic Web Services

- Increasing amount of Services
  → We need support for automation
- Semantic descriptions foster automatic …
  1. **Discovery**
     - *Synonymy*: different terms for service offers and requests
       - Search for `Video` won't find `Movie` services
     - User needs to understand the functionality of a service
       - Does the service deliver the ordered book or just any book?

Motivation of Semantic Web Services

- Increasing amount of Services
  → We need support for automation
- Semantic descriptions foster automatic …
  2. **Composition**
     – Create new applications, mashups, services, processes
     – *Data mediation*: reconcile service messages with that of a user or other services in the composition
       – Service A outputs a `StudRegNr`, service B expects a `MatrNr`. Can B be invoked with A's output?

Motivation of Semantic Web Services

- Increasing amount of Services
  → We need support for automation
- Semantic descriptions foster automatic …
  3. **Ranking**
     - Not only service functionality but also its quality is important
     - User need to understand the QoS attributes
       - How does the `Speed` of service A relate to `ResponseTime` of Service B?

Agenda

- Web Services
  - Definition
  - Standard technologies
- Semantic Web Services
  - Rationale
  - Application scenarios
  - Discovery, ranking, composition
- KASWS – Karlsruhe Semantic Web Services

Semantic Web Service Discovery

- WSDL documents describe type of input/output
  - Do not describe *functionality*,
    that is how output relates to input
- Discovery requires functionality description
- Semantics allows to describe what a service actually does
  - Depends on SWS formalism
  - Provides information needed for discovery
  - User goal is matched against service descriptions

Semantic Web Service Discovery

- Example:
  - **User goal**: service returning student's `last name` for a given `registration number`
  - **Background knowledge**:
    `Matrikelnummer` is the same as `registration number`
    `Last name` is part of `full name`
  - **Matching service example**:
    return `full name` for given `Matrikelnummer`

Global Semantic Web Service Ranking

- Example Google Web search
  - Order of Web pages is more critical than discovery
  - Proven by current popularity of Google
- Google does global ranking by relevance
  - Same for all users
  - Pre-computed

- Services can be ranked by *non-functional properties*
- Global preference structures
  - E.g., fast and reliable services first

Local Semantic Web Service Ranking

- Ranking also considers user's preference structure
  - E.g., Google product search: sort by rating
- Possibilities to define preference structures
  a. Sort first by X, then by Z
  b. Valuation function over non-functional parameter values
     value(responseTime, availability)
     = value(availability) – 0.1 value(responseTime)
     = 0.8 – 0.1*0.9
     = 0.71

Semantic Web Service Ranking

- SWS formalism must
  - Describe the user preferences
    - Criteria (e.g., availability, price)
    - Order (e.g., ascending vs. descending)
  - Model the service attributes
    - E.g., availability, price
  - Understand criteria (ontology reasoning)

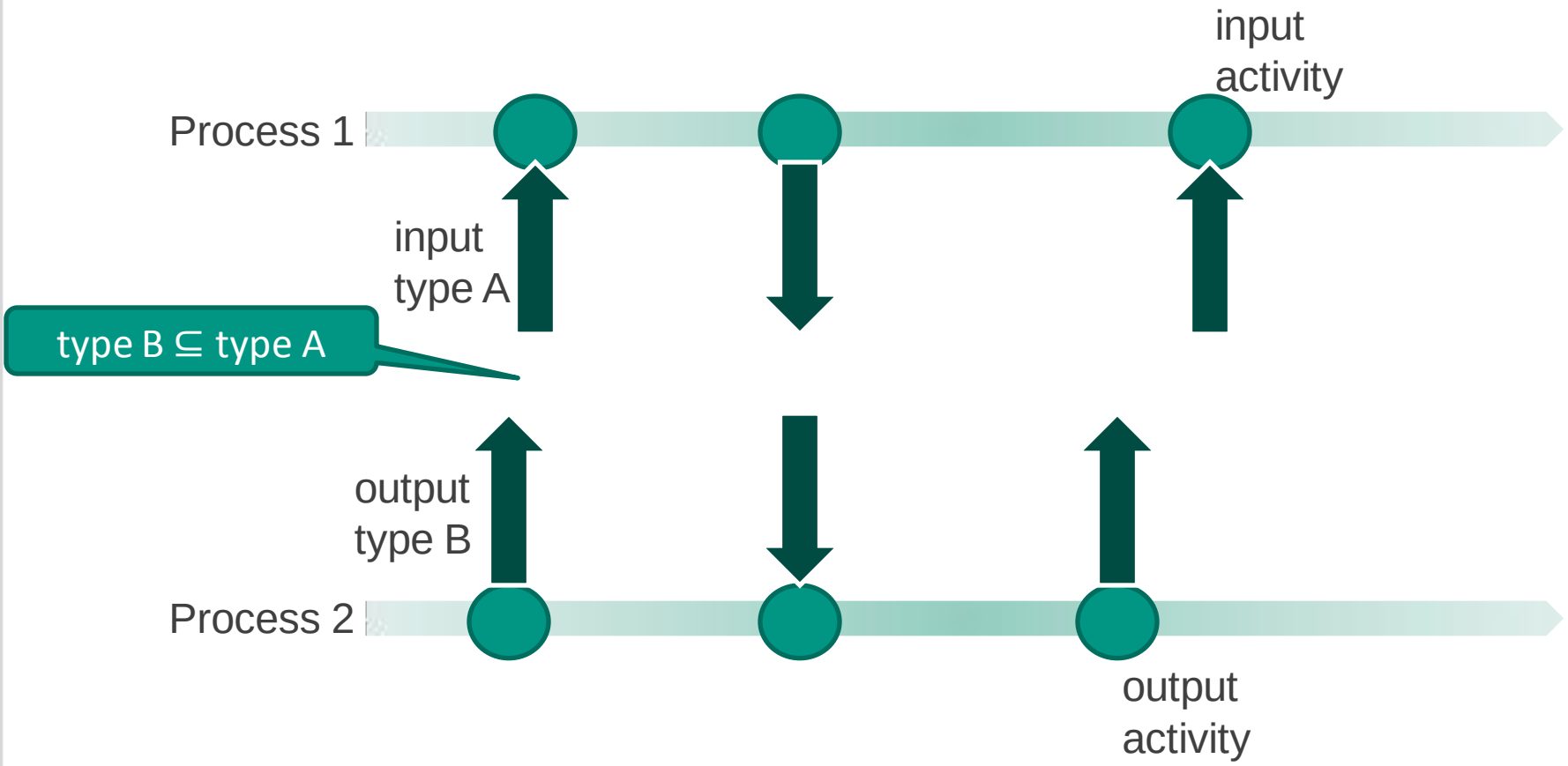
Semantic Web Service Composition

- Requirements
  - Data flow
    - Data types should fit together
  - *Choreography*
    - Inputs of a service dovetail outputs of other service(s) and vice versa

Choreography

- Coordinate collections of Web services
  - Data and control flow of different processes
  - Input/output requirements
  - Data type checking is a check of subclasses
    - Ontology reasoning
- Ensure interoperability
- Protocols of WS-*
  - WS-Choreography
  - WS-CDL

Choreography



Summary

- Web services
  - Principle
  - Standards
- Semantic Web services
  - Motivation
  - Need for a formalism to describe Web services
  - Discovery, ranking, composition

Outlook

- Semantic Web service formalism
  - In this lecture:
    - Karlsruhe Semantic Web Services (KASWS, 08.07.)
  - Not in this lecture:
    - OWL and SAWSDL
    - WSMO