



# Web Services

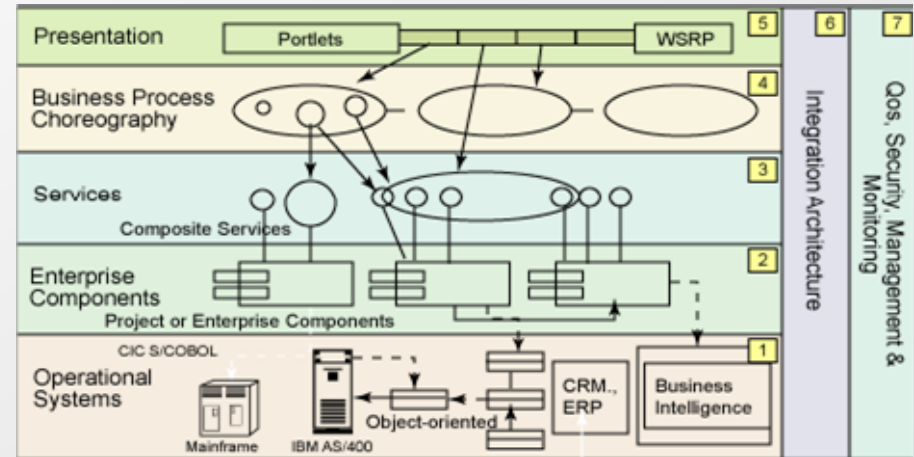
Moving towards Service Oriented  
Architectures

# Agenda

- Service Oriented Architectures (SOA)
- Web Services
  - Simple Object Access Protocol (SOAP)
  - Web Services Description Language (WSDL)
  - Universal Description, Discovery, and Integration (UDDI)
  - Tools
  - Examples
- Grid Services

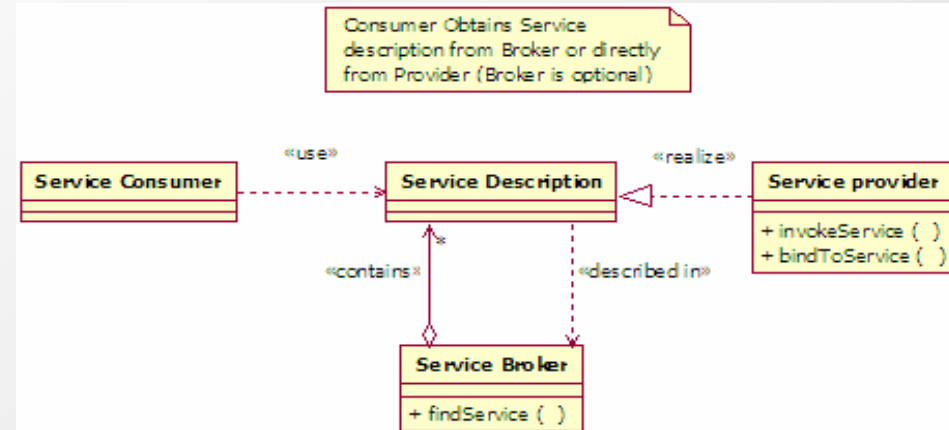
# Service Oriented Architecture (SOA)

- SOAD vs. OOAD
  - New
    - Services
    - Flows
    - Components
- Services Layer
  - Placed on top of existing products
- Roles
  - Service provider
  - Service consumer



# Service Oriented Architecture (SOA)

- Architectural style that defines an interaction model between 3 primary parties
  - Service Provider
    - Publishes a service description and provides the implementation for the service
  - Service Consumer
    - Uses either the uniform resource identifier (URI) for the service description directly or can find the service description in a service registry
  - Service Broker
    - Provides and maintains the service registry, although nowadays public registries are not in vogue.



Role	Activities in this role				
Consumer view	Service identification	Service categorization	Service exposure decisions	Choreography or composition	Quality of service
Provider view	Component identification	Component specification	Service realization	Service management	Standards implementation
	Service allocations to components	Layering the SOA	Technical prototyping	Product selection	Architectural decisions (state, flow, dependencies)

# Service Oriented Architecture (SOA)

- When To Use SOA
  - Heterogeneous Environment Present
  - Real-time Performance is not critical
  - Dynamic Information
  - Loose Coupling

# Service Oriented Architecture (SOA)

- SOA Design Strategy
  - “Top Down”
  - Strategic and Business Aligned
  - Stems from Business Process Choreography
- Web Services
  - “Bottom Up”
  - Tactical Implementation of SOA
  - Stems from existing components

# Services Layer

- Service Types
  - *Stateless Service*
    - Message exchanges, no information transferred from persistent resources
  - *Conversational Service*
    - Implements a series of operations
  - *Stateful Service*
    - Provides access to, or manipulates a set of logical stateful resources (documents or data sets)

# Service Layer Implementation

- Web Services System
  - Identified by a URI
  - XML defined interfaces and bindings
  - Discoverable by other software system
  - Other systems interact with the Web service based on its definition
  - XML based messages between systems



# Web Services

1. WS
2. RE
3. WS-  
Interface
4. Message  
Processing
5. Requests
6. Message  
Dispatch
7. Stateful  
Resource

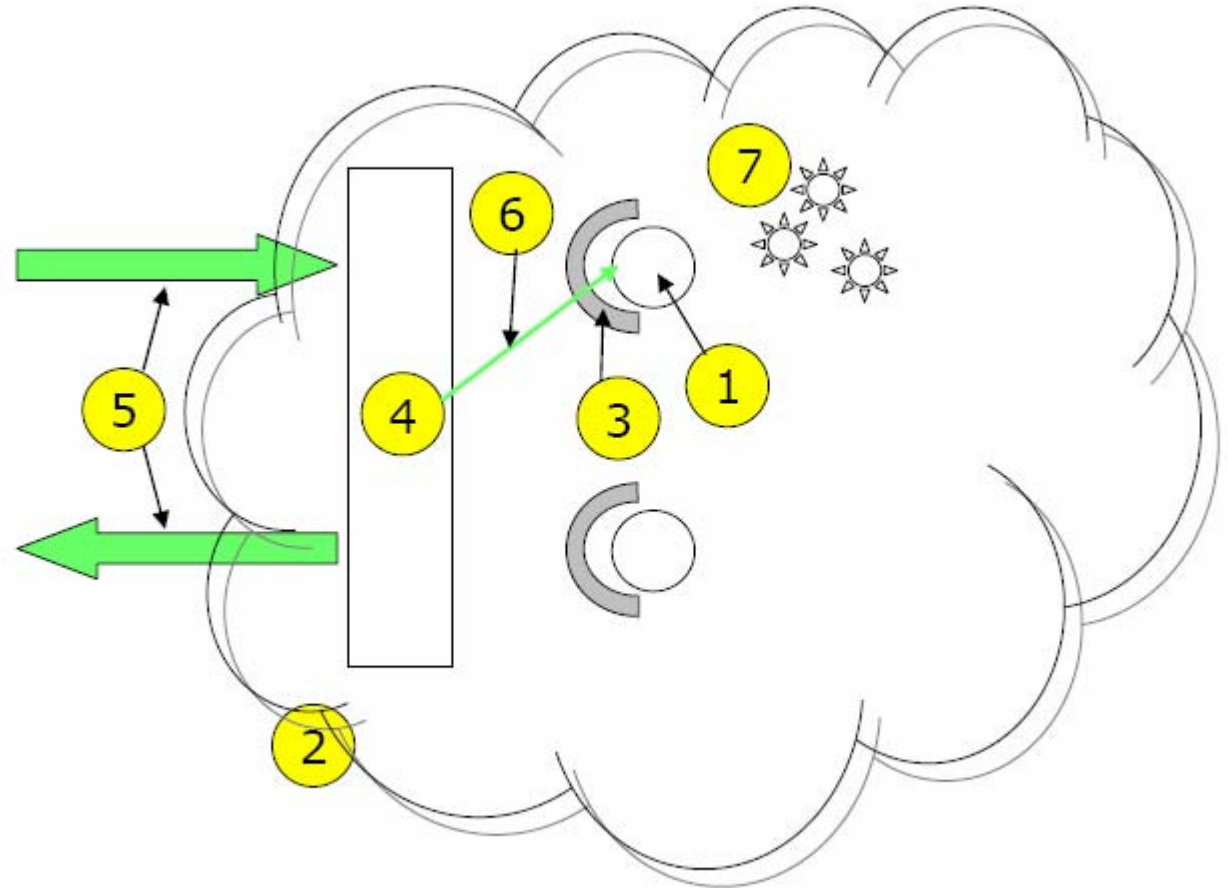
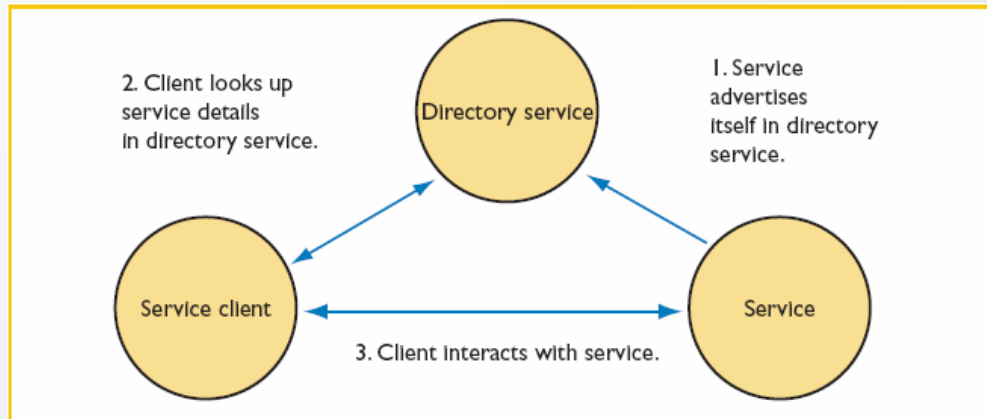


Figure 1 Facets related to a Web service

# Web Services

- Protocol Stack
  - Similar to TCP/IP



DISCOVER  
(UDDI)

DESCRIBE  
(WSDL)

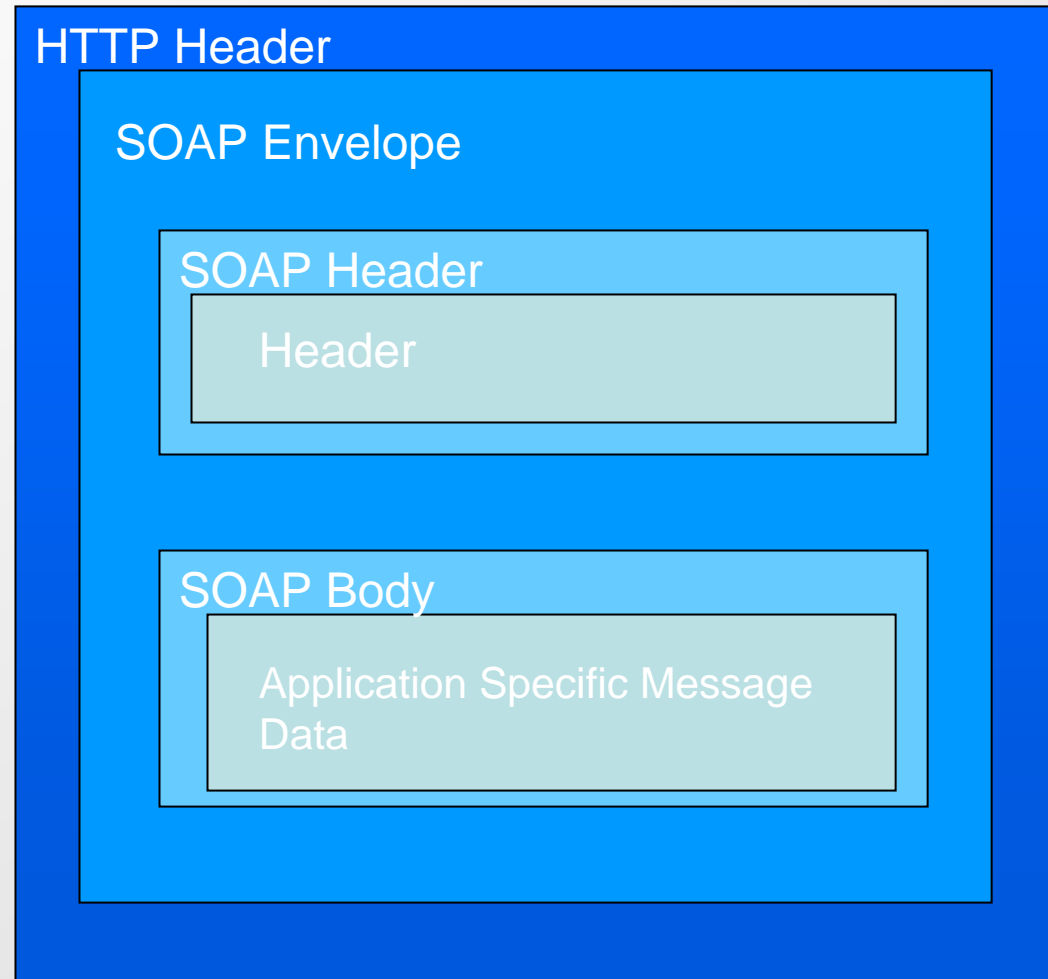
ACCESS  
(SOAP)

STRUCTURE  
(XML)

TRANSPORT  
(HTTP)

# Simple Object Access Protocol (SOAP)

- XML message protocol to access web services



# SOAP Travel WS Example

```
<?xml version='1.0' ?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-
envelope">
  <env:Header>
    <m:reservation
      xmlns:m="http://travelcompany.example.org/reservation"
      env:role="http://www.w3.org/2003/05/soap-
envelope/role/next"
      env:mustUnderstand="true">
      <m:reference>uuid:093a2da1-q345-739r-ba5d-
pqff98fe8j7d</m:reference>
      <m:dateAndTime>2001-11-29T13:20:00.000-
05:00</m:dateAndTime>
    </m:reservation>
    <n:passenger
      xmlns:n="http://mycompany.example.com/employees"
      env:role="http://www.w3.org/2003/05/soap-
envelope/role/next"
      env:mustUnderstand="true">
      <n:name>Åke Jógvan Øyvind</n:name>
    </n:passenger>
  </env:Header>
```

```
<env:Body>
  <p:itinerary
    xmlns:p="http://travelcompany.example.org/reservation/tra
vel">
    <p:departure>
      <p:departing>New York</p:departing>
      <p:arriving>Los Angeles</p:arriving>
      <p:departureDate>2001-12-14</p:departureDate>
      <p:departureTime>late afternoon</p:departureTime>
      <p:seatPreference>aisle</p:seatPreference>
    </p:departure>
    <p:return>
      <p:departing>Los Angeles</p:departing>
      <p:arriving>New York</p:arriving>
      <p:departureDate>2001-12-20</p:departureDate>
      <p:departureTime>mid-morning</p:departureTime>
      <p:seatPreference/>
    </p:return>
  </p:itinerary>
  <q:lodging
    xmlns:q="http://travelcompany.example.org/reservation/ho
tels">
    <q:preference>none</q:preference>
  </q:lodging>
</env:Body>
</env:Envelope>
```

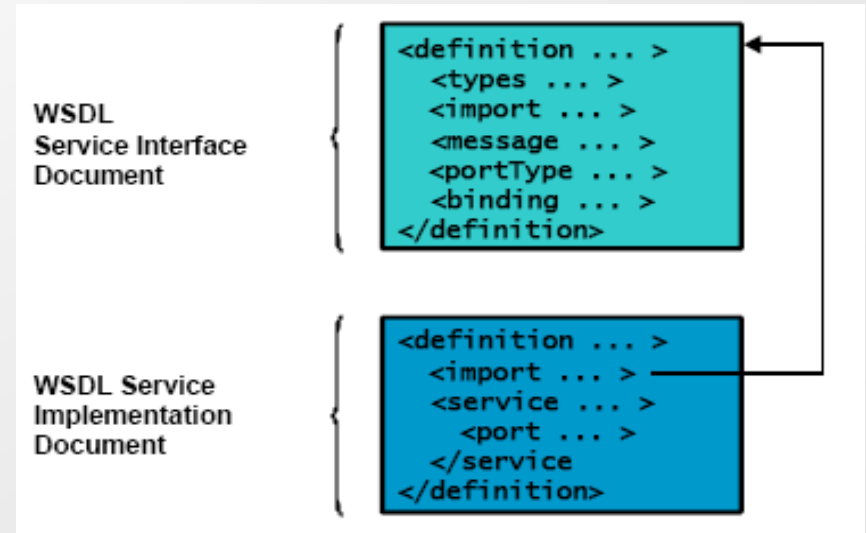
# Describing Web Services

- WSDL
  - Web Service Definition Language
- Defines
  - Web service can do
  - Where it resides
  - How to invoke it



# WSDL Types

- Service Interface
  - Contains *Types*, *import*, *message*, *portType*, and *binding* elements.
  - A service interface contains the WSDL service definition that will be used to implement one or more services. It is an abstract definition of a web service, and is used to describe a specific type of service
- Service Implementation
  - Contains *import* and *service* elements.



# WSDL Skeleton

```
<definitions>
  <types> definition of data types..... </types>
  <message> definition of a message.... </message>
  <portType> definition of a port {1 way | req/resp
  | illicit resp | notification}..... </portType>
  <binding> definition of binding to SOAP....
  </binding>
</definitions>
```

[5]

# WSDL Example

```
<message name="getTermRequest">
  <part name="term" type="xs:string"/>
</message>
<message name="getTermResponse">
  <part name="value" type="xs:string"/>
</message>
<portType name="glossaryTerms">
  <operation name="getTerm">
    <input message="getTermRequest"/>
    <output message="getTermResponse"/>
  </operation>
</portType>
<binding type="glossaryTerms" name="b1">
  <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
  <operation>
    <soap:operation soapAction="http://example.com/getTerm"/>
    <input> <soap:body use="literal"/> </input>
    <output> <soap:body use="literal"/> </output>
  </operation>
</binding>
```



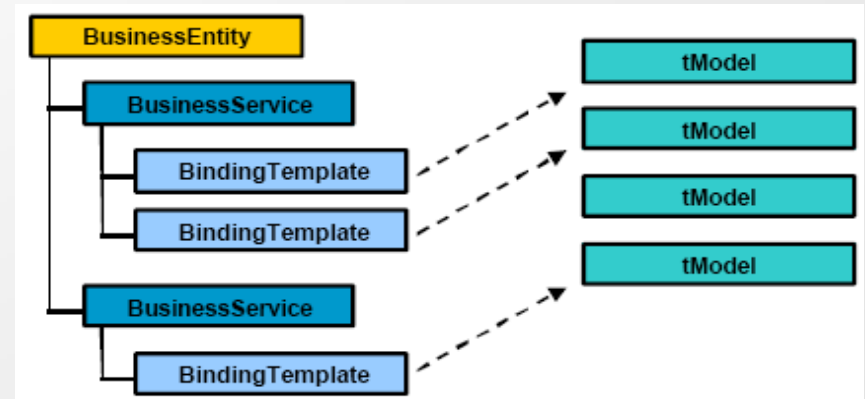
# UDDI

- Universal Description, Discovery, and Integration
  - Registry of web services
- Phone book for web services[6]
  - White pages
    - Basic contact information and identifiers.
  - Yellow pages
    - Information that describes a web service
  - Green pages
    - Information that describes behaviors and functions of a web service
- WS-Inspection
  - *WS-Inspection* is a new XML grammar that allows a Web server to be inspected for services that may be deployed.

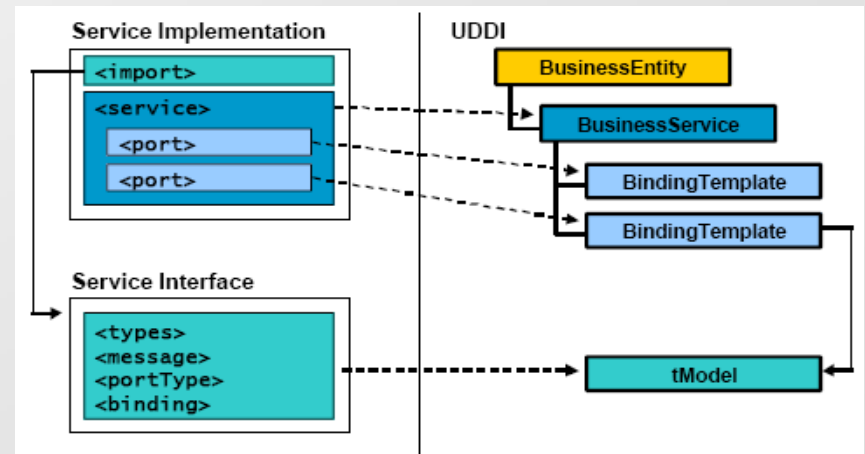
# UDDI Activities

- Using UDDI [7]
  - Establishing Connection
  - Publishing a Business Entity
  - Classifying a Business Entity
  - Publishing a Model
  - Publishing a Service
  - Deleting from a Registry

# UDDI Data Types and Mapping

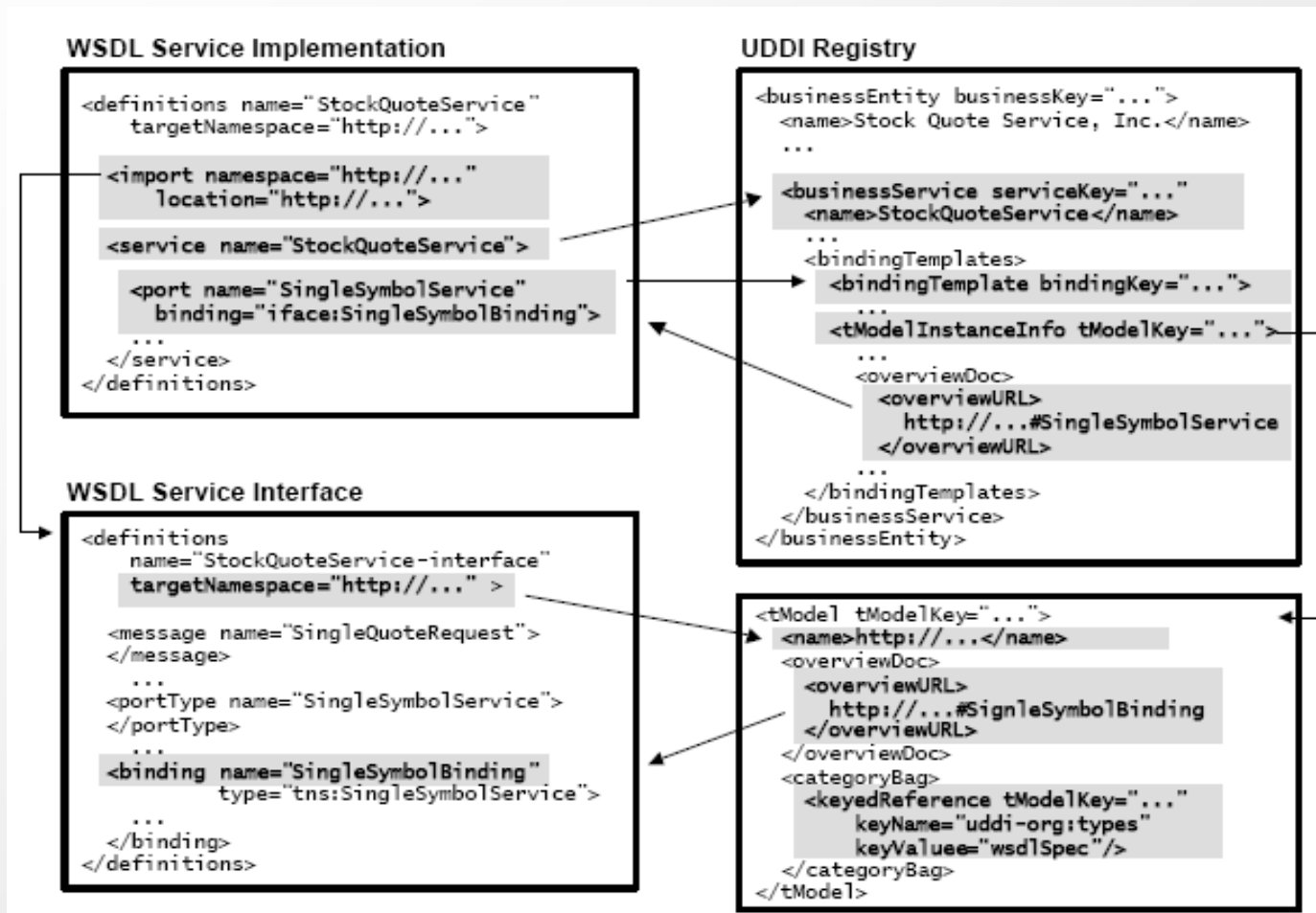


UDDI DataTypes

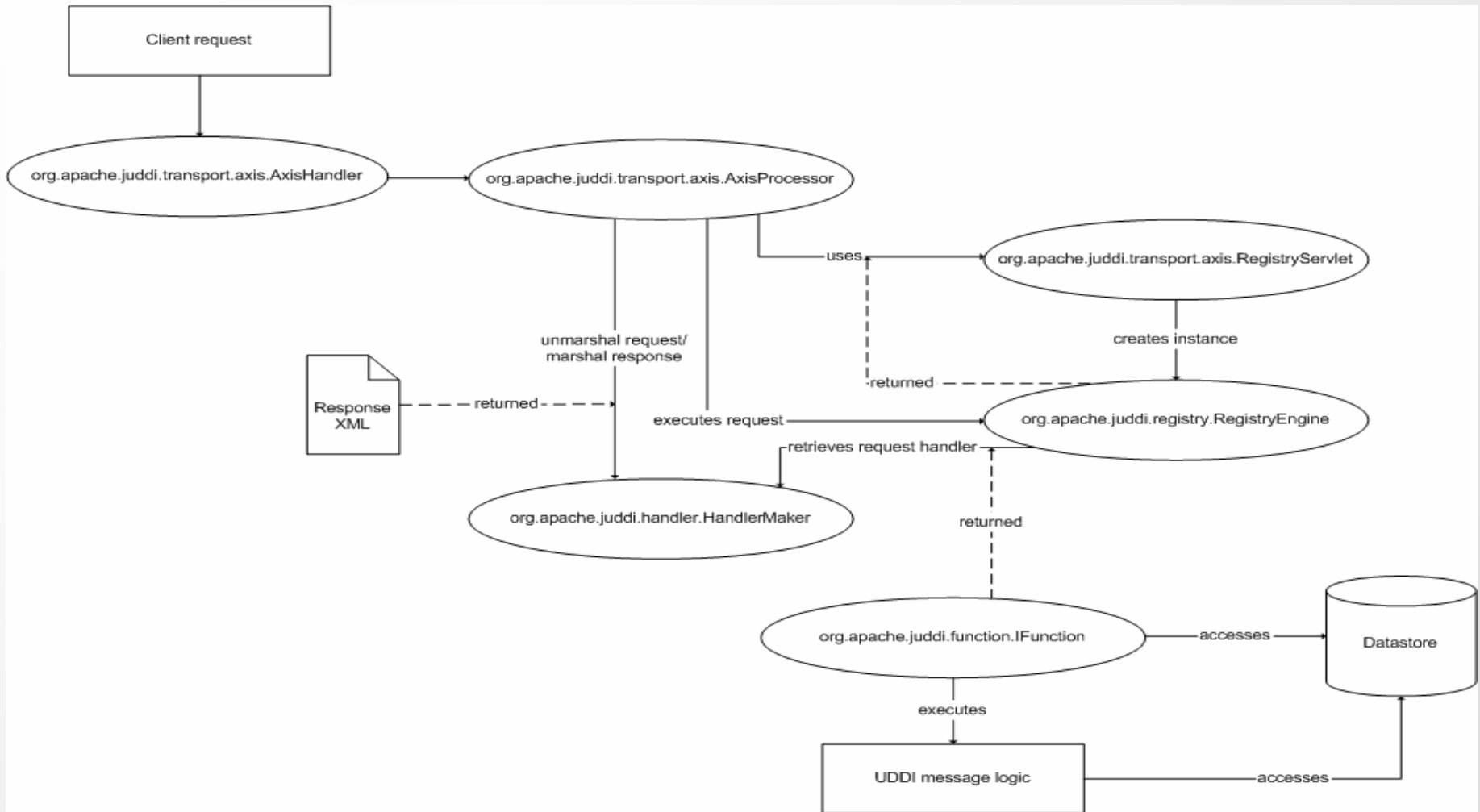


WSDL to UDDI Mapping

# WSDL to UDDI Mapping



# UDDI Process Flow



# Tools Used in Examples

- Apache Software Foundation
  - [Xerces](#)
    - XML Schema processor
  - [SOAP](#) Implementation (Axis Project is related)
    - Library to enable invocation of SOAP services
  - [Tomcat](#)
    - Servlet Container which is the official Reference Implementation for the [Java Servlet](#) and [JavaServer Pages](#) technologies

# Servlets and Containers

- Servlet
  - A Java program that runs within a JVM on a web server and sends HTML data to a web browser
  - Implements [javax.servlet.Servlet](#)
- Servlet Container
  - Web Server Component that interacts with Servlets
  - Responsible for forwarding the requests and responses to the appropriate Servlet
  - Maps URL's to Servlets

# Servlet Containers

- How Java Web Servers Work
  - HTTPServer Class
- How Servlet Containers Work
  - Servlet is called for the first time
    - Load the servlet class
    - Call its init method (once only).
  - For each request
    - Construct an instance of javax.servlet.ServletException
    - Construct an instance of javax.servlet.ServletResponse.
  - Invoke the servlet's service method
    - Pass the ServletException and ServletResponse objects to the servlet
  - When the servlet class is shut down
    - Call the servlet's destroy method and unload the servlet class.
- “How Tomcat Works”
  - Entire book on the internal workings of Tomcat



# Web Service Examples

- Automobile Finder Web Service
- WSDL Generation and Stub Generation
- Semantic Web Service Finder
- UDDI Publishing

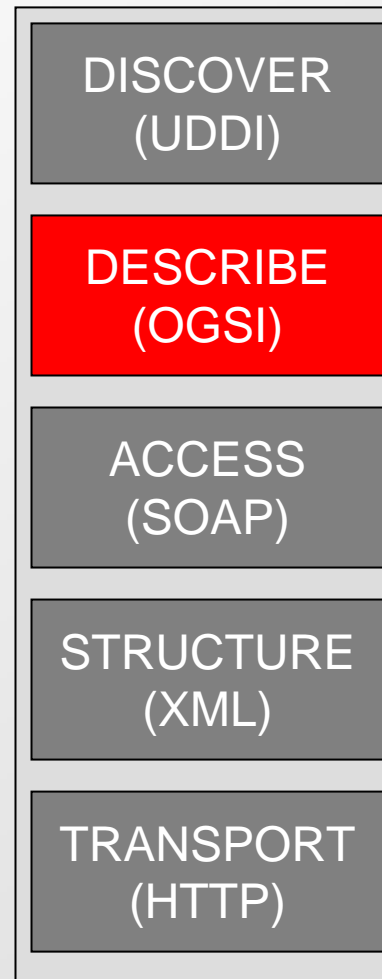
# Grid Services

- All Grid Services are Web Services
  - Vice versus not true
- “A web service that conforms to a set of conventions (interfaces and behaviors) that define how a client interacts with it”

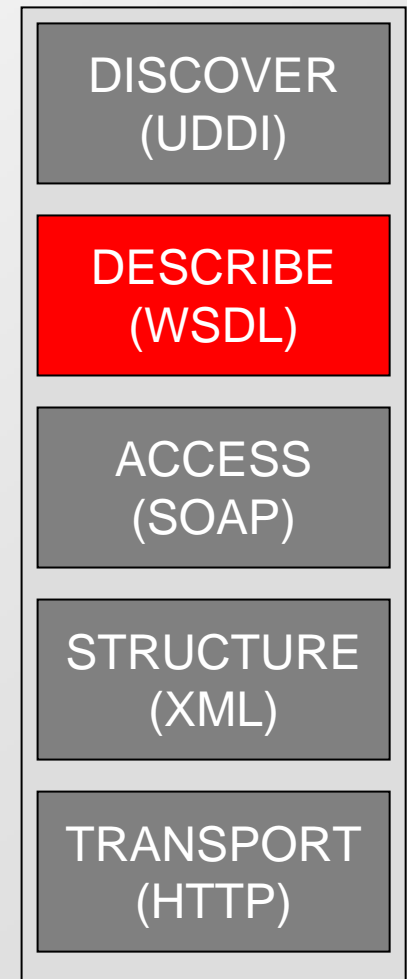
# Grid Services

- OGS I Describe Grid Services
  - Open Grid Services Infrastructure
  - Extension upon WSDL
  - Base for Open Grid Services Architecture (OGSA)

## Grid Services



## Web Services



- Defines Grid Services (since 2002)
  - WSDL 1.1 extensions (to be in WSDL 2.0)
  - Discovery (Grid Handles and References)
  - Dynamic Service Creation
  - Lifetime Management
  - Notification
  - Manageability
  - Fault Conventions

# Existing Grid Services (1/2)

- GILDA

SERVICE	HOST
Resource Broker (RB)	<a href="https://grid004.ct.infn.it">grid004.ct.infn.it</a>
Resource Broker for DAG (RB)	<a href="https://grid007.ct.infn.it">grid007.ct.infn.it</a>
Information Index (BDII)	<a href="https://grid017.ct.infn.it">grid017.ct.infn.it</a>
Backup BDII	<a href="https://grid018.ct.infn.it">grid018.ct.infn.it</a>
LDAP (for GILDA VOs)	grid-
GridICE	vo.cnaf.infn.it:10389 alifarm7.ct.infn.it:50080
Replica Location Service (RLS)	<a href="https://grid008.ct.infn.it">grid008.ct.infn.it</a>

Welcome to the GENIUS INFN GRID Portal - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media

Address <https://grid-demo.ct.infn.it/> Go Links

**INFN**  
Istituto Nazionale di Fisica Nucleare

- HadronTherapy Services
- Video on Demand
- Raster-3D
- GEANT4 Examples
- Other Job Services
- Data Services
- Back

powered by  
**EnginFrame 3.2**  
compliant with  
**LCG-2**  
**GRID.IT**

**enginframe**

**genius**

**eGEE**  
Enabling Grids for  
E-science in Europe

Grid Enabled web eNvironment for site Independent User job

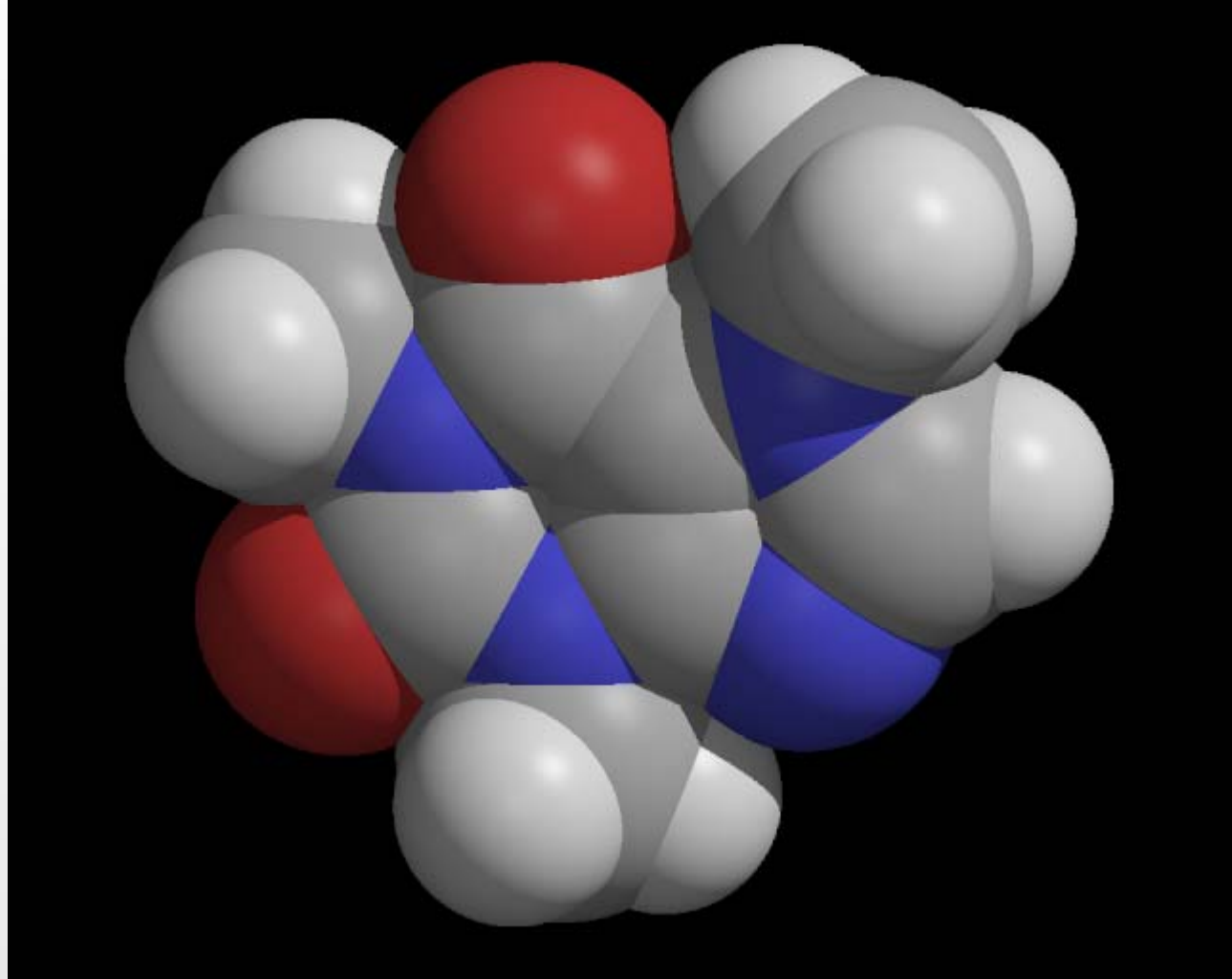
**Welcome to GILDA Services**

**GILDA**

**GRID INFN LABORATORY for  
DISSEMINATION ACTIVITIES**

# Existing Grid Services (2/2)

- GILDA  
raster  
image



# Reference

- [1] - Semantic Web Book
- [2] - IBM PPT
- [3] - monty
- [4] - GILDA <https://gilda.ct.infn.it/>
- [5] -  
[http://www.w3schools.com/wsdl/wsdl\\_documents.asp](http://www.w3schools.com/wsdl/wsdl_documents.asp)
- [6]-  
[http://www.onjava.com/pub/a/onjava/excerpt/jws\\_6/index1.html](http://www.onjava.com/pub/a/onjava/excerpt/jws_6/index1.html)
- [7] -  
<http://www.phptr.com/articles/article.asp?p=101595&seqNum=2>