Semantic Web Services: Approaches and Applications

John Domingue
Knowledge Media Institute,
The Open University

Some Slides Adapted from

• Mary Rowlatt, Leticia Gutierrez
  – Essex County Council
• Amit Sheth
  – Kno.e.sis Center, Wright State University
• Kunal Verma
  – Accenture Technology Labs
• Jacek Kopecký
  – STI Innsbruck
• Stephan Haller
  – SAP
Web Services

What’s a Web Service?

• A program programmatically accessible over standard internet protocols
• Loosely coupled, reusable components
• Encapsulate discrete functionality
• Distributed
• Add new level of functionality on top of the current web
Web Services Framework

The Sixth Summer School on Ontological Engineering and the Semantic Web (SSSW'08)

The Future Internet: Towards a Web-based Service Economy

SAP RESEARCH
(SAP 2008, page 1)
Welcome to Amazon Web Services

Amazon Web Services provides developers with direct access to Amazon's robust technology platform. Build on Amazon's suite of web services to enable and enhance your applications. We innovate for you, so that you can innovate for your customers. Browse developer innovations in our Solutions Catalog to see the possibilities!

What’s New?

High-CPU Instances Now Available (May 29, 2008)
A new family of Amazon EC2 instances types is now available: the High-CPU family. These instances have proportionally more CPU resources than RAM (compared to our Standard Instances) and are well suited for compute-intensive applications such as rendering, search indexing, and computational analysis.

Since this was a popular feature request among EC2 developers, we are excited to offer these new instances to complement our existing instance sizes. Read more about the Amazon EC2 Instance Types.

New Functionality Added to Amazon FPS Widgets (May 13, 2008)
Amazon FPS has released a new Marketplace Widget, which allows developers to easily move money between two third parties, and a new Marketplace Fee feature enables them to charge a fee for doing so. In addition, new functionality has been added to the "Pay Now" Widget allowing developers to quickly integrate with Amazon FPS. This new functionality includes Reserve and Settle, Instant Payment Notification, Refund API, and more.

OpenSolaris and MySQL Enterprise on Amazon EC2 (May 8, 2008)
Sun Microsystems and Amazon Web Services are piloting to offer two open source solutions on Amazon EC2: OpenSolaris and MySQL technical support. With OpenSolaris OS on Amazon EC2, you have access to a robust operating system on a scalable, cost-effective virtual computing environment. And now, MySQL Enterprise Edition on Amazon EC2, allowing MySQL performance to scale to meet your business needs.
GigaVox Architecture

[Diagram of GigaVox Architecture]

STARDUST aerogel collector

[Image of STARDUST aerogel collector with a comet particle]
ClickWorkers (2/2)

Artificial Artificial Intelligence
Problems with Web Services Today

- Descriptions are syntactic
- All tasks associated with web services application development have to be carried out by humans:
  - discovery, composition and invocation
- Problems of scalability

SWS Vision

<table>
<thead>
<tr>
<th>Dynamic</th>
<th>Static</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Services</td>
<td>Semantic Web Services</td>
</tr>
<tr>
<td>(UDDI, WSDL, SOAP)</td>
<td>(RDF, OWL)</td>
</tr>
<tr>
<td>Syntax</td>
<td>Semantics</td>
</tr>
<tr>
<td>Web (URI, HTML, HTTP)</td>
<td>Semantic Web (RDF, OWL)</td>
</tr>
</tbody>
</table>
Semantic Web Services (is)

- Semantic Web Technology
  - Machine readable data
  - Ontological basis

Applied to

- Web Services Technology
  - Reusable computational resources

To automate all aspects of application development through reuse

Semantic Web Service Broker

Client → Semantic Execution Environment → Services
Web Service Modelling Ontology (WSMO)

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 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

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

Non-functional Properties

- Advertising of Web Service
- Support for WS Discovery

Capability

functional description

realization of functionality by aggregating other Web Services
- functional decomposition
- WS composition

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

Choreography --- Service Interfaces --- Orchestration

client-service interaction interface for consuming WS
- External Visible Behavior
- Communication Structure
- ‘Grounding’

DC + QoS + Version + financial

WS
WS
WS

WSMO Web Service Description

- complete item description
- quality aspects
- Web Service Management

Non-functional Properties

- Advertising of Web Service
- Support for WS Discovery

Capability

functional description

realization of functionality by aggregating other Web Services
- functional decomposition
- WS composition

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

Choreography --- Service Interfaces --- Orchestration

client-service interaction interface for consuming WS
- External Visible Behavior
- Communication Structure
- ‘Grounding’

DC + QoS + Version + financial
WSMO Web Service Description

- complete item description
- quality aspects
- Web Service Management

Non-functional Properties
DC + QoS + Version + financial

Choreography --- Service Interfaces --- Orchestration

Capability
functional description

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

realization of functionality by aggregating other Web Services
- functional decomposition
- WS composition

- Advertising of Web Service
- Support for WS Discovery

client-service interaction interface for consuming WS
- External Visible Behavior
- Communication Structure
- ‘Grounding’

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

- Advertising of Web Service
- Support for WS Discovery

ontology, knowledge management, and the Semantic Web (SSSW’08)
WSMO Web Service Description

- complete item description
- quality aspects
- Web Service Management

Non-functional Properties
- DC + QoS + Version + financial

Capability
- Advertising of Web Service
- Support for WS Discovery

Web Service Implementation
(realization of functionality by aggregating other Web Services)
- functional description

Choreography --- Service Interfaces --- Orchestration

Non-functional Properties
- complete item description
- quality aspects
- Web Service Management

Capability
- Advertising of Web Service
- Support for WS Discovery

Web Service Implementation
(realization of functionality by aggregating other Web Services)
- functional description

Choreography --- Service Interfaces --- Orchestration

Orchestration Definition

VTA

Flight Request
Flight Request
flight.arrivaltime = hotel.arrivaltime

Hotel-Request
Hotel-Request

Book Flight
Book Flight

Book Hotel

process (control + data flow) of goals
The Sixth Summer School on Ontological Engineering and the Semantic Web (SSSW'08)

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

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

WSMO Mediators Overview
Mediator Structure

Source Component $\rightarrow$ Target Component

WSMO Mediator uses a Mediation Service via Source Component

- as a Goal
- directly
- optionally incl. Mediation

Mediation Services

Building SWS Applications
Generic Application Structure

Presentation

Semantic Web Services (WSMX/IRS-III)

Services Abstraction

Legacy Systems

IT systems DB
Organisation 1

IT systems DB
Organisation 2

Web Application
Web Application
Web Application

WS
WS
WS
WS

WS
WS
WS
WS

WEB
Generic Application Structure

Web Application  | Web Application  | Web Application | Presentation
SWS          | SWS          | SWS          | Semantic Web Services (WSMX/IRS-III)
SWS          | SWS          | SWS          | Services Abstraction
IT systems  | DB           | IT systems  | DB           | Legacy Systems
Organisation 1 | Organisation 2

Semantic Web Services (WSMX/IRS-III)
Services Abstraction
Legacy Systems

Web Application  | Web Application  | Web Application | Presentation
SWS          | SWS          | SWS          | Semantic Web Services (WSMX/IRS-III)
SWS          | SWS          | SWS          | Services Abstraction
IT systems  | DB           | IT systems  | DB           | Legacy Systems
Organisation 1 | Organisation 2

Semantic Web Services (WSMX/IRS-III)
Services Abstraction
Legacy Systems
Generic Application Structure

Supporting Emergency Planning for Essex County Council
Essex County Council

- A large local authority in South East England
- Comprised of 13 boroughs
- Population of 1.3M.

Emergency Planning Context
Emergency planning scenario

Severe weather event

Plan emergency response

WHAT INFORMATION AVAILABLE?

SELECT RELEVANT INFORMATION

Emergency planning scenario
Emergency planning scenario

Severe weather event forecast in the area by Met Office

get event details

Met Office

Identify most appropriate rest centres

get centre contact details

ViewEssex spatial data

District data

get facilities
kitchen, showers, number of beds, cookers, heaters…

get capacity
eMerges Ontologies

Demonstration of Emergency Planning (GIS) Prototype V1
EMerges Prototype Architecture

Affordances = Goals

Accommodation Goal
Environment Goal
Presence Goal

Archetypes
SGIS-Spatial

Google Web Toolkit

SAWSDL and WSMO-Lite
WSMO-Lite in WSDL

WSMO + WSDL
Types of Service Semantics

- **Functional**
  - What the service does
- **Nonfunctional**
  - Implementation or running env. information
- **Behavioral**
  - How to talk to the service
- **Information**
  - What the exchanged data means

(adapted from Amit Sheth)

WSMO-Lite Example

```xml
<wsdl:description>
  <wsdl:types>
    <xs:schema>
      <xs:element name="ReservationRequest"
        sawsdl:modelReference="&ex;Reservation"
        sawsdl:loweringSchemaMapping="&ex;ResMapping.xsparql" … />
    </xs:schema>
  </wsdl:types>

  <wsdl:interface name="HotelReservations"
    sawsdl:modelReference="&ex;AccommodationReservationService">
    <wsdl:operation name="searchForRooms"
      sawsdl:modelReference="&wsdlx;SafeInteraction">
      …
    </wsdl:operation>
    …
  </wsdl:interface>

  <wsdl:service name="RomaHotels" interface="HotelReservations"
    sawsdl:modelReference="&ex;RomaHotelReservationPrecondition
    &ex;ReservationFee" … />
</wsdl:description>
```
SAWSDL – Help From W3C

But: no predefined semantics!

WSMO-Lite vs. WSMO

• WSMO-Lite service model from WSDL
• Goals out of scope
• Ontologies imported
• Mediators out of scope
• No new syntaxes, just a few terms
  – Using the simplest parts of RDFS
• WSMO-Lite is an application of SAWSDL
WSMO-Lite Terms

wl:Ontology           rdf:type        rdfs:Class;
                     rdfs:subClassOf    owl:Ontology.
wl:ClassificationRoot rdf:type        rdfs:Class.
wl:NonFunctionalParameter rdf:type    rdfs:Class.
wl:Condition          rdfs:subClassOf  wl:Axiom.
wl:Effect             rdfs:subClassOf  wl:Axiom.
wl:Axiom              rdf:type        rdfs:Class.

- Allows different types of semantics on a single component
- E.g. functionality and nonfunc. property on a service

MicroWSMO
What are RESTful WS?

- Like Web applications, but for machines
- Or like WS-*, but with more resources

- A RESTful Web service is:
  - A set of Web resources
  - Interlinked
  - Data-centric, not functionality-centric
  - Machine-oriented (no more HTML-scraping)
Summary

• Semantic Web Services
  – Applies SW to automate application development through reuse of Web services
• WSMO
  – ontology describing Web services
  – Goals, web services, mediators
• Supporting Emergency Planning
• WSMO-Lite
• SAWSDL
• MicroWSMO

Relevant URLs (1/2)

• WSMO
  – http://www.wsmo.org/
• IRS-III
• DIP
  – http://dip.semanticweb.org/
• Emergency Planning Use Case
  – http://irs-test.open.ac.uk/sgis-dev/
Relevant URLs (2/2)

- Conceptual Models of Services
  - http://cms-wg.sti2.org/home/
- OWL-S
  - http://www.daml.org/services/owl-s/
- SAWSDL
  - http://www.w3.org/2002/ws/sawsdl/

Thanks