Services, Semantics and Linked Data

John Domingue
Knowledge Media Institute, The Open University & STI International
Agenda

• Software and Services
• A Short History of Semantic Web Services
• Linked Data Effect
• Rise of REST
• Linked Data and Services
• Example Applications
• Current and Future W3C work
• Summary
SOFTWARE AND SERVICES
Software

• Slow to produce!
  – 10 lines of code per day

• Code is ubiquitous
  – Car has 2 million lines of code (8,000 Person Years)
  – Windows XP 45 million lines of code (180,000 Person Years)

• Reusability is important
Service (economic)

- Intangible commodity
  - Consulting, childcare, educational
- Tertiary sector of the economy
  - Provision of value-adding services
  - Complementing
    - secondary sector manufacturing and
    - primary sector: agriculture, fishing, mining …
  - 63% of Global GDP $44 trillion
Service (computational)
Web Service Usage Process

• Deployment
  – create and publish Web service description

• Discovery
  – determine usable services for a task

• Composition
  – combine services to achieve a goal

• Selection
  – choose most appropriate service

• Mediation
  – solve mismatches (data, protocol, process) that hamper interoperation

• Execution
  – invoke Web services following programmatic conventions
Web Service Execution Support

- **Monitoring**
  - control the execution process

- **Compensation**
  - provide transactional support and undo or mitigate unwanted effects

- **Replacement**
  - facilitate the substitution of services by equivalent ones

- **Auditing**
  - verify that service execution occurred in the expected way
WSDL Simplified

Web service

\[ f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \]

Operation 1

Operation 2

\vdots

Operation N

input
output
input
output
input
output
A SHORT HISTORY OF SEMANTIC WEB SERVICES
Service Profile (partial)
Class Hierarchies of Services

- **ServiceProfile**
  - **ProductProviding Service**
  - **InfoService**
  - **PhysicalProduct Service**
  - **ActionService**
  - **FeeBased**
  - **Manufacturing**
  - **Repair**
  - **Transportation**

- Information - Product+
- Fee Basis+ Payment Method+
- Physical Product+
- Fee Basis+ Payment Method+
- Transportation Mode+ Geographic Region+
- Delivery Region*
- Delivery Provider*
- Delivery Type
- Name
- Provider
- Average Response Time?
- Manufacturer

Tie in with UNSPSC, etc.
DL Basis for matchmaking
Multiple profiles; multiple hierarchies
Objectives that a client may have when consulting a Web Service

WSMO Top-level Elements

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

(http://www.wsmo.org)
WSMO – the Web service element

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

DC + QoS + Version + financial

Capability
- Advertising of Web Service
- Support for WS Discovery

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

Choreography --- Service Interfaces --- Orchestration

realization of functionality by aggregation
- functional decomposition
- WS composition

client-service interaction interface for consuming WS
- external visible behavior
- communication structure
- ‘grounding’
Semantic Web Service Broker

Client

Services
Orchestration Definition

The diagram illustrates the process (control + data flow) of goals for booking a trip. It shows the steps involved in booking a flight and a hotel, with conditions for when one of the booking options is not available. For example, if the flight cannot be booked, the process proceeds to book the hotel instead, ensuring that the arrival times align.

Specifically:
- If the flight cannot be booked (flight = Ø), the process looks for a hotel (hotel_request).
- If the hotel cannot be booked (hotel = Ø), it checks if the flight's arrival time is the same as the hotel's arrival time (flight.arrivaltime = hotel.arrivaltime).
- If the hotel is available and the flight's arrival time matches the hotel's, the process proceeds to book the hotel (book_hotel).

The diagram connects these steps with arrows, showing the flow of control and data.
Runtime Orchestration

process (control + data flow) between “states” + communication behavior of orchestrating Web Service
Web Services and APIs in the Cloud

Domain & Task Models

Intelligence at the Interface
LINKED DATA EFFECT
Latest from the forum

Property price-paid data for £50k??
by AdamWynne
14/05/2011 - 14:54

Not finding my published apps
by aizolnai
14/05/2011 - 12:36

Why publish our data?

View all forum posts

Site feed
Opening up government

Looking for something specific, or just want to know more about how Government spends your money? You’ll find over 6900 datasets to help you get answers.

Browse by publisher
- Cabinet Office (43)
- Department for Business, Innovation and Skills (162)
- Department for Communities and Local Government (782)
- Department for Environment, Food and Rural Affairs (418)
- Department for Transport (151)
- Department of Health (1005)
- UK Statistics Authority (716)

Browse by nation
- England
- Northern Ireland
- Scotland
- Wales

Popular tags
- health (1,165)
- care (823)
- transparency (803)
- communities (659)
- child (609)
- health-and-social-care (608)
- children (571)
- local-government (568)

Developers
You can also access this catalogue via the API or download it as a CSV/JSON dump.
Data.gov.uk Applications

Where Does My Money Go?

UK WIDE  NATIONAL  REGIONAL

- Overview
- Long-term trends

Total spending £620bn

- Social protection £237bn
- Health £118bn

- Economic affairs £39bn
- Education £67bn
- General public services £53bn
- Defence £77bn
- Environment protection £1bn
- Public order and safety £13bn
- Housing and community amenities £13bn
- Recreation, culture and religion £13bn
- Public order and safety £13bn

2003-04  2008-09  2008-09
RISE OF REST
RESTful Service

- Lightweight compared to SOAP/WSDL
- Use HTTP methods explicitly (GET, POST, PUT, DELETE)
- Stateless
- Expose directory structure-like URIs
- Data transfer using XML, JavaScript Object Notation (JSON), or both.
## RESTful Service Takeup

<table>
<thead>
<tr>
<th>API</th>
<th>Description</th>
<th>Category</th>
<th>Mashups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Maps</td>
<td>Mapping services</td>
<td>Mapping</td>
<td>1711</td>
</tr>
<tr>
<td>Flickr</td>
<td>Photo sharing service</td>
<td>Photos</td>
<td>446</td>
</tr>
<tr>
<td>YouTube</td>
<td>Video sharing and search</td>
<td>Video</td>
<td>378</td>
</tr>
<tr>
<td>Amazon eCommerce</td>
<td>Online retailer</td>
<td>Shopping</td>
<td>298</td>
</tr>
<tr>
<td>Twitter</td>
<td>Microblogging service</td>
<td>Social</td>
<td>196</td>
</tr>
<tr>
<td>Microsoft Virtual Earth</td>
<td>Mapping services</td>
<td>Mapping</td>
<td>171</td>
</tr>
<tr>
<td>eBay</td>
<td>Online auction marketplace</td>
<td>Shopping</td>
<td>170</td>
</tr>
<tr>
<td>del.icio.us</td>
<td>Social bookmarking</td>
<td>Bookmarks</td>
<td>129</td>
</tr>
<tr>
<td>Yahoo Maps</td>
<td>Mapping services</td>
<td>Mapping</td>
<td>129</td>
</tr>
<tr>
<td>Google Search</td>
<td>Search services</td>
<td>Search</td>
<td>127</td>
</tr>
<tr>
<td>411Sync</td>
<td>SMS, WAP, and email messaging</td>
<td>Messaging</td>
<td>120</td>
</tr>
<tr>
<td>Yahoo Search</td>
<td>Search services</td>
<td>Search</td>
<td>120</td>
</tr>
<tr>
<td>Last.fm</td>
<td>Online radio service</td>
<td>Music</td>
<td>104</td>
</tr>
<tr>
<td>Facebook</td>
<td>Social networking service</td>
<td>Social</td>
<td>91</td>
</tr>
<tr>
<td>Google Homepage</td>
<td>Portal gadgets</td>
<td>Widgets</td>
<td>87</td>
</tr>
<tr>
<td>Yahoo Geocoding</td>
<td>Geocoding services</td>
<td>Mapping</td>
<td>87</td>
</tr>
<tr>
<td>Google Ajax Search</td>
<td>Web search components</td>
<td>Search</td>
<td>78</td>
</tr>
<tr>
<td>Technorati</td>
<td>Blog search services</td>
<td>Blog Search</td>
<td>65</td>
</tr>
</tbody>
</table>
REST vs SOAP API Searches

Google Trends
LINKED DATA AND SERVICES
Linked Open Services

http://www.linkedopenservices.org/
MicroWSMO & WSMO-Lite

WSMO-Lite Ontology

Annotations point to:

SAWSDL

Layer of semantic annotations

WSDL

Service description layer

MicroWSMO

Extension

hRESTS

Extension
WSDL
SAWSDL

But: no predefined semantics!
WSMO-Lite in WSDL
MicroWSMO

• Built on top of hRESTS which introduces the service model structure on Web pages
  – service (+ label)
  – operations (+ address, method)
  – input, output

• MicroWSMO extends hRESTS
  – model, lifting, lowering

• Applies WSMO-Lite semantics
Last.fm Web Services

API | Feeds | Your API Account

album.getInfo

Get the metadata for an album on Last.fm using the album name or a musicbrainz id. See playlist.fetch on how to get the album playlist.

Example URL:
http://ws.last.fm/user/JohnDoe/album.getInfo

Params

- artist (Optional) : The artist name in question
- album (Optional) : The album name in question
- mbid (Optional) : The musicbrainz id for the album
- username (Optional) : The username for the context of the request. If supplied, the user's playcount for this album is included in the response.
- lang (Optional) : The language to return the biography in, expressed as an ISO 639 alpha-2 code.
- api_key (Required) : A Last.fm API key.

Auth

This service does not require authentication.

Sample Response

```xml
<album>
  <name>Believe</name>
  <artist>Cher</artist>
  <mbid>61bf0388-b8a9-48f4-91d1-7eb02706dfb0</mbid>
  <url>http://www.last.fm/music/Cher/Believe</url>
  <releasedate>6 Apr 1999, 00:00</releasedate>
  <image size="small">.../image</image>
</album>
```
Last.fm Web Services

API | Feeds | Your API Account

album.getInfo

Get the metadata for an album on Last.fm using the album name or a musicbrainz id. See playlist.fetch on how to get the album playlist.

e.g. http://ws.audioscrobbler.com/2.0/?method=album.getInfo&api_key=b25b959554ed76058ac220b7b2e0a026...

Params

artist (Optional) : The artist name in question
album (Optional) : The album name in question
mbid (Optional) : The musicbrainz id for the album
username (Optional) : The username for the context of the request. If supplied, the user's playcount for this album is included in the response.
lang (Optional) : The language to return the biography in, expressed as an ISO 639 alpha-2 code.
api_key (Required) : A Last.fm API key.

Auth

This service does not require authentication.

Sample Response

```xml
<album>
  <name>Belleve</name>
  <artist>Cher</artist>
  <id>2b261281</id>
  <mbid>61b0388-b8e948f4-81d1-7eb02706dfb0</mbid>
  <url>http://www.last.fm/music/Cher/Belleve</url>
  <releasedate>6 Apr 1999, 00:00</releasedate>
  <image size="small">... </image>
  <image size="medium">... </image>
  <image size="large">... </image>
  <albumartist>Cher</albumartist>
  <description></description>
  <playcount>2124445</playcount>
</album>
```
Last.fm Web Services

API Docs

album.getInfo

Get the metadata for an album on Last.fm using the album name or a musicbrainz id. See playlist.fetch on how to get the album playlist.

e.g. http://ws.audioscrobbler.com/2.0/?method=album.getInfo&api_key=b25b959554ed76058ac220b7b2e0a026..

Params

artist (Optional) : The artist name in question
album (Optional) : The album name in question
mbid (Optional) : The musicbrainz id for the album
username (Optional) : The username for the context of the request. If supplied, the user's playcount for this album is included in the response.
lang (Optional) : The language to return the biography in, expressed as an ISO 639 alpha-2 code.
api_key (Required) : A Last.fm API key.

Auth

This service does not require authentication.

Sample Response

```xml
<album>
  <name>Believe</name>
  <artist>Cher</artist>
  <id>2026125</id>
  <mbid>61bf0388-b9a9-48f4-81d1-7eb02706dfb6</mbid>
  <url>http://www.last.fm/music/Cher/Believe</url>
  <releasedate>6 Apr 1999, 00:00</releasedate>
  <image size="small"/>
  <image size="medium"/>
  <image size="large"/>
  <listenercount>47602</listenercount>
  <playcount>21299</playcount>
</album>
```
Service

Operation

Input

Parameter

MicroWSMO

Last.fm Web Services
API | Feeds | Your API Account

album.getInfo

The metadata for an album on Last.fm using the album name or a musicbrainz id. See the fetch on how to get the album playlist.

e.g. http://ws.audioscrobbler.com/2.0/?method=album.getInfo&api_key=b25b95954ed76058ac220b7b2e0a026...

Params

artist (Optional) : The artist name in question
album (Optional): The album name in question
mbid (Optional): The musicbrainz id for the album
username (Optional): The username for the context of the request. If supplied, the user’s playcount for this album is included in the response.
lang (Optional): The language to return the biography in, expressed as an ISO 639 alpha-2 code.
api_key (Required): A Last.fm API key.

Auth

This service does not require authentication.

Sample Response

<!-- XML content -->
Minimal Service Model

Namespace Abbreviations

xsd: http://www.w3.org/2001/XMLSchema
owl: http://www.w3.org/2002/07/owl#
sawSDL: http://www.w3.org/ns/sawSDL#
msm: http:// ims-wg.st2.org/ns/minimal-service-model#
rest: http://www.wsno.org/ns/rests#
http: http://www.w3.org/2009/hypermedia
wl: http://www.wsno.org/ns/wsno-lite#
Service Template

Service Template

- hasInput
- hasOutput
- hasFunctionalCategory
- hasPreference
- hasRequirement
- hasContextualSens.

RDF-based domain ontology
- e.g., RDFS/WSML-core, OWL/WSML-DL, WSML-Rule

RDFS/WSML-Core taxonomy
- e.g., ServiceFinder

RDFS/WSML-Core preference model
- e.g., WSMO Pref, U-Sevilla

RDF-based context ontology
- e.g., RDFS/WSML-core, OWL/WSML-DL, WSML-Rule

WSMO-Lite annotation
- e.g., Condition, Effect

WSML Flight/Rule logical expression
Resource-Orientedness (ROSM)

ServicedResource

ServicedResource Collection

Service

Operation

supports

Request

Response

Parameter

Optional Parameter

Required Parameter

1. requestURIParameter
2. requestBodyParameter
3. requestHeaderParameter
4. hasBody

Method

Response Code

HTTP *

* http://www.w3.org/2006/http#
EXAMPLE APPLICATIONS
Example 1: House Hunting ...
... public transport ...
... schools
SOA4All Real Estate Finder

Description

SOA4RealEstate is a user friendly mobile client for finding real estate in desired areas developed within EU SOA4ALL project.

SOA4RealEstate allows users to make educated decisions about property he or she would like to buy or to rent based on the information coming from multiple resources including semantically enriched data storages such as ones coming from the UK government open data initiative. For example, at the moment SOA4RealEstate visualizes data about houses available, train stations, bus stations, schools on one map view.

SOA4RealEstate team tries to make your experience as pleasant as possible hiding behind a hood a lot of fancy and complicated technologies, including Semantic Web and Web services. Visit http://www.soa4all.eu/ for more information about SOA4ALL and underlying technologies.

What's New In Version 2.0

Application is fully integrated with SOA4All underlaying Technologies and bugfixes (as the old version was crashing some times).

iPhone Screenshots

[Images of iPhone screenshots showing the application interface]
Behind the Scenes

Public Data and Services

- Real estate
- Train stations
- Bus stops
- Schools

Service Broker

- Publishing
- Discovery
- Invocation

Invocation Engine
CURRENT AND FUTURE WORK
The Future Internet
– Enabler for Global Business Networks

Network of the Future

Suppliers

Manufacturers

Wholesalers

Carriers

Retailers

Consumers

Governments

Internet of the Future

Internet of Services

Internet of Things

Cloud Computing

Network of the Future

Transport Logistic

eEnergy

Urban Management

Manufacturing

© SAP 2010
The Internet of Services
– Global Service Delivery Supply Chain

SaaS, On-Demand Enterprise Suites

Cloud Services

B2B Gateways

Service Delivery Framework

Business Process Platform

Service-Oriented Architecture

Business Process Outsourcing

Service Provider

Service Hoster

Service Gateway

Service Aggregator

Service Broker

Service Channel Maker

Multi-Enterprise Business process platforms

Service Marketplaces

© SAP 2010 /
Service Transformation stands for a value-driven, smooth and effective provision of services along the Global Service Delivery Supply Chain.

Service Transformation implies that Services are being:
- **Described** considering business, operational and legal requirements
- **Maintained**, extended and assembled where needed
- Applying a **common notation** named USDL
W3C USDL Incubator

http://www.linked-usdl.org/
This group is based on the idea of combining two Web-related concepts to help solve some of the long-standing challenges involved in building and combining software:

1. RDF, the Resource Description Framework.... RDF is the basis for Linked Data and the Semantic Web.

2. With RESTful APIs and RESTful Web Services, ....

The Linked Data Platform is envisioned as an enterprise-ready collection of standard techniques and services based on using RESTful APIs and the W3C Semantic Web stack.
Summary

• Web services successful history in corporate settings
  – Supporting reuse of software components
• Semantic Web services
  – Lots of research
  – Takeup limited
• New trends
  – RESTful services
  – Linked Data portals/applications
• Manage applications which make use of these trends
• Fit to the goals, needs and contexts of users
Acknowledgements