Semantic Web Services

Introduction

What is the course about?

- New, emerging sciences: web science, service science
- Service based technologies: Web services, Web2.0/Restful services
- Semantic Web services: vision, approaches, usage

Course Organization

- Course is organized as follows:
  - Lecture every Monday 12:15-14:00
  - Tutorial every Wednesday 13:15-15:00
- The lecturers is:
  - Dieter Fensel
    (dieter.fensel@sti2.at)
- The tutors are:
  - Srdjan Komazec and Iker Larizgoitia

Course material

- Web site:
  - http://www.sti-innsbruck.at/teaching/curriculum/semantic-web-services/
  - http://www.sti-innsbruck.at/teaching/course-schedule/ss2011/details/?title=semantic-web-services
- Slides available online
  - https://lists.sti2.at/mailman/listinfo/sws2011
• Exam grade:

<table>
<thead>
<tr>
<th>score</th>
<th>grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>75-100</td>
<td>1</td>
</tr>
<tr>
<td>65-74.9</td>
<td>2</td>
</tr>
<tr>
<td>55-64.9</td>
<td>3</td>
</tr>
<tr>
<td>45-54.9</td>
<td>4</td>
</tr>
<tr>
<td>0-44.9</td>
<td>5</td>
</tr>
</tbody>
</table>

• Tutorial and Exam have separate grades since this is not an integrated course

---

**Where are we?**

<table>
<thead>
<tr>
<th>#</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
</tr>
<tr>
<td>2</td>
<td>Web Science</td>
</tr>
<tr>
<td>3</td>
<td>Service Science</td>
</tr>
<tr>
<td>4</td>
<td>Web services</td>
</tr>
<tr>
<td>5</td>
<td>Web2.0 services</td>
</tr>
<tr>
<td>6</td>
<td>Semantic Web</td>
</tr>
<tr>
<td>7</td>
<td>Web Service Modeling Ontology (WSMO)</td>
</tr>
<tr>
<td>8</td>
<td>Web Service Modeling Language (WSML)</td>
</tr>
<tr>
<td>9</td>
<td>Web Service Execution Environment (WSMEE)</td>
</tr>
<tr>
<td>10</td>
<td>OWL-S and other</td>
</tr>
<tr>
<td>11</td>
<td>Light-weight Annotations</td>
</tr>
<tr>
<td>12</td>
<td>Applications</td>
</tr>
<tr>
<td>13</td>
<td>Mobile Services</td>
</tr>
</tbody>
</table>
SEMANTIC WEB

http://www.sti-innsbruck.at/results/movies/serviceweb30-the-future-internet/

Motivation

The Future Internet: Service Web 3.0 Video

http://www.sti-innsbruck.at/results/movies/dip-promotion-movie

The traditional Web

More than a 2 billion users
more than 50 billion pages

Static

WWW

URI, HTML, HTTP
Semantics in information finding, extraction, representation, interpretation, and maintenance. 

Static WWW (URI, HTML, HTTP) → Semantic Web (RDF, RDF(S), OWL)

- "An extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."
  - Sir Tim Berners-Lee et al., Scientific American, 2001: tinyurl.com/i59p
  - “…allowing the Web to reach its full potential…” with far-reaching consequences
  - "The next generation of the Web"

- The next generation of the WWW
- Information has machine-processable and machine-understandable semantics
- Not a separate Web but an augmentation of the current one
- Ontologies as basic building block

- Web Data Annotation
  - connecting (syntactic) Web objects, like text chunks, images, … to their semantic notion (e.g., this image is about Innsbruck, Dieter Fensel is a professor)
- Data Linking on the Web (Web of Data)
  - global networking of knowledge through URI, RDF, and SPARQL (e.g., connecting my calendar with my rss feeds, my pictures, …)
- Data Integration over the Web
  - Seamless integration of data based on different conceptual models (e.g., integrating data coming from my two favorite book sellers)
Semantic Web - Ontologies

- Concept
  - conceptual entity of the domain
- Property
  - attribute describing a concept
- Relation
  - relationship between concepts or properties
- Axiom
  - coherency description between Concepts / Properties / Relations via logical expressions

To make the Semantic Web working we need:

- **Ontology Languages:**
  - expressivity
  - reasoning support
  - web compliance
- **Ontology Reasoning:**
  - large scale knowledge handling
  - fault-tolerant
  - stable & scalable inference machines
- **Ontology Management Techniques:**
  - editing and browsing
  - storage and retrieval
  - versioning and evolution Support
- **Ontology Integration Techniques:**
  - ontology-mapping, alignment, merging
  - semantic interoperability determination

"Semantic Web Language Layer Cake"
### Web Services: Definition

1. “Loosely coupled, reusable software components that encapsulate discrete functionality and are distributed and programmatically accessible over standard Internet protocols”, The Stencil Group
2. Web service applications are encapsulated, loosely coupled Web "components" that can bind dynamically to each other, F. Curbera
3. “Web Services are a new breed of application. They are self-contained, self-describing, modular applications that can be published, located, and invoked across the Web. Web Services perform functions, which can be anything from simple request to complicated business processes”, The IBM Web Services tutorial

Common to all definitions:
- Components providing functionality
- Distributed
- Accessible over the Web

### Web Services

- Loosely coupled, reusable components
- Encapsulate discrete functionality
- Distributed
- Programmatically accessible over standard internet protocols
- Add new level of functionality on top of the current web
Web Service vs. Service

- **Service**
  - A provision of value in some domain (not necessarily monetary, independent of how service provider and requestor interact)

- **Web Service**
  - Computational entity accessible over the Internet (using Web Service Standards & Protocols), provides access to (concrete) services for the clients.

The Service Society

80% of jobs can be found in the service sector

Service Dimensions

- From "Others" to 80% of business activity
- The productivity of production and provisioning of services is therefore of high importance for the overall productivity of a developed economy
- Like in the primary and secondary sector also here information and communication technologies will be very important
- The usage of modern ICT in the service area is called internet of services

Definitions

Def 1. Software Architecture

Def 2. New concept for eWork and eCommerce

Def 3. New programming technology
Definitions

Def 1. Software architecture

- Web Services connect computers and devices with each other using the Internet to exchange data and combine data in new ways.
- The key to Web Services is on-the-fly software creation through the use of loosely coupled, reusable software components.
- Software can be delivered and paid for as fluid streams of services as opposed to packaged products.

Def 2. Web Services as a new Concept for eWork and eCommerce

- Business services can be completely decentralized and distributed over the Internet and accessed by a wide variety of communications devices.
- The internet will become a global common platform where organizations and individuals communicate among each other to carry out various commercial activities and to provide value-added services.
- The dynamic enterprise and dynamic value chains become achievable and may be even mandatory for competitive advantage.

Def 3. Web Services as a programming technology

Web Services are Remote Procedure Calls (RPC) over HTTP
**WSDL**

- Web Service Description Language

  describes interface for consuming a Web Service:
  - Interface: operations (in- & output)
  - Access (protocol binding)
  - Endpoint (location of service)

**SOAP**

- Simple Object Access Protocol
- W3C Recommendation

  XML data transport:
  - sender / receiver
  - protocol binding
  - communication aspects
  - content

**UDDI**

- Universal Description, Discovery, and Integration Protocol
- OASIS driven standardization effort

  Registry for Web Services:
  - provider
  - service information
  - technical access

**Restful services**

- Another way of realizing services, other than SOAP/WSDL/UDDI approach
- Follows the Web principles (REST principles)

  Services expose their data and functionality through resources identified by URI
  Services are Web pages that are meant to be consumed by an autonomous program
  Uniform interfaces for interaction: GET, PUT, DELETE, POST
  HTTP as the application protocol
People as a Service
Amazon - Mechanical Turk

“People as a service”

- Amazon Mechanical Turk
  - An API to Human Processing Power
  - The Computer Calls People
  - An Internet Scale Workforce
  - Game-Changing Economics

Infrastructure as a Service
Amazon – S3 & EC2

“Infrastructure as a service”

- Amazon Simple Storage Service (S3)
  - Write and read objects up to 5GB
  - 15 cents GB/month to store
  - 20 cents GB/month to transfer
- Amazon Elastic Compute Cloud (EC2)
  - Allows customers to rent computers on which to run their own computer applications
  - Virtual server technology
  - 10 cents/hour

Data as a Service
Google – Unified Cloud Computing

- An attempt to create an open and standardized cloud interface for the unification of various cloud APIs
- Key drivers of the unified cloud interface: to create an API about other APIs
- Use of the resource description framework (RDF) to describe a semantic cloud data model (taxonony & ontology)

Semantic Web Services
Semantic Web Services

Dynamic

Web Services
UDDI, WSDL, SOAP

→ Semantic Web Services

Static

WWW
URI, HTML, HTTP

→ Semantic Web
RDF, RDF(S), OWL

Deficiencies of WS Technology

• current technologies allow usage of Web Services
• but:
  – only syntactical information descriptions
  – syntactic support for discovery, composition and execution
  ⇒ Web Service usability, usage, and integration needs to be inspected manually
  – no semantically marked up content / services
  – no support for the Semantic Web

⇒ current Web Service Technology Stack failed to realize the promise of Web Services

So what is needed?

• Mechanized support is needed for
  – Annotating/designing services and the data they use
  – Finding and comparing service providers
  – Negotiating and contracting services
  – Composing, enacting, and monitoring services
  – Dealing with numerous and heterogeneous data formats, protocols and processes, i.e. mediation

⇒ Conceptual Models, Formal Languages, Execution Environments
Semantic Web Technology
- allow machine supported data interpretation
- ontologies as data model

Web Service Technology
automated discovery, selection, composition, and web-based execution of services

=> Semantic Web Services as integrated solution for realizing the vision of the next generation of the Web

Semantic Web Services
- define exhaustive description frameworks for describing Web Services and related aspects (Web Service Description Ontologies)
- support ontologies as underlying data model to allow machine supported data interpretation (Semantic Web aspect)
- define semantically driven technologies for automation of the Web Service usage process (Web Service aspect)

Tasks to be automated
- Service Publishing
  - Describe the service explicitly, in a formal way
  - Make available the description of the service
- Service Enactment & Monitoring
  - Locate different services suitable for a given goal
  - Choose the most appropriate service among the available ones
- Service Composition
  - Combine services to achieve a goal
- Service Negotiation & Contracting
  - Find
  - Compare
  - Compose
  - Invoke

Semantic Web Services
- Semantic Web Services are a layer on top of existing Web service technologies and do not aim to replace them
- Provide a formal description of services, while still being compliant with existing and emerging technologies
- Distinguish between a Web service (computational entity) and a service (value provided by invocation)
- Make Web services easier to:
  - Find
  - Compare
  - Compose
  - Invoke
Semantic Web Services benefits

- Brings the benefits of Semantics to the executable part of the Web
  - Ontologies as data model
  - Unambiguous definition of service functionality and external interface
- Reduce human effort in integrating services in SOA
  - Many tasks in the process of using Web services can be automated
- Improve dynamism
  - New services available for use as they appear
  - Service Producers and Consumers don’t need to know of each others existence
- Improve stability
  - Service interfaces are not tightly integrated so even less impact from changes
  - Services can be easily replaced if they are no longer available
  - Failover possibilities are limited only by the number of available services

Service Oriented Architecture

Semantically Enabled SOA (SESA)

SESA Architecture

SESA functionality

- Middleware for Semantic Web Services
  - Allows service providers to focus on their business.
- Environment for goal based discovery and invocation
  - Run-time binding of service requesters and providers,
  - Provide a flexible Service Oriented Architecture
    - Add, update, remove components at run-time as needed.
- Keep open-source to encourage participation
  - Developers are free to use in their own code, and
- Define formal execution semantics
  - Unambiguous model of system behavior.

Realizing Semantic Web Services Vision

Dynamic
- Web Services
  - UDDI, WSDL, SOAP
- Semantic Web Services

Static
- WWW
  - URI, HTML, HTTP
- Semantic Web
  - RDF, RDF(S), OWL

- Take the WSDL/SOAP web service stack as a starting point and add semantic annotations.

Realizing Semantic Web Services Vision

Dynamic
- Web Services
  - UDDI, WSDL, SOAP
- Semantic Web Services

Static
- WWW
  - URI, HTML, HTTP
- Semantic Web
  - RDF, RDF(S), OWL

- Alternative way to realize Semantic Web Services vision is to focus on further developing the Semantic Web.

Semantic Spaces - Motivation

- Are WSDL/SOAP web services really web services? - No!
- Web services require tight coupling of the applications they integrate.
  - Applications communicate via message exchange requiring strong coupling in terms of reference and time.
- The Web is strongly based on the opposite principles. Information is published in a persistent and widely accessible manner.
  - Any other application can access this information at any point in time without having to request the publishing process to directly refer to it as a receiver of its information.
- Web services can use the Web as a transport media, however that is all they have in common with the Web.
Semantic Spaces - Motivation

- Distributed systems dominated by **messaging**
  - Web services / SOAP
  - CORBA / RPC / RMI / MOM
  - Agents
- Web architecture different
  - **Persistent publication** as the main principle
  - Uniform interface
  - Uniform addressing
- Web clearly scales to a large size

Semantic Spaces - Space-based Communication

Semantic Spaces

- **Persistent publication** of semantic data
- Retrieval by **semantic matching**
- **Mediation** of data between heterogeneous services
- Semantics-aware **distribution** of data
- **Coordination** of concurrent access situations
- Appropriate **security and trust** mechanisms
- Use of **Web service protocol stack** and **Semantic Web** technologies

LOD Cloud March 2009

Linked Data, [http://linkeddata.org/](http://linkeddata.org/) (last accessed on 18.03.2009)
Data Linking on the Web

• Linked Open Data statistics:
  – data sets: 121
  – total number of triples: 13,112,409,691
  – total number of links between data sets: 142,605,717

• Statistics available at (last accessed on 04.02.2010):
  – http://esw.w3.org/topic/TaskForces/CommunityProjects/LinkingOpenData/DataSets/Statistics
  – http://esw.w3.org/topic/TaskForces/CommunityProjects/LinkingOpenData/DataSets/LinkStatistics

Data linking on the Web principles

• Use URIs as names for things
  – anything, not just documents
  – you are not your homepage
  – information resources and non-information resources

• Use HTTP URIs
  – globally unique names, distributed ownership
  – allows people to look up those names

• Provide useful information in RDF
  – when someone looks up a URI

• Include RDF links to other URIs
  – to enable discovery of related information

DBpedia

• DBpedia is a community effort to:
  – Extract structured information from Wikipedia
  – Make the information available on the Web under an open license
  – Interlink the DBpedia dataset with other open datasets on the Web

• DBpedia is one of the central interlinking-hubs of the emerging Web of Data

The DBpedia Dataset

• 91 languages
• Data about 2.9 million “things”. Includes for example:
  – 282,000 persons
  – 339,000 places
  – 119,000 organizations
  – 130,000 species
  – 88,000 music albums
  – 44,000 films
  – 19,000 books
• Altogether 479 million pieces of information (RDF triples)
  – 807,000 links to images
  – 3,840,000 links to external web pages
  – 4,878,100 data links into external RDF datasets
LinkedCT

- LinkedCT is the Linked Data version of ClinicalTrials.org containing data about clinical trials.
- Total number of triples: 6,998,851
- Number of Trials: 61,920
- RDF links to other data sources: 177,975
- Links to other datasets:
  - DBpedia and YAGO (from intervention and conditions)
  - GeoNames (from locations)
  - Binc2RDF.org's PubMed (from references)

SUMMARY

Why Semantic Web Services?

- To overcome limitations of traditional Web-Services Technology by integrating it with Semantic Technology;
- To enable automatic and personalized service discovery;
- To enable automatic service invocation and execution monitoring;
- To enable automatic service integration;
- To enable semantic mediation of Web-Services.

Summary

- Two new sciences are currently emerging: Web science and Service Science.
- Core pillar of these sciences are:
  - Semantic Web
    - the next generation of the Web in which information has machine-processable and machine-understandable semantics.
  - Semantic Web Services
    - overcome limitations of traditional Web-Services Technology using Semantic Technology to enable automatic service discovery, ranking, selection, composition, etc.
REFERENCES

References

• Mandatory reading:

• Further reading:
  – SOAP: http://w3.org/TR/soap12
  – WSDL: http://wsd.org/Weave20
  – UDDI: http://uddi.xml.org/
  – http://dbpedia.org/About

• Wikipedia links:
  – http://en.wikipedia.org/wiki/Web_Services_architecture
  – http://en.wikipedia.org/wiki/Web_Services_Description_Language
  – http://en.wikipedia.org/wiki/Universal_Description_Discovery_and_Integration

Next Lecture

<table>
<thead>
<tr>
<th>#</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
</tr>
<tr>
<td>2</td>
<td>Web Science</td>
</tr>
<tr>
<td>3</td>
<td>Service Science</td>
</tr>
<tr>
<td>4</td>
<td>Web services</td>
</tr>
<tr>
<td>5</td>
<td>Web2.0 services</td>
</tr>
<tr>
<td>6</td>
<td>Semantic Web</td>
</tr>
<tr>
<td>7</td>
<td>Web Service Modeling Ontology (WSMO)</td>
</tr>
<tr>
<td>8</td>
<td>Web Service Modeling Language (WSML)</td>
</tr>
<tr>
<td>9</td>
<td>Web Service Execution Environment (WSME)</td>
</tr>
<tr>
<td>10</td>
<td>OWL-S and other</td>
</tr>
<tr>
<td>11</td>
<td>Lightweight Annotations</td>
</tr>
<tr>
<td>12</td>
<td>Applications</td>
</tr>
<tr>
<td>13</td>
<td>Mobile Services</td>
</tr>
</tbody>
</table>