Semantic Web & Services for Management

Dr. Athanasios Staikopoulos
(Athanasios.Staikopoulos@scss.tcd.ie)
Semantics is …

Provide a well defined meaning, that computers can process

But for us:

**Semantics** = “a representation of the link between a term in a statement to the entity in the world that the term refers to”

- p31, Semantic Web for the Working Ontologist
Semantics vs Syntax …

**Semantics** = a way of encoding meaning (link between term and a model of the world).
⇒ Good for building “intelligent” applications

**Syntax** = a way of encoding terms so that they can be distinguished, structured, grouped and related to each other in a grammar. (Symbolic representation)
⇒ Good for building parsers

Note! We need a syntax (or syntaxes) for expressing a machine-readable semantics. (RDF is the candidate syntax for the Semantic Web)
Structured vs Unstructured Information

**Structured** Information (files, databases…)
- interrelationships in structure is “implicit”
- easier to deal with computers, designed primarily for human interpretation
- e.g. patient table: name, surname, age, prescription

**Unstructured** Information (web documents …)
- information retrieval/querying based on “clever pattern matching” and human interpretation
- interrelationships of information based on “context” of document placement
- e.g. retail website
today, more than 4.7 billion pages
Dynamic Web

Web Services
UDDI, WSDL, SOAP

using the computer as a device for computation

Static
WWW
URI, HTML, HTTP
Semantic Web

Dynamic

Web Services
UDDI, WSDL, SOAP

Static

WWW
URI, HTML, HTTP

serious problems in information
- finding
- extracting
- representing
- interpreting
- and maintaining

Semantic Web
RDF, RDF(S), OWL
Intelligent Web Services

bringing the web to its full potential

Dynamic

Web Services
UDDI, WSDL, SOAP

Semantic Web Services
(OWL-S)

Static

WWW
URI, HTML, HTTP

Semantic Web
RDF, RDF(S), OWL
Web Services

- **A web service** is a **network accessible interface** that exposes the application functionality by a URI
  - Once it is deployed, other applications (and other Web services) can discover and invoke
  - It is implemented by using standard technologies
  - Ability to be synchronous or asynchronous

- **Web Services** **connect computers and devices** with each other using the Internet to exchange and combine data

- **Loosely-coupled** Integration
  - Clients do not need to know how it is implemented (programming language)
  - Agree on the object model
Web Services Advantages

- **Modular** : Service Components are useful in themselves, reusable, and it is possible to compose them into larger components.
- **Available** : Services are available to systems that wish to use them. Services must be exposed outside of the particular paradigm or system they are available in.
- **Described** : Services have a machine-readable description that can be used to identify the interface of the service, and its location and access information.
- **Implementation-independent** : The service interface must be available in a way that is independent of the ultimate implementation.
- **Published** : Service descriptions are made available in a repository where users can find the service and use the description to access the service.
Service-Oriented Architecture

An environment where

- Services are ubiquitous discovered and integrated
- Systems are assembled from loosely coupled collection of services (via published and accessed interfaces)
Service Oriented Standards & Technologies

- **XML**: Markup language designed to carry/transport data. Describes the format/structure of the request and response; data types.
- **SOAP**: Message format for communication between the parties involved (e.g., lightweight wire protocol and encoding format).
- **WSDL**: A mechanism for describing a web service in an independent way (e.g., interface, data types, binding information, address location).
- **Web Services**: Loosely coupled software components providing their functionalities via well-defined interfaces (WSDL).
- **UDDI**: Open framework for publishing and discovering Web Services descriptions into a searchable directory.
- **WS-BPEL**: A orchestration/process protocol for specifying coordinated web-service interactions among different stakeholders.
Managing Web Service Interactions

- Business Processes such as BPMN and WS-BPEL - provide a standard graphical notation for specifying business processes and an execution language for coordinated Web Service interactions.
Enabling Semantic Interoperation

• Next step is semantic interoperation
  • Understanding what the data means
  • Linking in insightful ways
  • Automated support for data integration
  • Develop Smart applications
    Sharing data ⇒ Sharing meaning

• Need a proper infrastructure for a real Web of Data
  • data is available on the Web - accessible via Web technologies and standards
  • data are interlinked over the Web
  • data are integrated over the Web

• This is where Semantic Web technologies come in
“The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in co-operation.”

[Berners-Lee et al, 2001]

Semantic Web is a web of data that machines can “understand” too.
Semantic Web Technologies

- A collection of standard technologies to realise the Web of Data – make the integration possible

  - Structured Web Documents (XML, XSD)
  - Describe Web Resources (RDF)
  - Web Ontology Languages (OWL)
  - Rule Languages (RIF, RuleML, SWRL)
  - Reasoning on the Semantic Web: reasoning tools (e.g. Jena)
  - Searching - Query language (SPARQL)
  - Storing the Semantic Web: Repositories (e.g. Sesame)
  - Semantic Web Services (OWL-S, WSMO)
  - Applications
Structured Web Documents

- XML is used to **encode** documents (e.g. knowledge bases) - provides the means to serialising structured documents
- It provides user definable and domain specific mark up (tags)
- XML Schema: determines the **syntax** (structure) of the document
- There is no commitment
  - On domain specific vocabulary to be used
example

<title>A Guide to XML</title>
<author>Norman Walsh</author>
<chapter>
  <title>What Do XML Documents Look Like?</title>
  <paragraph>If you are [...]</paragraph>
  <ol>
    <item>
      <paragraph>The document begins [...]</paragraph>
    </item>
    <item>
      <paragraph>Empty elements have [...]</paragraph>
      <paragraph>In a very [...]</paragraph>
    </item>
  </ol>
  <section> [...]</section>
</chapter>
<chapter> [...]</chapter>
</doc>
Describing Web Resources

- Resource Description Framework (RDF) is a framework for describing and interchanging metadata (data describing the web resources - anything on the Web).

- **Statements** are expressed as triples: a labelled connection between two resources or

  \[ \text{[subject=>predicate=>object]} \]

- RDF can integrate information from multiple resources
  - URIs form the basis of identifying and joining graphs

- RDF graphs can be serialised in multiple ways
  - (most commonly XML)
Example RDF Triples as Graphs

- unv:Person1234 = http://www.scss.tcd.ie/owen.conlan
- dbpedia:Dublin = http://dbpedia.org/resource/Dublin
Ontologies

• More elaborate constructs are needed to capture the meaning of the data
• An ontology is a specification of a conceptualization
• It describes the common concepts (vocabulary), hierarchies among concepts, relationships between concepts - represents an area of knowledge (see RDFS and OWL), and restrictions
• There should be a compromise between
  • rich semantics for meaningful applications
  • feasibility, implementability
Example

Logic and Inference

• Logic is the study of systems of reasoning - drawing conclusions
  • First-order logic: the logic of individual things
  • Second-order logic: the logic of types and relationships – can be complex and computational intensive

• Logic plays many different roles for the Semantic Web
  • Applying and evaluating rules
  • Inferring facts that haven’t been explicitly stated
  • Explaining why a particular conclusion has been reached (trace)
  • Detecting contradictory statements and claims
  • Key role in the statement of queries
Logic and Inference

• The rules take the form

IF <logical conditions are met>
THEN <perform specified actions>

(this kind of rules used by so-called expert systems)

• Evaluating the truth of the logical conditions involves logic. Rules are often chained together

• A processor can work backward from one condition to work out what had to happen to get there (trace)

• What is needed
  • A web language for expressing rules (e.g. SWRL, RuleML)
  • Be able to specify among rules relationships and constraints
  • Tools/Engines to handle the rules and reason about the data
Storing the Semantic Web: Repositories

• Semantic Web creates a wealth of data. Keeping them in one big text (e.g. Turtle or RDF/XML) is not the most efficient option (e.g. data are not indexed)
• Need for semantic repositories to support the efficient manipulation of Semantic Web data
• An RDF store holds place for storing the RDF datamodel as a sequence of: s (subject), p (predicate), o (object)
  • tools that combine the characteristics of database management systems (RDBS) – efficient storage, querying, management
  • inference engines – allow reason about the data
• Example of a Semantic Repository Engine
  • Sesame: most popular semantic repository that supports RDF(S) and all the major syntaxes and query languages related to it
• **SPARQL** is a query language of the Semantic Web - get information from RDF graphs
  • Is a *declarative* query language (similar to SQL)
  • Based on *pattern matching* against the RDF graph
  • *extract* information - e.g. triples, URIs, plain and typed literals
  • *construct* new RDF graphs from the queried graphs

• Different types of graph patterns are supported
  • Basic, Group, Optional, Alternative, Constraints

• Matching a *triple pattern* (subject, predicate, object) to a graph: bindings between variables and RDF Terms

Semantic Web Services

- **Semantic Web Services** are Web Services with a formal description (semantics) that can enable a better description, discovery, selection, invocation, composition, monitoring, and interoperability.
  - see Semantic Markup for Web Services (OWL-S)
    [http://www.w3.org/Submission/OWL-S/](http://www.w3.org/Submission/OWL-S/)

- **Processes** are created from the composition of Web Services and/or other components and allow to carry out more complex tasks such as e-commerce business activities, management activities, etc
Semantic Technologies for Unstructured Data

They are related to Natural-language processing, Information Retrieval and Extraction

- **Entity extraction** – (people, places, events, dates)
- **Cluster analysis** – group related information where relationships are unknown
- **Classification** – map to specific categories
- **Dependency identification** – rule generation
- **Coreference resolution** – two or more expressions in a text refer to the same entity
- **Automatic Summarization** – identify key concepts and key sentences
- **Example Tools:** GATE (General Architecture for Text Engineering)
Conclusions

- **Semantics** allow a common interpretation/meaning

- **Web Standards** facilitate interoperability

- **Data** on the Web is a major challenge
  - technologies are needed to use them, to interact with them, to integrate them

- Semantic Web and **best practices**, play a major role in publishing and managing Data on the Web
Thank you

Questions?