

# Semantic Web

Eleni Stroulia

## Levels of Interoperability (Manola 1995)

Frank Manola: Interoperability Issues in Large-Scale Distributed Object Systems.  
ACM Computing Surveys 27 (2): 268-270 (1995)  
<http://portal.acm.org/citation.cfm?id=210391>

Level of interoperability	Middleware support
<ul style="list-style-type: none"><li>• Physical (sharing data)</li></ul>	<ul style="list-style-type: none"><li>• Marshaling</li></ul>
<ul style="list-style-type: none"><li>• Object model<ul style="list-style-type: none"><li>– sharing objects</li></ul></li></ul>	<ul style="list-style-type: none"><li>• ORB architectures<ul style="list-style-type: none"><li>– OMG IDL, Microsoft OLE</li></ul></li></ul>
<ul style="list-style-type: none"><li>• Cooperation of sets of classes</li></ul>	<ul style="list-style-type: none"><li>• Application Frameworks</li></ul>
<ul style="list-style-type: none"><li>• Semantic<ul style="list-style-type: none"><li>– sharing the meaning of data and operations</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Metadata<ul style="list-style-type: none"><li>– data ontology, matching algorithms, functional and expected side-effects specification</li></ul></li></ul>

# Semantics: A Many-Splendored Thing

- “Semantics” means *meaning*.
- What has semantics? Where are they?  
What do they look like? How are they used?
- Kinds of Semantics:
  - Real-world Semantics
  - Axiomatic Semantics
  - Model-theoretic Semantics
  - Denotational, Procedural, Operational ... Semantics



2

## A Semantic Continuum



**Pump:** “a device for moving a gas or liquid from one place or container to another”



**Implicit**

**Informal**  
(explicit)

**Formal**  
(for humans)

**Formal**  
(for machines)

**Further to the right means:**

- Less ambiguity
- More likely to have correct functionality
- Better inter-operation
- Less hardwiring
- More robust to change
- More difficult!



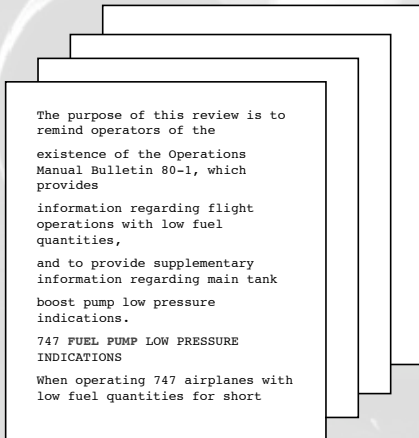
© Michael Uschold, The Boeing Company <sup>4</sup>

# Machine Processible Semantics



## Simple Task:

Find documents about mechanical devices.



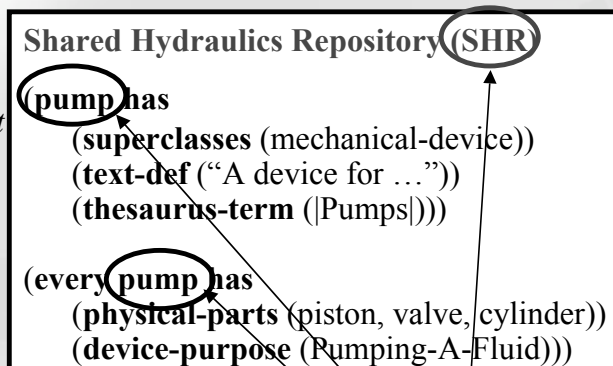
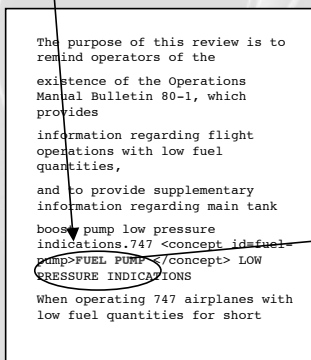
© Michael Uschold, The Boeing Company <sup>6</sup>

# Machine Processible Semantics



*Hey, I know this ontology, so now I know something about Fuel Pump.*

*What the heck is a Fuel Pump?*



## Semantic Markup

<concept id=fuel-pump>FUEL PUMP</concept>

(fuel-pump has (superclasses SHR pump))

© Michael Uschold, The Boeing Company



7

# Dynamic Discovery of Meaning

- **Machine can discover something about meaning.**
  - No idea what kind of pump, just a kind;
  - Rules out polar bears and cars;
  - Still plenty of scope for ambiguity;
  - Definition of fuel-pump can never be complete.
- **Inference for a Practical Task?**
  - For search application looking for content about mechanical devices.
  - May be no better than conventional search...



© Michael Uschold, The Boeing Company <sup>8</sup>

# Various Knowledge Technologies

## Artificial Intelligence

## Variety of Logics for Knowledge Representation

W3C HTML

RDF, RDF Schema

RDF, RDF Schema  
OWL, Taxonomies

## Library Science

Dublin Core  
Thesauri  
Taxonomies  
Topic Maps

Topic Maps

## Document Management

SGML, XML

SGML, XML

? XML Schema ?  
(datatypes)

Shopping Agents

Implicit

Informal  
(explicit)

Formal  
(for humans)

Formal  
(for machines)



© Michael Uschold, The Boeing Company <sup>9</sup>

# Ontologies

- An ontology is a theory about what types of things exist.
  - For the semantic web, an ontology is a document that formally defines the relations among terms.
    - The most typical kind of ontology for the Web has
      - **A taxonomy:** An inheritance forest among terms
      - **A set of inference rules:** Additional relation rules "If a city code is associated with a state code, and an address uses that city code, then that address has the associated state code."
      - **A set of equivalence relations (possibly):** Equivalent terms can exploit each-other's inheritance and other relations
- <http://www-ksl.stanford.edu/kst/what-is-an-ontology.html>

12-07

10

## Why is XML not enough?

- XML makes no commitment on:
  - Domain-specific vocabulary
    - The vocabulary of the tags chosen for a schema is up to the designer and (possibly) the agreement of the community
  - Ontological modeling primitives
    - Schemas offer limited syntax for expressing semantic constraints, such as
      - Equivalence
      - Inheritance

12-07

11

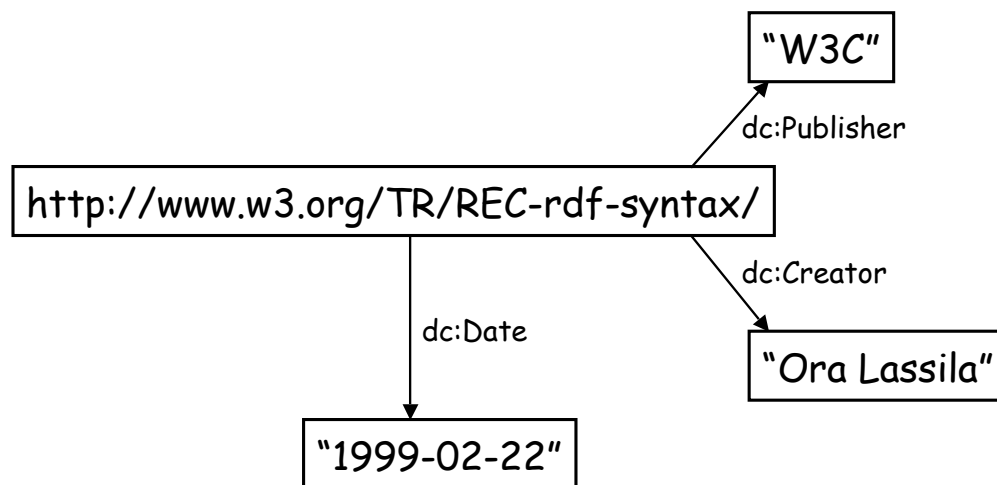
# RDF

- RDF encodes meaning in sets of triples, written using XML tags
- In RDF, a document makes assertions that
  - particular things (people, Web pages or whatever)
  - have properties ("is a sister of," "is the author of")
  - with certain values (another person, another Web page)
- Subject, verb and object are each identified by a Universal Resource Identifier (URI)

12-07

12

## RDF Model Example

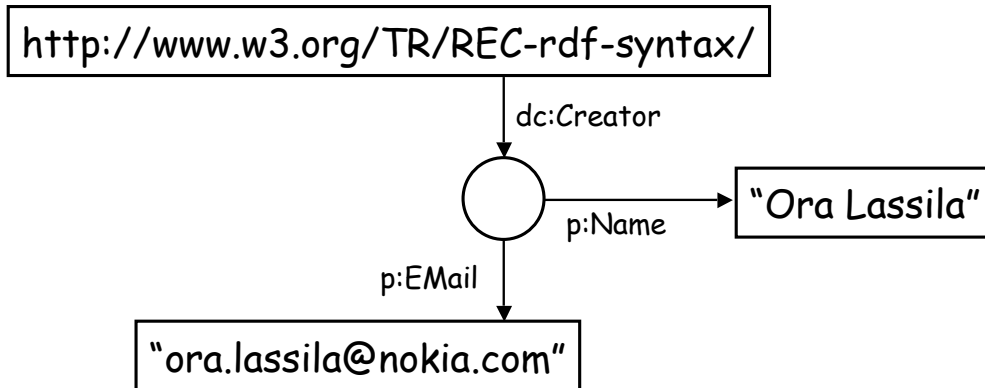


12-07

13

# A more complex example

- A graph node (corresponding to a resource) also can be the value of a property
- Corresponding triples
  - { “http://www.w3.org/TR/PR-rdf-syntax/”, dc:Creator, x }
  - { x, p:Name, “Ora Lassila” }
  - { x, p:EMail, “ora.lassila@nokia.com” }

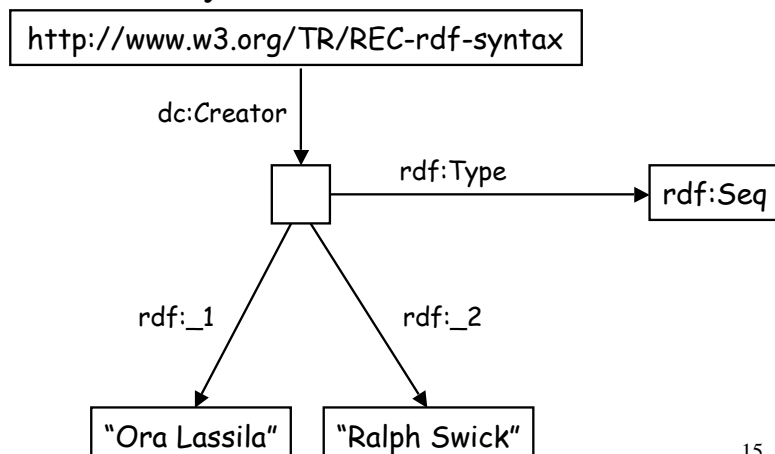


12-07

14

## Containers

- Containers are collections that allow grouping of resources (or literal values)
  - bag - unordered collection
  - seq - ordered collection (= “sequence”)
  - alt - represents alternatives
- It is possible to make statements about the container (as a whole) or about its members individually

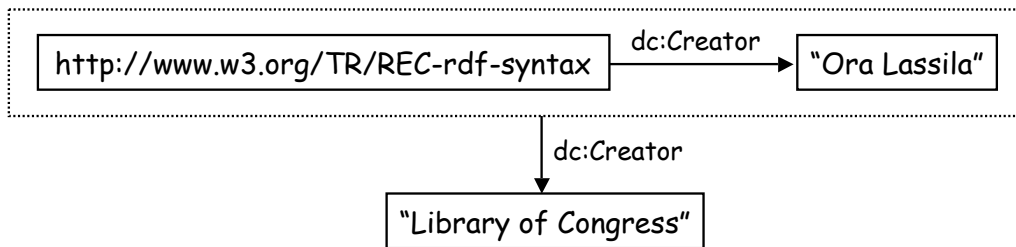


12-07

15

# Higher-order statements

- One can make RDF statements about other RDF statements



- The dotted box corresponds to the following statements
  - { x, rdf:predicate, “dc:creator” }
  - { x, rdf:subject, “http://www.w3.org/TR/REC-rdf-syntax” }
  - { x, rdf:object, “Ora Lassila” }
  - { x, rdf:type, “rdf:statement” }

12-07

16

## RDF Schema

- Defines small vocabulary for RDF:
    - class
    - subclassOf
    - type
    - property
    - subPropertyOf
    - domain
    - range
- ```

<rdf:Description ID="MotorVehicle">
  <rdf:type resource="http://www.w3.org/...#Class"/>
  <rdfs:subclassOf rdf:resource="http://www.w3.org/...#Resource"/>
</rdf:Description>
<rdf:Description ID="Truck">
  <rdf:type resource="http://www.w3.org/...#Class"/>
  <rdfs:subclassOf rdf:resource="#MotorVehicle"/>
</rdf:Description>
<rdf:Description ID="registeredTo">
  <rdf:type resource="http://www.w3.org/...#Property"/>
  <rdfs:domain rdf:resource="#MotorVehicle"/>
  <rdfs:range rdf:resource="#Person"/>
</rdf:Description>
<rdf:Description ID="ownedBy">
  <rdf:type resource="http://www.w3.org/...#Property"/>
  <rdfs:subPropertyOf rdf:resource="#registeredTo"/>
</rdf:Description>
  
```

12-07

17



# Beyond RDF: OIL+DAML=OWL

- OIL extends RDF Schema to a fully-fledged knowledge representation language.
  - logical expressions
  - data-typing
  - cardinality
  - quantifiers
  - See <http://www.ontoknowledge.org>
- DAML = US sister of OIL
- Merged as DAML+OIL in 2001
- Became OWL W3C standard in March '03 (look at <http://www.w3.org/TR/webont-req> )

12-07

18

## DAML+OIL (by example)

|                                                                                               |                                      |
|-----------------------------------------------------------------------------------------------|--------------------------------------|
| <b>class-def</b> animal                                                                       | % animals are a class                |
| <b>class-def</b> plant                                                                        | % plants are a class                 |
| <b>subclass-of</b> NOT animal                                                                 | % that is disjoint from animals      |
| <b>class-def</b> tree                                                                         |                                      |
| <b>subclass-of</b> plant                                                                      | % trees are a type of plants         |
| <b>class-def</b> branch                                                                       |                                      |
| <b>slot-constraint</b> is-part-of                                                             | % branches are parts of some tree    |
| <b>has-value</b> tree                                                                         |                                      |
| <b>max-cardinality</b> 1                                                                      |                                      |
| <b>class-def</b> defined carnivore                                                            | % carnivores are animals             |
| <b>subclass-of</b> animal                                                                     |                                      |
| <b>slot-constraint</b> eats                                                                   | % that eat any other animals         |
| <b>value-type</b> animal                                                                      |                                      |
| <b>class-def</b> defined herbivore                                                            | % herbivores are animals             |
| <b>subclass-of</b> animal, <b>NOT</b> carnivore                                               | % that are not carnivores, and       |
| <b>slot-constraint</b> eats                                                                   | % they eat plants or parts of plants |
| <b>value-type</b> plant <b>OR</b> ( <b>slot-constraint</b> is-part-of <b>has-value</b> plant) |                                      |

12-07

19

# **DAML-S Services Ontology**

- The class Service is at the top of the DAML-S ontology.
- The ontology of services provides two essential types of knowledge about a service: (a) Profile and (b) Model
- The ServiceProfile provides information about
  - Functionality
  - Constraints (security, locality, affordability)
- The ServiceModel enables an agent to
  - (1) perform a more in-depth analysis of whether the service meets its needs;
  - (2) compose multiple service descriptions to perform a specific task;
  - (3) coordinate the activities of different agents; and
  - (4) monitor the execution of the service.

12-07

20

## **Service Profile**

- Used to request or advertise services
- Service profiles consist of three types of information:
  - a description of the service and the service provider;
  - the functional behavior of the service;
  - functional attributes tailored for automated service selection.

12-07

21

### Description Properties

|                 |                                                                                                                                   |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------|
| serviceName     | The name of the service.                                                                                                          |
| intendedPurpose | A high-level description of what constitutes (typical) successful execution of a service.                                         |
| textDescription | A brief, human readable description of the service, summarizing what the service offers or what capabilities are being requested. |
| role            | An abstract link to <i>Actors</i> involved in the service execution.                                                              |
| requestedBy     | A sub-property of role referring to the service requester.                                                                        |
| providedBy      | A sub-property of role referring to the service provider.                                                                         |

### Functional Attributes

|                   |                                                                                                                                                          |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| geographicRadius  | Geographic scope of the service, either at the global scale (e.g. e-commerce) or at a regional scale (e.g. pizza delivery).                              |
| degreeOfQuality   | Quality qualifications, such as providing the <b>cheapest</b> or <b>fastest</b> possible service.                                                        |
| serviceParameter  | An expandable list of properties that characterize the execution of a service, such as <code>averageResponseTime</code> or <code>invocationCost</code> . |
| communicationThru | High-level summary of how a service may communicate, e.g. what communication language is used (e.g., KQML, SOAP).                                        |
| serviceType       | Broad classification of the service that might be described by an ontology of service types, such as B2B, B2C etc.                                       |
| serviceCategory   | Categories defined within some service category ontology. Such categories may include <i>Products</i> , <i>Information Services</i> etc.                 |
| qualityGuarantees | Guarantees that the service promises to deliver, e.g. guaranteeing to provide a response within 3 minutes, etc.                                          |
| qualityRating     | Industry-based ratings, such as the “Dun and Bradstreet Rating” for businesses.                                                                          |

22

## Process model

- The two chief components of the process model are
  - the Process Ontology
    - Service inputs, outputs, preconditions, effects, sub-processes
  - the Process Control Ontology
    - state, including initial activation, execution, and completion

# Resources

- RDF and RDF Schema tutorials
  - <http://lsdis.cs.uga.edu/SemWebCourse/RDF.ppt>
  - <http://www710.univ-lyon1.fr/~champin/rdf-tutorial>
  - <http://lsdis.cs.uga.edu/SemWebCourse/rdfdarpa.ppt>
- Database and Information Systems Research for Semantic Web and Enterprises  
Invitational Workshop Sponsored by NSF CISE-IIS-IDM, Co-Sponsored by EU  
Thematic Network OntoWeb
  - <http://lsdis.cs.uga.edu/semNSF/>
- Semantic web W3C activity
  - <http://www.w3.org/2001/sw/Activity>
- DAML
  - <http://www.daml.org/>
- The DAML Services Coalition: DAML-S: Web Service Description for the  
Semantic Web
  - <http://www.daml.org/services/>
- <http://www.w3.org/2005/Talks/0623-sb-IEEEStorConf/>