Adding formal semantics to the Web

building on top of RDF Schema

Jeen Broekstra
aidministrator nederland b.v.
On-To-Knowledge project

Context

On-To-Knowledge

- IST project about content-driven knowledge management through evolving ontologies
- http://www.ontoknowledge.org/

OIL = Ontology Inference Layer

- http://www.ontoknowledge.org/oil

Goal of this presentation

- Show how we can apply ontologies to semi structured data
- Show how defining OIL in an RDF syntax adds formal semantics to the Web

Contents

- Semantic annotation: why, and how?
  - how: XML?
  - how: RDF(S)?
  - how: The W3C vision
- OIL
- extending RDF Schema
- Conclusion and summary

Semantic annotation: why

- Semantic annotations make the meaning **machine-accessible:**
  - Intelligent information brokering
  - Meaning-based searches
  - Prerequisite for many agent applications

Is XML sufficient for semantic annotation?

- XML: user definable and domain specific markup
- XML document is a labeled tree
- constraints on structure via DTD or XML Schema

```
<animal>
      <name>Tux</name>
      <species>penguin</species>
      <eats>fish</eats>
      ...
</animal>
```

aidministrator nederland b.v. http://www.aidministrator.nl/

Shortcomings of XML (for our purposes)

XML makes no commitment on:

❶ Domain specific ontological vocabulary

❷ Ontological modeling primitives

⇨ requires pre-arranged agreement on ❶ & ❷

Only feasible for closed collaboration

– agents in a small & stable community
– pages on a small & stable intranet

not for sharable Web-resources

Is RDF(S) sufficient for semantic annotation?

- RDF provides metadata about Web resources
- Object -> Attribute-> Value triples

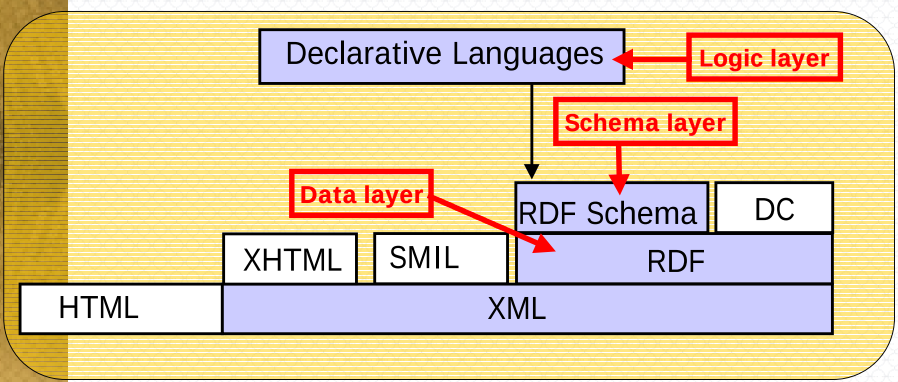
- RDF Schema
  - Defines vocabulary for RDF
  - Organizes this vocabulary in a typed hierarchy
    - `Class, subClassOf, type`
    - `Property, subPropertyOf`
    - `domain, range`

Conclusions about RDF(S)

- Next step up from plain XML:
  - (small) ontological commitment to modeling primitives
  - possible to define vocabulary
- However:
  - no precisely described meaning
  - no inference model

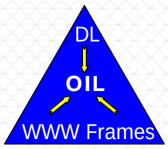
Semantic annotation: how

W3C's vision: The Semantic Web



OIL

- Based on standard frame languages (OKBC)
  - restricts & extends
- formalized by DL style logical constructs
- Still has frame “look and feel”
- Can still function as a basic frame language
- OIL language restricted:
  - to allow for reasoning support

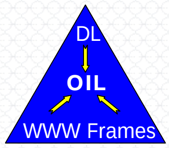


OIL (explained by example)

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

OIL has a formal semantics

- Defined by mapping to expressive DL
  - slide-constraint eats has-value meat, fish
    =
    $\exists$ eats:meat $\cap$ $\exists$ eats:fish
- Mapping is used to provide reasoning support from a DL system (e.g., FaCT)



aidministrator nederland b.v. http://www.aidministrator.nl/

Extending RDF Schema

Goal

- make RDFS useable as ontology language
  - give RDF(S) precise semantics
  - extend RDF(S) with additional modeling primitives
- to facilitate semantically grounded metadata

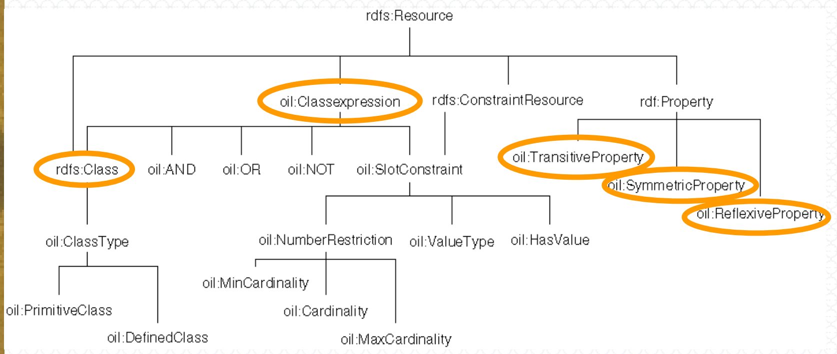
Procedure

- formulate ontology language as RDF Schema document
  - using existing primitives as much as possible
  - placing additional primitives in the hierarchy of RDFS primitives

aidministrator nederland b.v. http://www.aidministrator.nl/

OIL as extension to RDFS (1)

- part of the is-a hierarchy of RDFS extension
- ontology language is defined in RDFS



OIL as extension to RDFS (3)

```
<rdfs:Class rdf:ID=”herbivore”>
   <rdf:type
        rdf:resource=”http://www.ontoknowledge.org/#DefinedClass”/>
   <rdfs:subClassOf rdf:resource=”#animal”/>
   <rdfs:subClassOf>



   </rdfs:subClassOf>
</rdfs:Class>
```

Using the extension: three levels

❶ OIL modeling primitives

slot-constraint, subclass-of, value-type,…

– RDF-S document which extends RDF-S

❷ a specific OIL ontology

animal, plant, herbivore, leaf

– RDF-S document (using ❶)

❸instances of this ontology

"Mel the giraffe", "Tux the penguin"

– RDF expressions (uses ❶&❷)

– explicit metadata

What did we gain?

- Any RDF agent can
  process OIL instances
- Any RDF-S agent can
  process OIL ontologies
- Any OIL-aware agent can
  exploit semantics & reasoning

(and materialize the OIL derivations
for use by OIL-ignorant RDF agents)

Summary

- to enable intelligent information handling, machine-understandable semantics are needed
- advantages of our approach
  - reuse of modeling primitives
  - conform W3C view of the world
  - added benefits (from OIL):
    - reasoning support
    - formal semantics