How is it going, Semantic Web?
What’s new in the Semantic Web

- **RDF**: a new data model for the Web
- **OWL**: standard web ontology language to define the meaning of vocabulary
- **SPARQL**: query language

SW became widely accepted

Two key steps:

1. Building reusable ontologies
2. Using ontological vocabulary to annotate data
Linking open data

[Photo source: talis.com]
What is Schema.org?

This site provides a collection of schemas, i.e., tmpl tags, that webmasters can use to markup their pages so they are recognized by major search providers. Search engines including Bing, Google, Yahoo! and Yandex rely on this markup to improve the display of search results, making it easier for people to find the right web pages.

Many sites are generated from structured data, which is often stored in databases. When this data is formatted into HTML, it becomes very difficult to recover the original structured data. Many applications, especially search engines, can benefit greatly from direct access to this structured data. On-page markup enables search engines to understand the information on web pages and provide richer search results in order to make it easier for users to find relevant information on the web. Markup can also make new tasks and applications that make use of all the structure.

A shared markup vocabulary makes it easier for webmasters to decide on a markup schema and get the maximum benefit for their efforts, too. In the spirit of OpenData.org, search engines have come together to provide a shared collection of schemas that webmasters can use.

We invite you to get started!


Official OWL ontology: http://schema.org/docs/schemaorg.owl

HTML microdata: http://www.w3.org/TR/microdata/

Data as rich objects

The Open Graph protocol

Introduction

The Open Graph protocol enables any web page to become a rich object in a social graph. For instance, this is used on Facebook to allow any web page to have the same functionality as any other object on Facebook.

While many different technologies and schemas exist and could be combined together, there isn’t a single technology which provides enough information to richly represent any web page within the social graph. The Open Graph protocol builds on these existing technologies and gives developers one thing to implement. Developer simplicity is a key goal of the Open Graph protocol which has informed many of the technical design decisions.
Creating semantic data with mobiles

Your mobile knows “everything” about you:

Where you are
Who your social network contacts are

Semantic Web Enabled Software Engineering: Why and How
Model Driven Software Development

- Hard to find details
- Loss of oversight

Ontology enabled software analysis

EvoOnt: A Software Evolution Ontology
  • Software Analysis

Many tasks in MSR (Mining Software Repository) Workshop 2004 - 2007 can be done by 1~2 SPARQL query
[Tappolet et al. 2010]
Ontology enabled MDSD

Ontology driven software development
How to bridge SE models and ontologies

Approach 1: Integrated models
• Static SE models
• Ontologies

Integrated model: example 1

Two components
• UML class diagrams
• Ontologies
Integrated models: example 2

Two components
- Physical Device DSL (PDDSL) models
- Ontologies

How to bridge SE models and ontologies

Approach 2: problem reduction to task ontologies
- To reduce an SE problem
- Into ontology reasoning problem
Business process refinement

To reduce business process refinement validation

Into ontology consistency/coherency checking

"Traceability": Which activities cause the problem?

Justification: Which axioms cause the problem?

• Ren et al. 2009b; Groener and Staab 2009

26/04/2012

Business process refinement: example

1. In pre-refinement process:
   Component_A subclassOf (to only (Component_A or Component_B);
   Component_B subclassOf from only (Component_B or Component_A);

2. In post-refinement process:
   a31 subclassOf to some End
   b12 subclassOf (to some b23) and (to some a22), etc.

3. Mapping:
   a31 subclassOf Component_A;
   b23 subclassOf Component_B; etc.

4. Uniqueness:
   Disjoint(Start, End, Component_A, Component_B), etc.
Traceability: exploiting the integrated models and task ontologies

- Help understand the associations and dependencies
- By exploiting reasoning and query answering

Dynamics and complete of knowledge

Two types of knowledge
- Some we have complete understanding (DB)
- Some we only know partially (SW)
Semantic Web supports BOTH

NBox (Negation As Failure Box) enabled ontologies [JTS]

- TBox: schema axioms
- ABox: data axioms
- **NBox**: a set of closed vocabulary
  - NAF of NBox concepts can be simulated by full negation
  - Such as Spicy
  - Require **incremental reasoning**

<table>
<thead>
<tr>
<th>Food</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curry Chicken</td>
<td>Minor Spicy</td>
</tr>
<tr>
<td>Salmon Fillet</td>
<td></td>
</tr>
<tr>
<td>Spicy Grilled Shrimp</td>
<td>Spicy</td>
</tr>
<tr>
<td>Pepper Salad</td>
<td>Vege</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Vegetarian</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jeff</td>
<td>No</td>
</tr>
<tr>
<td>Yuting</td>
<td>Yes</td>
</tr>
<tr>
<td>Jek</td>
<td></td>
</tr>
<tr>
<td>Yuan</td>
<td>No</td>
</tr>
</tbody>
</table>

Architecture

[Ren et al. 2009]
Guidance services

Todo list services for developers:
• To recommend the next available actions/options
• To validate the current action/option

Guidance services: example
28

Guidance services: example

Input Port

Data source
port type

SQL data port type

Query 1 data source
Classifier data
Merge data 1

SQL connection 1

ONLY

Some connection

ONLY

class connection

Connection

Output Port

Which input port can NOT be used here?

29

Guidance services: example

Input Port

Data source
port type

SQL data port type

Restricted connection

Occupied

Merge data 1

SQL connection 1

ONLY

Some connection

ONLY

class connection

Connection

Output Port

Which input port can NOT be used here?
**Guidance services: example**

- **Data source port type**
  - Restricted connection
  - Occupied
  - Valid input port
- **SQL data port type**
  - Query 1 SQL data
- **Which input port can be used here?**

**OWL: Standard Ontology Language**

- **OWL 2 Full**
  - Undecidable
- **OWL 2 DL**
  - 2NExpTime-Complete
- **OWL 1 DL**
  - NExpTime-Complete
- **OWL 2 QL**
  - PTime-Complete
- **OWL 2 RL**
  - In AC^0
- **OWL 2 EL**
  - EL++
Approximate reasoning plays a key role

Inputs
- Ontology
- Queries

Outputs
- soundness preserving
- completeness preserving

I can’t find any algorithms, but neither can all these famous people.


Faithful Syntactic Approximate Reasoning
[AAAI2010]

Approximation
- Directly represent non-OWL2-EL concepts with fresh named concepts
  - E.g., ∀r.C subClassOf D \( \rightarrow \) A\(_{∀r.C}\) subClassOf D
- Maintain semantic relations for these named concepts
  - complementary relations
  - cardinality relations

Reasoning
- Using additional tractable completion rules to recover the semantics
Example: How Does it Work

Additional completion Rules (on top of the EL ones), e.g.

- Handling complement
  - E.g. B subClassOf C ⇒ ¬C subClassOf ¬B
- Handling cardinality
  - E.g. A subClassOf >= 3 r. B ⇒ A subClassOf >= 2 r. B
- Soundness preserving and tractable

Evaluations for the Oxford Benchmarks
(REL – EL reasoner in TrOWL)
TrOWL: a tractable semantic reasoning infrastructure

Stream reasoning [CIKM2011]  Local closed world reasoning [JTS]

Benefits for Software Engineering

- Integrated modelling and querying
- Guidance services for developers:
  - To recommend the next available actions
  - To validate the current action
- Ontologies available at deployment time
Vision: Programs that can understand and improve themselves

API4KB
Possible impacts on the Semantic Web

- More data
  - Linked software data
- Easier to debug, reuse and extend semantic applications
- Software becomes semantics software
- Software/service as a first class citizen like data?

Where to find more information

International SWESE workshop series
http://www.abdn.ac.uk/~csc280/event/workshop/swese2011/

Two books coming:
Semantic Web Enabled Software Engineering

WebIST2012

Thank you

... questions?