Introduction to the Semantic Web

Sebastian Wandelt
Overview

• Sebastian Wandelt
• B.Sc. in (technical) computer science from HWR Berlin
• M. Sc. in computer science from Dresden University of Technology
• Ph.D. in computer science from Hamburg University of Technology
• => I am a computer scientist by heart, with all its advantages and drawbacks
Computer scientist stereotypes

- "... it's a bunch of people addicted to sitting in their mom's basement playing World of Warcraft and drinking Dr Peppers"
  Cameron Murray, CS student, Cleveland

- Talk fast in presentations
- Very limited theory of mind
- Reluctant/resistant to feedback
Computer scientist stereotypes

- "... it's a bunch of people addicted to sitting in their mom's basement playing World of Warcraft and drinking Dr Peppers"
  
  Cameron Murray, CS student, Cleveland

- Talk fast in presentations ✔
- Very limited theory of mind ❓
- Reluctant/resistant to feedback
How about you?

- Name
- Degree course
- Bachelor in ...?
- Background with web technology?
告訴我，我會忘記。
教我，我會記住。
讓我參予，我會理解。
Tell me, I will forget.

教我，我會記住。
讓我參予，我會理解。
Tell me, I will forget.

Show me and I may remember.

讓我參予，我會理解。
Tell me, I will forget.

Show me and I may remember.

Involve me and I will understand.
Let's start: How to use Facebook?

- Go to URL facebook.com
- Complete the form with username, password, ..
- Wait for an email to ...
Let's start: How to use facebook?

- Go to facebook.com
- Complete the form with username, password, ...
- Wait for an email to...
Let's start: How to use Facebook?

- Go to facebook.com
- Complete the form with username, password, etc.
- Wait for an email...

Scared?

You should be!
Disclaimer

- This is not a lecture on “Hot to use X,Y,Z”, especially not Facebook, Twitter, IFTTT, ...

- So what is it?
  - A course on the foundations of the Semantic Web as it is/was a vision by academia.
  - A course which requires interest and knowledge in theoretical computer science and will, at some points, make (heavy) use of it.
  - A course which intends to give you an idea on how the Semantic Web might change our future, once it is there.
Overview

- Organization
- A short introduction to the Semantic Web
- Detailed outline of the course
- Repetition of Propositional logic and First order logic
Organization

- Lecture
  - Every Monday, 9:15 - 10:45 AM

- Lab session
  - Every Monday, 11:15 - 12:45 AM
  - Usually paper & pencil exercises
  - Other suggestions?

- Reduce break, start lecture more late, etc.?

- Language: English or German?
Organization

• Examination
  – At the end of the term
  – Oral or written? You tell me ...

• Diploma students?
  – Need to solve some extra task in order to obtain 8 SP, instead of 5 SP
  – What kind of task?
A short introduction to the Semantic Web

- World Wide Web - Limits
- Semantic Web - Vision
The World Wide Web

- A long time ago ...
- The WWW was born 1990 at CERN
- Basically
  - HTML
  - Usenet
  - Forums
What Beautiful HTML Code Looks Like

**DOCTYPE Properly Declared**

It looks like a lot of gibberish, but DOCTYPES are important. They not only allow your code to validate, but they tell browsers things about how to render your page. Simple `<em>` tags don't cut it.

**Body IDEd**

Putting an ID on your body allows you to create CSS properties that are unique to that page. For instance, you may want your `<h2>` tags to look different on the homepage. In your CSS, you can write: `
home h2 {}` to accomplish this and not affect `<h2>` tags elsewhere.

**Main DIV for all Page Content**

Putting all the content of your page into one main `wrap` DIV gives you lots of control right off the bat. There is where you can set the width of your page for a fixed width site or maximums and minimums for a fluid width site.

**Common Content INCLUDED**

A lot of web content is common from page to page. Think menu bars, sidebars, footers, “boxes”, etc. This kind of content should be dynamically loaded. Either from a database or with simple PHP include statements.

**Proper Ending Tags**

You started strong, now end strong. Don’t be lazy and exclude closing tags for any element, even if the page renders OK without them.

**Content, Content, Content**

This is where your content belongs, so go nuts. Remember to keep your paragraphs distinct and in `<p>` tags. Use lists where appropriate. Use codes like &copy; for ® symbols. Don’t go overboard with `<br />` tags, that’s sloppy formatting.

**Tidy Head Section**

Title is set. Character set declared. Stylesheets linked (including a print stylesheet). Scripts linked and NOT included in full. External files have their own related folders (e.g. "CSS" & "Scripts").

**Semantically Clean Menu**

```html
<body id="home">
    <div id="page-wrap">
        <php include_once("menu.html") ?>
        <div id="main-content">
            <div id="search-area">
                <input value="Search..." onfocus="this.value=''" />
                <div style="clear:both;">
            </div>
        </div>
        <php include_once("left-sidebar.html") ?>
        <div id="right-content">
            <div class="new-box">
                <php include_once("new.html") ?>
            </div>
            <div class="events-box">
                <php include_once("events.html") ?>
            </div>
        </div>
    </main-content>
</body>
```

**Important Content First**

It is best if your most important content, like news and events, can be listed first in the HTML. If your sidebar is just navigation or less important content, it is best if it comes last in the HTML.

**Code is Tabbed into Sections**

If each section of code is tabbed in once, the structure of the code is much more understandable. Code that is all left-justified is horrific to read and understand.

**Hierarchy of Header Tags**

Use header tags as they were designed, to create titles for sections and signify their position in the content hierarchy.

**No Styling!**

Your HTML should be focused on structure and content, not styling! Keep all of your styling in your CSS, there should be no deprecated `<font>` tags in site.
Reality?

Content vs. Design!
WWW -> Web 2.0

- Web 2.0 (not to be confused with SW) is
  - Blogs
  - Wikis
  - Social XYZ

  => Cooperation, Power of the Masses
Limits of WWW / Web 2.0

- What is important?
- Information vs. Advertisements?
- Reliability?
- What is related?
- What is redundant?
Limits of WWW / Web 2.0

Main challenge:

Automatic extraction and processing of information.

=> We need formal semantics!

(HTML? No!)
What is “semantics”?

- Semantics is a subarea of linguistics concerned with the sense and meaning of a language.
- Usually the definition of a semantic is based on a given syntax.
- To sum up:
  - Syntax = structure
  - Semantics = meaning
- For instance:
  - $3+\dd=\&$ (syntactically wrong)
  - $3+8=23$ (syntactically correct, but semantically wrong)
  - $3+8=11$ (syntactically and semantically correct)
The vision of a Semantic Web

- „The Semantic Web is a web of data, in some ways like a global database.“
- „...a universal Web of semantic assertions“
- „...emphasises the intelligent manipulation of the knowledge that is made available via the web.“
- „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 cooperation"
The original road map

Stufe 1: Textdokumente und Datenbankeinträge

Stufe 2: XML-Dokumente mit einheitlichem Vokabular

Stufe 3: RDF-Taxonomien und Dokumente mit unterschiedlichem Vokabular

Stufe 4: OWL-Ontologien und automatisches Schlussfolgern
The result (up to now)
Gartner's hype cycle

Expectations:
- Database Platform as a Service (dbPaaS)
- Data Stewardship Applications
- Data Services
- Complex-Event Processing
- NoSQL Database Management Systems
- In-Memory Data Grids
- Enterprise Taxonomy and Ontology Management
- Information Capabilities Framework
- Data Integration Tools and Data Quality Tools Convergence
- Data Integration Platform as a Service
- Enterprise Information Management Programs
- "Big Data" and Extreme Information Processing and Management
- Open-Source Data Quality Tools

Peak of Inflated Expectations:

Trough of Disillusionment:
- Enterprise Metadata Repositories
- Data Quality Software as a Service
- Cross-Platform Structured Data Archiving
- Entity Resolution and Analysis
- In-Memory Database Management Systems
- Open-Source Data Integration Tools
- Data Stewardship
- MapReduce
- Master Data Management

Slope of Enlightenment:
- Data Integration Tool Suites
- Database Software as a Service
- Data Federation/Virtualization
- Open-Source Database Management Systems
- Data Profiling
- Content Integration and Migration

Plateau of Productivity:
- Database Appliances
- Data Quality Tools
- Column-Store Database Management Systems

As of July 2011

Years to mainstream adoption:
- ○ less than 2 years
- ⬜ 2 to 5 years
- ⬤ 5 to 10 years
- ▲ more than 10 years
- ● obsolete before plateau

Source: Gartner (July 2011)
What is the Semantic Web not?

- The Semantic Web is not AI!
- RDF and OWL are simple ontology languages

„One goal in SW is to keep things relatively simple and not necessarily seek absolute completeness“

What is the Semantic Web not?

- The Semantic Web does not cover
  - Complex inferences
  - Complex temporal/spatial reasoning
  - OCR, gesture recognition
  - Reasoning about incomplete knowledge

“Just as Prolog is not AI but merely a useful tool for it, SW might be just a good tool for AI“
Where is the cheapest place to refill the gas?
Where is the cheapest place to refill my gas?
“Where is the cheapest place to refill the gas?”
“Where is the cheapest place to refill the gas?”

=> Direct Answer:
   Jet, Märkische Allee 246, 3.90 €/l

=> Indirect consequences:
   - Check opening hours of the gas station
   - Forward destination to the navigation system
   - Automatically reduce power in order to save gas until arriving at the gas station
   - ...

SW Vision
Intelligent agents

WWW today

User

Presentation service, e.g. Firefox

Search service, e.g. Google

Documents

SW future

User

Personal assistant

Infrastructure services

Documents
Detailed Outline

- Basically 16 topics for 16 time slots
- Detailed topics are still open for discussion ...
1. Introduction / Repetition

• Repeat the basics of propositional and first order logic
  => Have a common understanding of notions and common knowledge of theoretical results
  => An important basis for the remaining lectures!
2. URIs, XML, Namespaces

- A rather boring topic: the general ideas are pretty simple and intuitive, but the technical details can be ridiculous.

- Kick it out?
2. URIs, XML, Namespaces

- Syntax and properties of URIs
- Foundations of XML
  - Trees, tags, attributes, links
  - Well-formedness and validity of documents
  - Applications
- What are namespaces? How are they defined and accessed.
**URIs**

**QUALIFIED NAME**

- **PREFIX**
  - **LOCAL NAME**

**“BK” NAMESPACE DECLARATION**

- **PREFIX**
  - **NAMESPACE NAME (URI)**

```xml
<BK:BOOKSTORE XMLNS:BK="http://www.example.org/bookstore"/>
```

**DEFAULT NAMESPACE DECLARATION**

- **NO PREFIX**
  - **LOCAL NAME**

- **NAMESPACE NAME (URI)**

```xml
<BOOKSTORE XMLNS="http://www.example.org/bookstore"/>
```
<?xml version="1.0" encoding="UTF-8"?>
<customers>
  <customer id="1">
    <name>Woodworks</name>
    <city>Baltimore</city>
    <projects>
      <project id="1">
        <name>Medusa</name>
      </project>
    </projects>
  </customer>
  <customer id="4">
    <name>Hardware Shop</name>
    <city>Washington</city>
    <projects>
      <project id="2">
        <name>Pegasus</name>
      </project>
      <project id="8">
        <name>Typhon</name>
      </project>
    </projects>
  </customer>
</customers>
<!-- ### SNIP ### -->
3. **RDF, RDF Schema, SparQL**

- **RDF**: Defines a simple data model for the description of machine usable semantics of data
  - Only very basic semantics
  - Triples represent a graph
- **SparQL**: graph query language
3. RDF, RDF Schema, SparQL

- RDF Beispiel:
  - hatKind(homer, lisa)
  - hatKind(homer, bart)
  - hatGeschlecht(lisa, w)

- SparQL Beispiel:
  - SELECT ?person
    homer hatKind ?person.
    ?person hatGeschlecht w.

=> Intuition of these languages is pretty simple, but the technical details can be tricky!
Example for graph representation
4. **OWL: syntax and intuition**

- OWL ontologies consist of classes, properties, individuals
- Kinds of axioms and their intuition
- We will mainly look at DLs!
4. **OWL: syntax and intuition**

- maria : Parent
- christine : Woman
- (christine, zoe) : hasChild
- (frank, christine) : hasChild
- (christine, toby) : hasChild
- frank : Man ⊑ Mother
- maria : ∃hasChild.(∃hasChild.Man)
5. **OWL: formal semantics**

- What does an expression/axiom really mean?
- How to (automatically) deduce additional knowledge?
6. **OWL: Tableau-based decision procedures**

- Early reasoning systems were incomplete
- Tableau proofs are one way for sound and complete reasoning about very expressive description logics
- To grasp the intuition needs time!
6. **OWL: Tableau-based decision procedures**

- A Tableau is (usually) a tree-like data structure, which gradually explores an ontology in each level of the tree, looking for contradictions.
7. The inference system RACER
8. Extensions to description logics

- Fuzziness/Probability
- Non-standard inference problems and their reduction
- Concrete domains
- Modal and temporal extensions
9. Open (christmas) slot

- A topic up to you (if you want)!
- You propose/agree on a topic until one week before the lecture, and I will prepare a presentation
- Only constraint: Related to Semantic Web!
- What do you think?
10. OWL: Solving extended decision problems

- Many, many more decision problems
- Reduction
- Optimization
11. **OWL: Scalability and lightweight subsets of OWL plus rule languages**

- Basic decision problems have N2EXPTIME worst-case time complexity
- How complex is that?
- LOG, P, NP, EXP, NEXPTIME, N2EXPTIME, R
12. OWL: Common misbeliefs and problems in modeling ontologies

- Usual pitfalls when modeling an ontology
  - Open world assumption
  - Practical undecidability
  - Used reasoning system is still a design decision
  - Common misunderstandings about inclusion and equivalences
13. Ontology engineering and design patterns

- Best practices and common patterns for recurring problems
- Ontology learning
- Ontology Reuse
- Modularization
14. Programming the Semantic Web

- Accessing Triple Stores
- Using the OWLAPI
- Common data silos and how to use them
  - Accessing facebook/twitter data
- Code in Java (?)
15. Real world application examples (of working ontologies and SW systems)

- From toy ontologies to Linked-data silos
- Mostly ontologies from Bioinformatics
  - GO, Uniprot,....
- RDFa published data by many parties

- But is this really interesting?
16. Human computation based on SW technologies

- reCAPTCHA
- ESP Game
- Citizen Science projects
  - Galaxy Zoo, eBird
- Crowdsourcing marketplaces
  - Amazon mechanical turk
Discussion of the outline

- Is there anything you
  - feel is missing,
  - are totally excited about, or
  - would like to know more about,
which is related to the Semantic Web?

=> The outline is a proposal only and open for discussion!
Lecture roadmap

- 1. Intro (22.10.2012)
- 2. URI/XML (29.10.2012)
- **3. RDF/SparQL** (05.11.2012)
- **4. - 6. OWL** (12./19./26.11.2012)
- 7. Racer (03.12.2012)
- 8. Extensions to Description Logics (10.12.2013)

- **10. - 12. OWL in practice** (07./14./21.01.2013)
- **14. Programming the Semantic Web** (04.02.2013)
Serious Hint

If you intend to skip a lecture, fine.
If you intend to skip Number 4-6, I would highly recommend you to reconsider your plan!

=> OWL/DL syntax, semantics, and decision procedures are the foundation for at least half of the course (and will be the foundation for at least half of the examination)!
Propositional Logic

- Syntax
- Well-formed formulae
- Interpretation
- Satisfiability, Validity
- Minimum set of constructors
- Implication, Equivalence
- Decision algorithms
Propositional Logic Test

- If Mary loves Pat, then Mary loves Quincy. If it is Monday, Mary loves Pat or Quincy.
- Prove(!) that, if it is Monday, then Mary loves Quincy.
Heads you win. Tails I lose. Show that you always win!