CSC 481: The Semantic Web

Based on Chapter 1 of “Semantic Web for the Working Ontologist - Modeling in RDF, RDFS and OWL” by Allenmag and Hendler

Rodrigo Canaan
Assistant Professor
Computer Science Department
Cal Poly, San Luis Obispo
rcanaan@calpoly.edu

Recap of course so far

Recap of course so far

- **Knowlege** usually expressed via propositions

Recap of course so far

- **Knowlege** usually expressed via propositions
- **Representations** map objects from one domain to another

Recap of course so far

- **Knowlege** usually expressed via propositions
- **Representations** map objects from one domain to another
- **Reasoning** can be used to produce new representations of knowledge

Recap of course so far

- **Knowlege** usually expressed via propositions
- **Representations** map objects from one domain to another
- **Reasoning** can be used to produce new representations of knowledge
    - Can be thought of as manipulation of symbols in a language

Recap of course so far

Recap of course so far

First-order logic was the language of choice for the course to represent and reason with logic

Recap of course so far

First-order logic was the language of choice for the course to represent and reason with logic

- **Syntax** tells us which sentences (combinations of symbols) are well-formed

Recap of course so far

First-order logic was the language of choice for the course to represent and reason with logic

- **Syntax** tells us which sentences (combinations of symbols) are well-formed
- **Semantics** allow us to define what a sentence means with respect to a model (or interpretation)

Recap of course so far

First-order logic was the language of choice for the course to represent and reason with logic

- **Syntax** tells us which sentences (combinations of symbols) are well-formed
- **Semantics** allow us to define what a sentence means with respect to a model (or interpretation)
    - In particular, the notion of entailment

Recap of course so far

First-order logic was the language of choice for the course to represent and reason with logic

- **Syntax** tells us which sentences (combinations of symbols) are well-formed
- **Semantics** allow us to define what a sentence means with respect to a model (or interpretation)
    - In particular, the notion of entailment
- **Inference Rules** such as resolution let us design algorithms to produce new symbols that represent sentences entailed by a KB

Recap of course so far

First-order logic was the language of choice for the course to represent and reason with logic

- **Syntax** tells us which sentences (combinations of symbols) are well-formed
- **Semantics** allow us to define what a sentence means with respect to a model (or interpretation)
    - In particular, the notion of entailment
- **Inference Rules** such as resolution let us design algorithms to produce new symbols that represent sentences entailed by a KB
    - Horn Clauses are a subset of FOL with efficient inference algorithms

Should you represent everything with FOL?

Should you represent everything with FOL?

- *Ontological commitment*: world consists of facts about objects and their relationships

Should you represent everything with FOL?

- *Ontological commitment*: world consists of facts about objects and their relationships
- Applications (e.g. Prolog) often operate under the *closed world assumption and unique name assumption*

Should you represent everything with FOL?

- *Ontological commitment*: world consists of facts about objects and their relationships
- Applications (e.g. Prolog) often operate under the *closed world assumption and unique name assumption*
  - Facts we cannot prove are considered false

Should you represent everything with FOL?

- *Ontological commitment*: world consists of facts about objects and their relationships
- Applications (e.g. Prolog) often operate under the *closed world assumption and unique name assumption*
  - Facts we cannot prove are considered false
  - Differently named entities are assumed to be different objects

Should you represent everything with FOL?

- No difference between *types, roles* and *attributes* of objects
  - All represented by predicates: person(Bob), baker(Bob), hungry(Bob)
- Terminological facts can be handled by implication but don't have a dedicated representation
  - A statement like $dog(x) \rightarrow animal(x)$ has no "special status"
- Best suited for applications with a single source of truth
  - Hard to reason about conflicting statements e.g. the IAU does not consider Pluto a planet, astrologists do.
- Also hard to reason about probabilities, degrees of belief, temporal beliefs, etc.

Other types of representations

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts
- **Default logics** and nonmonotonic logics (ch. 11)

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts
- **Default logics** and nonmonotonic logics (ch. 11)
  - Focus on general statements that are true unless we have reason to believe otherwise

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts
- **Default logics** and nonmonotonic logics (ch. 11)
  - Focus on general statements that are true unless we have reason to believe otherwise
  - Monotonicity: if $KB_1 \models \alpha$, and $KB_1 \subseteq KB_2$, then we must have $KB_2 \models \alpha$

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts
- **Default logics** and nonmonotonic logics (ch. 11)
  - Focus on general statements that are true unless we have reason to believe otherwise
  - Monotonicity: if $KB_1 \models \alpha$, and $KB_1 \subseteq KB_2$, then we must have $KB_2 \models \alpha$
    - In default logic and other nonmonotonic logics, the above does not hold

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts
- **Default logics** and nonmonotonic logics (ch. 11)
  - Focus on general statements that are true unless we have reason to believe otherwise
  - Monotonicity: if $KB_1 \models \alpha$, and $KB_1 \subseteq KB_2$, then we must have $KB_2 \models \alpha$
    - In default logic and other nonmonotonic logics, the above does not hold
  - e.g. birds fly, penguins are birds, but penguins don't fly

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts
- **Default logics** and nonmonotonic logics (ch. 11)
  - Focus on general statements that are true unless we have reason to believe otherwise
  - Monotonicity: if $KB_1 \models \alpha$, and $KB_1 \subseteq KB_2$, then we must have $KB_2 \models \alpha$
    - In default logic and other nonmonotonic logics, the above does not hold
  - e.g. birds fly, penguins are birds, but penguins don't fly
- **Fuzzy logics**, **probabilistic logics** (ch. 12)

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts
- **Default logics** and nonmonotonic logics (ch. 11)
  - Focus on general statements that are true unless we have reason to believe otherwise
  - Monotonicity: if $KB_1 \models \alpha$, and $KB_1 \subseteq KB_2$, then we must have $KB_2 \models \alpha$
    - In default logic and other nonmonotonic logics, the above does not hold
  - e.g. birds fly, penguins are birds, but penguins don't fly
- **Fuzzy logics**, **probabilistic logics** (ch. 12)
- **Situation calculus** (ch. 14)

Other types of representations

- **Object-Oriented** representations (ch. 8), description logics (ch. 9) and inheritance (ch. 10)
  - Focus on categories (often hierarchical), which can be defined in terms of objects and their parts
- **Default logics** and nonmonotonic logics (ch. 11)
  - Focus on general statements that are true unless we have reason to believe otherwise
  - Monotonicity: if $KB_1 \models \alpha$, and $KB_1 \subseteq KB_2$, then we must have $KB_2 \models \alpha$
    - In default logic and other nonmonotonic logics, the above does not hold
  - e.g. birds fly, penguins are birds, but penguins don't fly
- **Fuzzy logics**, **probabilistic logics** (ch. 12)
- **Situation calculus** (ch. 14)
  - Focus on beliefs about a changing world

Other types of representations

Other types of representations

- As we'll see, many of these lend themselves naturally to **graph** representations (knowledge graphs).

What’s next?

Source: https://unsplash.com/photos/j642UIFMekY under Unsplash license

Semantic Web

Goals

Semantic Web

Goals

Also known as Web 3.0, it aims to make Internet data machine-readable

Semantic Web

Goals

Also known as Web 3.0, it aims to make Internet data machine-readable

- Based on standards set by the World Wide Web Consortium (W3C)

Semantic Web

Goals

Also known as Web 3.0, it aims to make Internet data machine-readable

- Based on standards set by the World Wide Web Consortium (W3C)
  - Main technologies: RDF, RDFS, OWL

Semantic Web

Goals

Also known as Web 3.0, it aims to make Internet data machine-readable

- Based on standards set by the World Wide Web Consortium (W3C)
  - Main technologies: RDF, RDFS, OWL
- Vision by Tim Berners-Lee (1999):

> I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A "Semantic Web", which makes this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The "intelligent agents" people have touted for ages will finally materialize.[9]

Semantic Web

Goals

Also known as Web 3.0, it aims to make Internet data machine-readable

- Based on standards set by the World Wide Web Consortium (W3C)
  - Main technologies: RDF, RDFS, OWL
- Vision by Tim Berners-Lee (1999):

  > I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A "Semantic Web", which makes this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The "intelligent agents" people have touted for ages will finally materialize.[9]

- Not to be confused with Web3 (based on Blockchain technology)!

Semantic Web

Goals

Also known as Web 3.0, it aims to make Internet data machine-readable

- Based on standards set by the World Wide Web Consortium (W3C)
  - Main technologies: RDF, RDFS, OWL
- Vision by Tim Berners-Lee (1999):

  > I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A "Semantic Web", which makes this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The "intelligent agents" people have touted for ages will finally materialize.[9]

- Not to be confused with Web3 (based on Blockchain technology)!
- Main theme: data on the web was (or is) optimized for human readability

Semantic Web

Goals

Also known as Web 3.0, it aims to make Internet data machine-readable

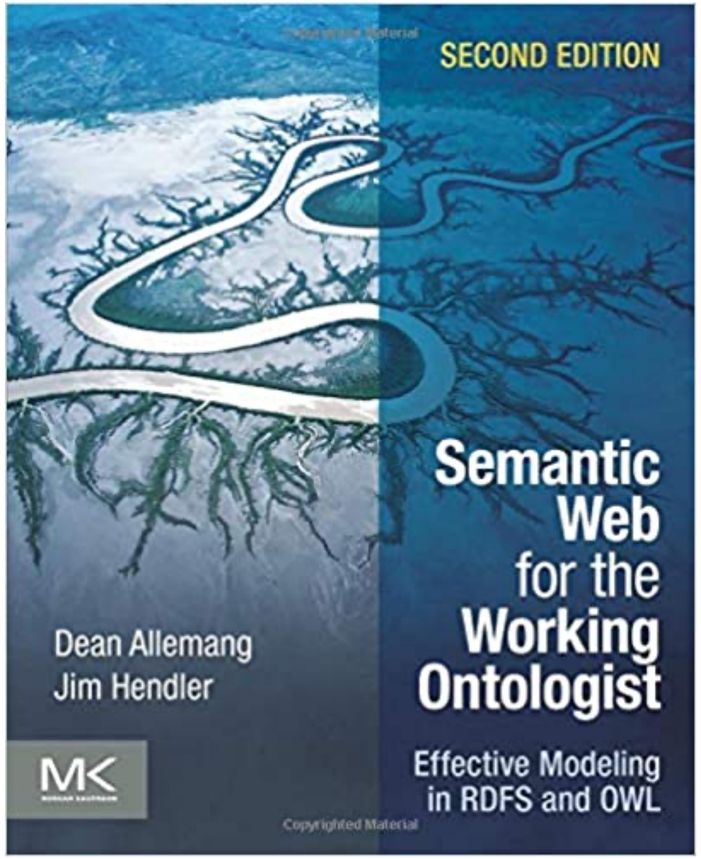
- Based on standards set by the World Wide Web Consortium (W3C)
  - Main technologies: RDF, RDFS, OWL
- Vision by Tim Berners-Lee (1999):

  > I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A "Semantic Web", which makes this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The "intelligent agents" people have touted for ages will finally materialize.[9]

- Not to be confused with Web3 (based on Blockchain technology)!
- Main theme: data on the web was (or is) optimized for human readability
  - We need to make it more machine-readable

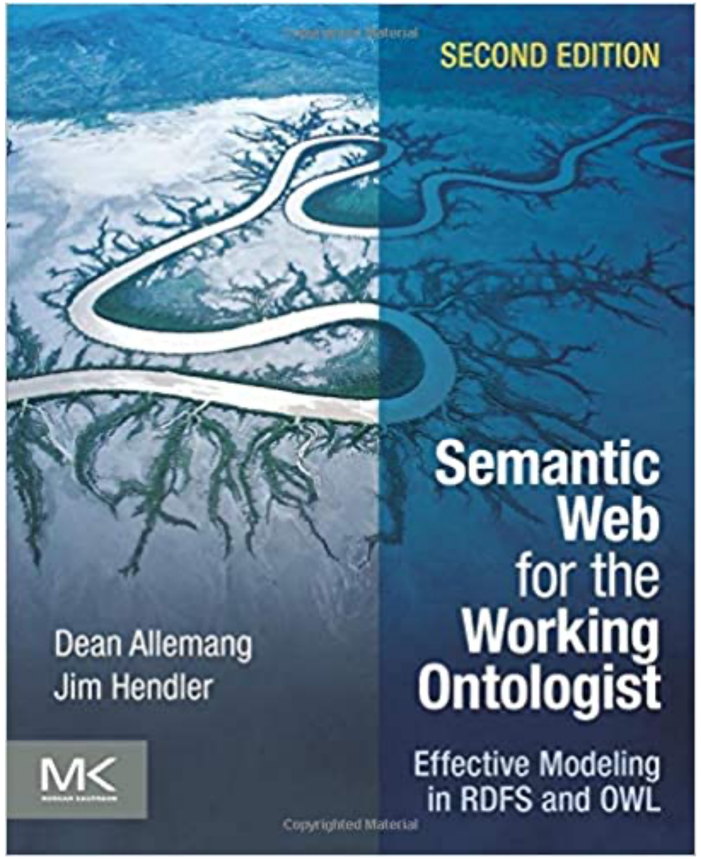
Semantic Web

Textbook



Semantic Web

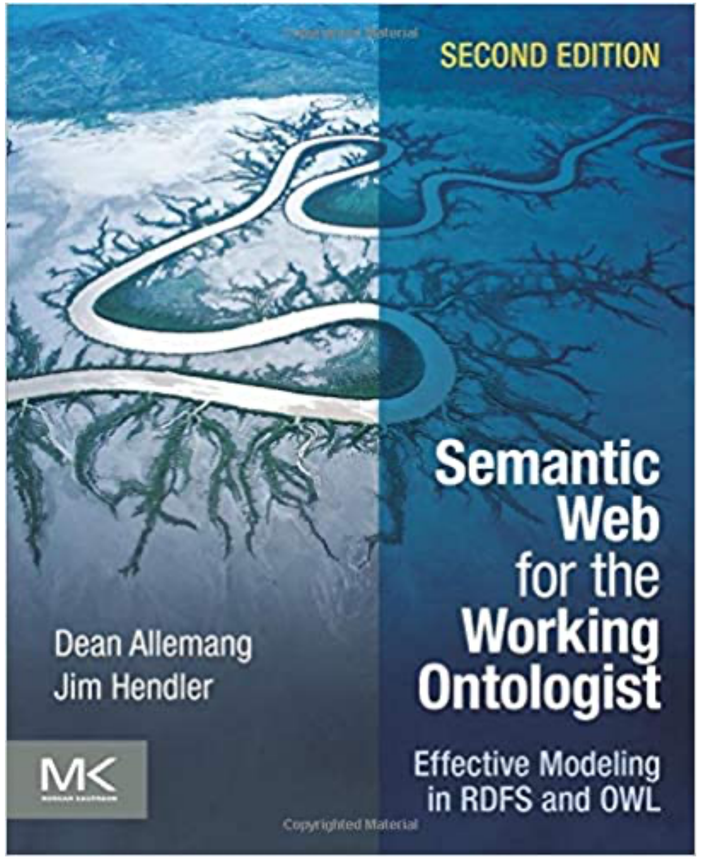
Textbook



- Available in the digital library

Semantic Web

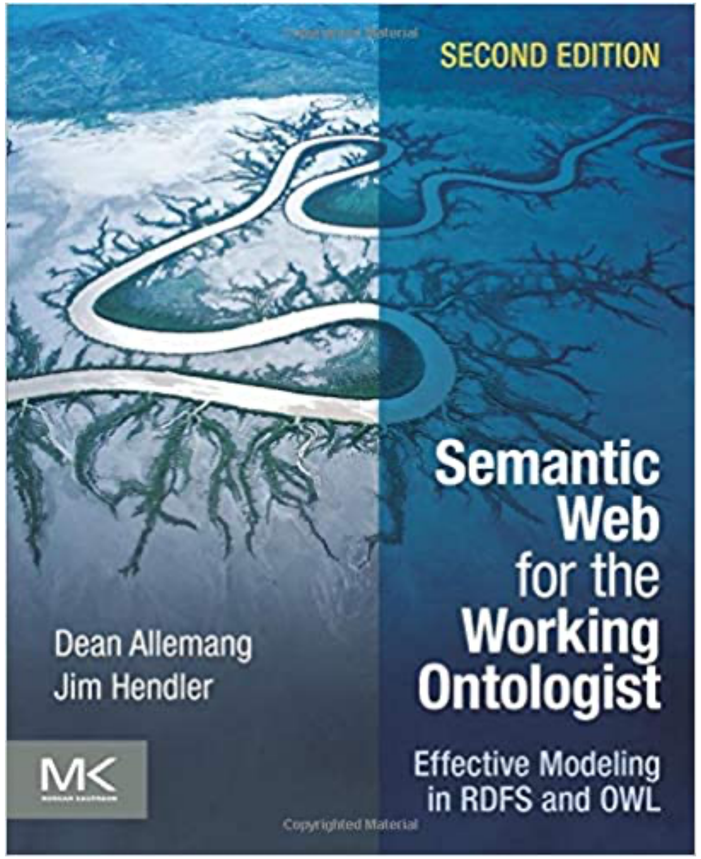
Textbook



- Available in the digital library
- Very accessible language

Semantic Web

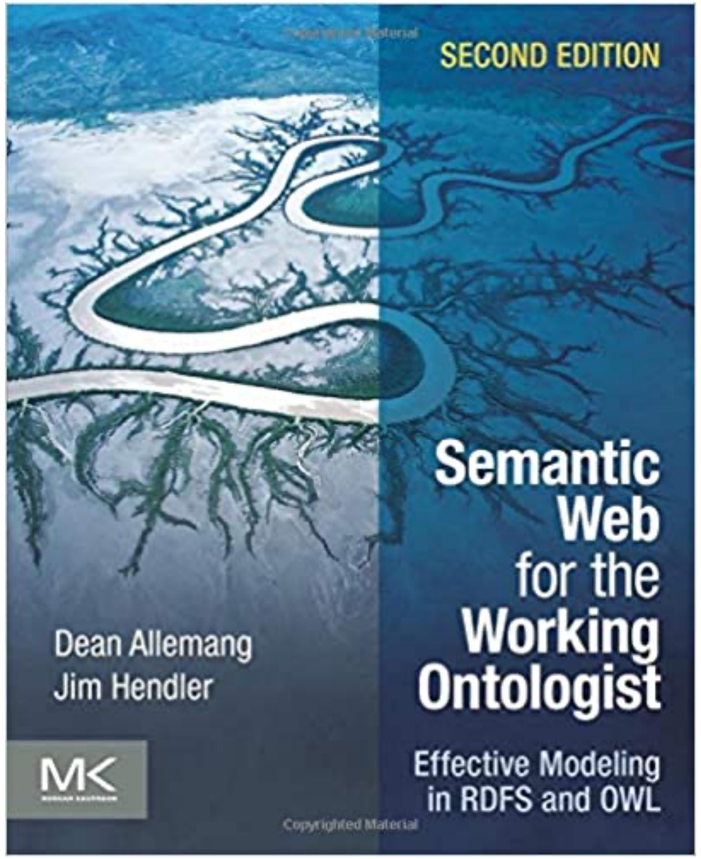
Textbook



- Available in the digital library
- Very accessible language
- There is also a third edition (I'm basing lectures off the second)

Semantic Web

Textbook



- Available in the digital library
- Very accessible language
- There is also a third edition (I'm basing lectures off the second)
- Direct quotes from the book will be in gray and italics

Semantic Web

Representation needs

Semantic Web

Representation needs

A very unique and relevant domain with its own needs for representation

Semantic Web

Representation needs

A very unique and relevant domain with its own needs for representation

- AAA slogan: **A**nyone can say **A**nything about **A**ny topic

Semantic Web

Representation needs

A very unique and relevant domain with its own needs for representation

- AAA slogan: **A**nyone can say **A**nything about **A**ny topic
  - Multiple sources of truth contradicting facts

Semantic Web

Representation needs

A very unique and relevant domain with its own needs for representation

- AAA slogan: **A**nyone can say **A**nything about **A**ny topic
  - Multiple sources of truth contradicting facts
- Open world assumption: new information may be added at any time, so we often treat things we cannot prove as simply unknown

Semantic Web

Representation needs

A very unique and relevant domain with its own needs for representation

- AAA slogan: **A**nyone can say **A**nything about **A**ny topic
  - Multiple sources of truth contradicting facts
- Open world assumption: new information may be added at any time, so we often treat things we cannot prove as simply unknown
- Non-unique naming: different sources may use different names for the same entity

Semantic Web

Representation needs

A very unique and relevant domain with its own needs for representation

- AAA slogan: **A**nyone can say **A**nything about **A**ny topic
  - Multiple sources of truth contradicting facts
- Open world assumption: new information may be added at any time, so we often treat things we cannot prove as simply unknown
- Non-unique naming: different sources may use different names for the same entity
- Information that has been typically presented in a human-readable way now needs to be machine-readable as well.

Semantic Web

Motivating examples

Semantic Web

Motivating examples

Example 1

Semantic Web

Motivating examples

Example 1

- A hotel chain (Mongotel) has a list of branches in its website.

Semantic Web

Motivating examples

Example 1

- A hotel chain (Mongotel) has a list of branches in its website.
- One of these branches is near a certain park

Semantic Web

Motivating examples

Example 1

- A hotel chain (Mongotel) has a list of branches in its website.
- One of these branches is near a certain park
- The park also lists nearby hotels in its website

Semantic Web

Motivating examples

Example 1

- A hotel chain (Mongotel) has a list of branches in its website.
- One of these branches is near a certain park
- The park also lists nearby hotels in its website
- But the Mongotel branch is not listed on the park's website!

Semantic Web

Motivating examples

Semantic Web

Motivating examples

Example 2

Semantic Web

Motivating examples

Example 2

- A trip-planning site lists all hotels in a city and their addresses

Semantic Web

Motivating examples

Example 2

- A trip-planning site lists all hotels in a city and their addresses
- A geolocation website finds the shortest path from any two addresses

Semantic Web

Motivating examples

Example 2

- A trip-planning site lists all hotels in a city and their addresses
- A geolocation website finds the shortest path from any two addresses
- But a user who wants the closest hotel to a point of interest has to copy-paste a lot of addresses to find which is closest to the target!

Semantic Web

Motivating examples

Semantic Web

Motivating examples

Example 3

Semantic Web

Motivating examples

Example 3

- A website lists objects in the solar system

Semantic Web

Motivating examples

Example 3

- A website lists objects in the solar system
- The IAU changed Pluto’s standing in 2006 from “planet” to “dwarf planet”

Semantic Web

Motivating examples

Example 3

- A website lists objects in the solar system
- The IAU changed Pluto's standing in 2006 from "planet" to "dwarf planet"
- On the webpage of Pluto, the "category" field says "dwarf planet"

Semantic Web

Motivating examples

Example 3

- A website lists objects in the solar system
- The IAU changed Pluto's standing in 2006 from "planet" to "dwarf planet"
- On the webpage of Pluto, the "category" field says "dwarf planet"
- But Pluto still appears on the main page as a planet !

Semantic Web

Motivating examples

Semantic Web

Motivating examples

Example 4 (not in book)

Semantic Web

Motivating examples

Example 4 (not in book)

- I can easily find a list of cities that hosted a FIFA World Cup finals with a search

Semantic Web

Motivating examples

Example 4 (not in book)

- I can easily find a list of cities that hosted a FIFA World Cup finals with a search
- I can also easily find a list of cities that have hosted the Olympics

Semantic Web

Motivating examples

Example 4 (not in book)

- I can easily find a list of cities that hosted a FIFA World Cup finals with a search
- I can also easily find a list of cities that have hosted the Olympics
- But there’s no easy way for me to find a city that has hosted both!

Group exercise

Discuss in groups

- What makes each of these examples “dumb”?
- How would you go about solving them?
  - You can start simple, by assuming for example all the data is maintained by a single actor, then consider more complicated scenarios
- What are potential complications?

The four examples:

1. Hotel lists itself as close to park, but park doesn’t list them

2. Copy-pasting required to use data from one application (hotel search) into another (geolocation)

3. Pluto categorized as a dwarf planet in its page, but as regular planet in the “planets of the solar system” page

4. How to find cities that have hosted both a FIFA finals and an Olympics?

Results of discussion

- Some queries may be too specific to be specifically supported
- Data may be represented differently
- How to know that two different sources are talking about the same thing?
- Conflicting information: who to trust?
  - Sometimes we might want to aggregate, sometimes we might want to trust a single source
- Any website may either define a concept or refer to another source of truth for that concept
- Security concerns, one point of failure

Applications vs infrastructure

- The Web is “dumb” in the sense that data are inconsistent, out of synch, disconnected.
- Given the right data, it’s not too hard to write a “smart” application to solve any of these problems.
- The hard part is how to get the data in the first place!
- *“What can the Web infrastructure provide to improve the consistency and availability of Web data?”*
- *“Infrastructure is something that one would rather buy than build”*
- Adoption of Semantic Web standards and advancements in Natural Language Processing (NLP) already help build applications with the desired properties
  - But it’s important to understand how!

Semantic data

Local database approach

An initial solution to some of these examples (1,3,4) would be to back data in a database

- Pages are the result of queries to the DB rather than display static content
- If data changes, page is automatically updated (preserves consistency)
- However, doesn’t work well if different entities control different parts of the data (example 1)

Semantic data

Another approach would be to write a program to keep data from various sources updated

- Requires custom code for each linkage of data
- Data linked at the level of pages
- Changes in one source may affect applications used by other sources
- Still requires some coordination

Network Effect

But who will be tasked with converting all this data to RDF?

- Little incentive at the beginning (why do it if no one is consuming the info?)
- The more applications convert, the higher the incentive
- Same network effect that drove the original Web

Non-Unique naming

- Different sources may disagree on the properties of a real world object
- Or they may call the same object by different names
    - “Pluto as viewed by the IAU” is a different notion than “Pluto as viewed by astrologists”
    - Pluto is also known by different names (e.g. plu134340)

Fundamental concepts

Practical considerations

Fundamental concepts

Practical considerations

- The AAA slogan (Anyone can say Anything about Any topic)

Fundamental concepts

Practical considerations

- The AAA slogan (Anyone can say Anything about Any topic)
- Open world assumption

Fundamental concepts

Practical considerations

- The AAA slogan (Anyone can say Anything about Any topic)
- Open world assumption
- Non-unique naming

Fundamental concepts

Practical considerations

- The AAA slogan (Anyone can say Anything about Any topic)
- Open world assumption
- Non-unique naming
- Network effect (the more people use it, the more attractive using it becomes)

Core technologies

- **RDF - Resource Description Framework**. Models data as triplets (subject, predicate, object)
- **RDFS - RDF Schema language**. Enhances RDF with notions of classes, subclasses and properties.
- **OWL - Web Ontology Language**. adds more vocabulary for describing properties and classes (e.g. disjointness, cardinality, symmetry of properties…)
    - Related to Description Logics (ch. 9 of the Brachman & Levesque book)