Foundations of the Semantic Web: Ontology Engineering

Building Ontologies 1

Alan Rector & colleagues

Goals for this module: for you

- Be able to implement an ontology representation in OWL-DL
  - Be able to elicit a conceptualisation
  - Be able to formulate an ontology representation
  - Be able to implement the ontology representation in OWL-DL
    - Or be able to say you can't
    - To understand the limits of OWL-DL ontologies
  - Be able to test the resulting ontology implementation
  - Be ready to apply ontology representations in any of several use cases
    - In one week, we can't build the applications…
      …but to build an ontology is only a means to building applications
      - Without applications ontologies are pointless

Goals for this Module: For us

- Still experimental – we need your feedback
  - Feedback
    - On tools – we treat this as a User Centred Design experiment
      - Please be patient
        - The good news is they are getting better
    - On the course
      - Did the content work for you?
      - What other content would you like?
      - Balance of labs and lecture
      - Content of labs
    - For the Semantic Web Best Practice Working Group
  - New ideas

Mechanics - reminder

- Assessment
  - 30% lab
  - 30% Mini project
  - 40% Exam
- All labs to be handed in by number via Boddington – see lab handout
- Theoretical deadline – end term before Christmas
  - Will allow to go until the first day of exam period but don't advise it
    - You are better to study for the exams!

Ontologies and Ontology Representations

- "Ontology" – a word borrowed from philosophy
  - But we are necessarily building logical systems
    - "Physical symbol systems"
      - Simon, H. A. (1969, 1981). *The Sciences of the Artificial*, MIT Press
- "Concepts" and "Ontologies"/ "conceptualisations" in their original sense are psychosocial phenomena
  - We don't really understand them
- "Concept representations" and "Ontology representations" are engineering artefacts
  - At best approximations of our real concepts and conceptualisations (ontologies)
    - And we don't even quite understand what we are approximating

Ontologies and Ontology Representations (cont)

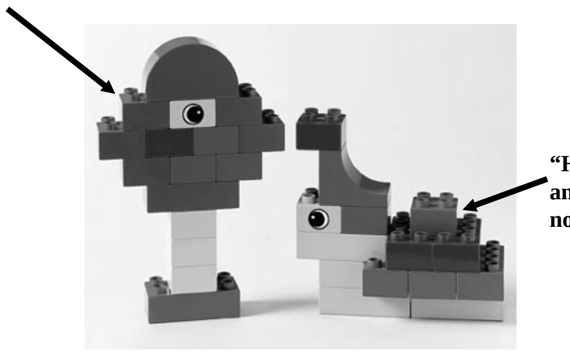
- Most of the time we will just say "concept" and "ontology" but whenever anybody starts getting religious, remember…
  - ***It is only a representation!***
    - We are doing engineering, not philosophy – although philosophy is an important guide
- *There is no one way!*
  - But there are consequences to different ways
    - and there are wrong ways
      - and better or worse ways for a given purposes
  - *The test of an engineering artefact is whether it is fit for purpose*
    - Ontology representations are engineering artefacts

Why build an ontology

- Interworking and information sharing
  - Providing a well organised controlled vocabulary
- Indexing complex information
  - "Knowledge is fractal"
    - Ontologies are fractal
      - Self similar structure at every level of granularity (detail)
- Combat combinatorial explosions
  - The exploding bicycle
    - "Conceptual Lego"
      - A "dictionary and grammar" instead of a "phrasebook"

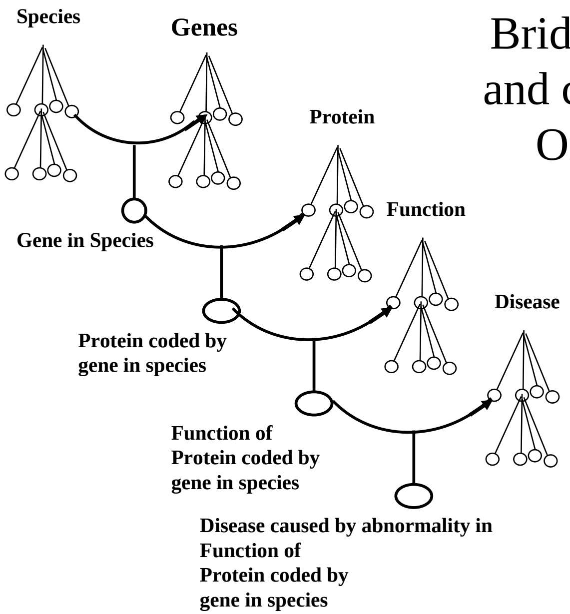
Logic-based Ontologies: Conceptual Lego: A BioInformatics View

"SNPolymorphism of CFTRGene causing Defect in MembraneTransport of ChlorideIon causing Increase in Viscosity of Mucus in CysticFibrosis…"

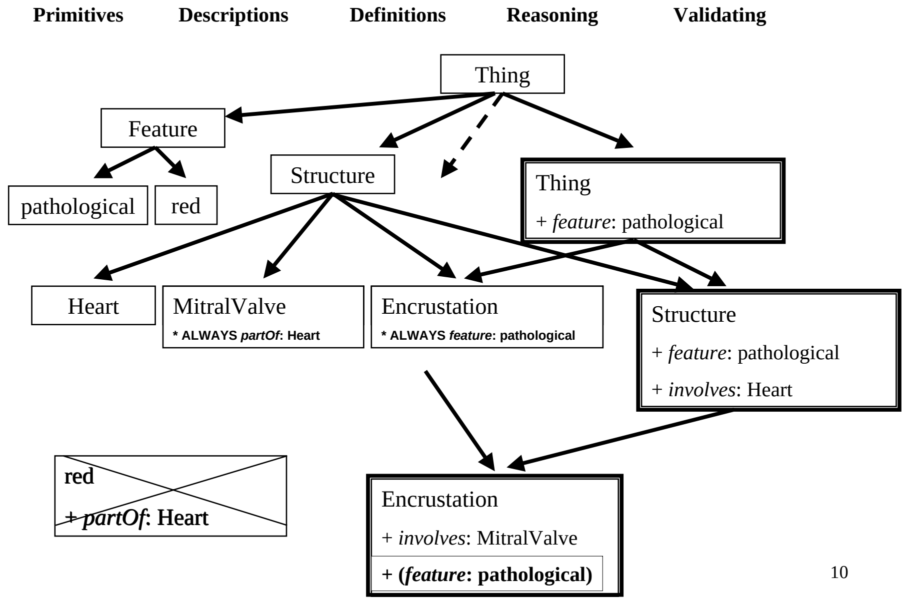


"Hand which is anatomically normal"

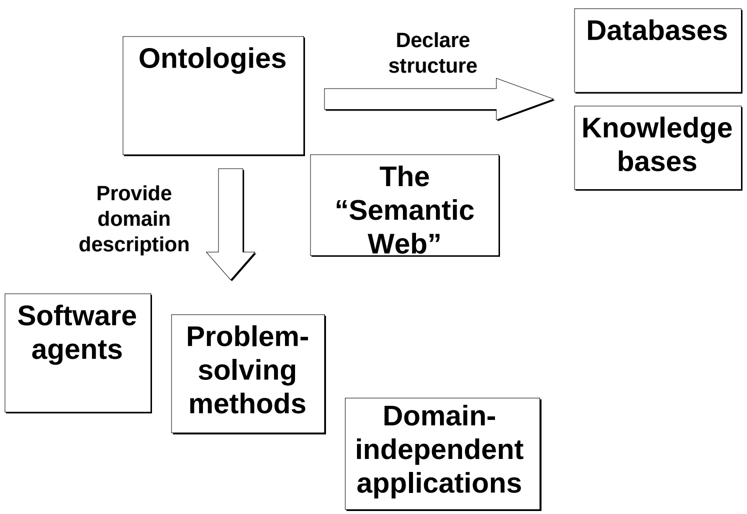
Bridging Scales and context with Ontologies



Logic Based Ontologies: A crash course



An Ontology should be just the Beginning



And beware Ontologies are not databases!

- Ontologies are (mostly) about the classes –
  - Can be used to represent database ***schemas***
    - *What must be true of any database consistent with the schema*
  - The Terminology
    - *What must be true of any concept consistent with the ontology*
  - The "T-Box" – for "terminology box"
- Limited functionality for individuals ('instances')
  - Primarily to help define classes
    - The class of John's shirts, The class of cities in Japan
  - To describe individuals use
    - A database
    - Triple representation (RDF or Topic Maps)
    - An instance store
    - Perhaps with an ontology as the schema
  - Individuals in ontologies (The "A-Box") poorly understood and very high computational complexity

Approach

- Design patterns
  - Analogous to Java design patterns
    - Standard ways to do things
      - Someday they will be supported by tools, but today you have to do it yourself
  - Being codified by Semantic Web Best Practice Working Group
- Elephant traps
  - Common errors & misconceptions
    - Especially those that seem to work at first
- Foundations of knowledge representation
  - 200 to 2000 years of experience & mistakes you need not repeat
- Common dilemmas & tradeoffs
  - Things for which we don't have a perfect answer

Why does the W3C Semantic Web need a "Best Practice working Group"?

- There is no established "best practice"
  - It is new; We are all learning
  - A place to gather experience
  - A catalogue of things that work – Analogue of Software Patterns
    - Some pitfalls to avoid

  – ***…but there is no one way***

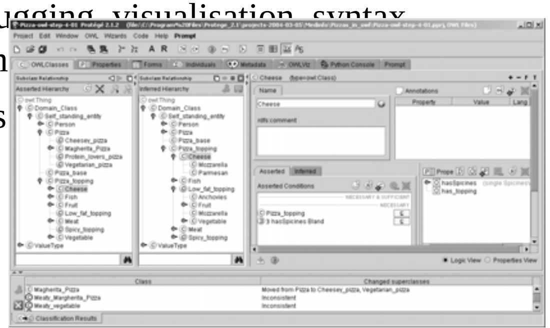
- Learning to build ontologies
  - Too many choices
    - Need starting points for gaining experience
- Provide requirements for tool builders

You can contribute to identifying "best practice"

- Please give us feedback
  - Your questions and experience
    - On the SW in general: **semanticweb@yahoogroups.com**
    - For specific feedback to SWBP
      - Home & Mail Archive: **http://www.w3.org/2001/sw/BestPractices/** **public-swbp-wg@w3.org**

Protégé OWL: New tools for ontologies

- Transatlantic collaboration
- Implement robust OWL environment within PROTÉGÉ framework
- New ideas for debugging, visualisation, syntax ontology managem
- Tell us what works and ideas for improvements



Protégé-OWL & CO-ODE

- Joint work: Stanford & U Manchester +
  Southampton & Epistemics
  - Please give us feedback on tools – mailing lists & forums at:
    - **protege.stanford.edu**
    - **www.co-ode.org**
- Don't beat your head against a brick wall!
  - Look to see if others have had the same problem; If not…
  - ***ASK!***
    - ***We are all learning.***

OWL-DL & Classification

- Not all of OWL-DL can yet be implemented
  - We will deal mostly with the subset that can be classified using FaCT, Racer or FaCT++
  - Not all of the things that are implemented scale successfully
    - All classifiers are worst-case exponential (or worse)
- Racer
  - Standard classifier for Protégé OWL
- FaCT++
  - New classifier being developed here
    - Faster, more expressive, better, …
      - but not quite yet done
- We will try to provide warnings of things which cannot be classified or do not scale
  - But you may discover new things on your own

Example Ontologies for this Module

- Pizzas
  - For the mechanics of OWL and Protégé/OWL
    - Simple – no ontological problems, just mechanics
- Animals for best practice examples and ontology building
  - The example for you to work from
    - Also for examples of parts and wholes
- The University and courses
  - Your job is to build an ontology for the University by analogy to the examples
    - with some specific help
    - Leads on to major ontological issues
- Simple Upper Ontology
  - To put it together
    - Mostly about the University

Building Ontologies

- Basic Concepts and Mechanics

Why it's hard (1)

- Clash of intuitions
  - Subject Matter Experts motivated by custom & practice
    - Prototypes & Generalities
  - Logicians motivated by logic & computational tractability
    - Definitions and Universals
- Transparency & predictability vs Rigour & Completeness
- Neophytes (you?) caught in the muddled middle

Why it's hard (2)

- Conflation of Models
  - Meaning: Correctness of Classification & retrieval
  - Indexing: Task of discovery, search, or finding
  - Use: Task of data entry, decision support, …
  - Acquisition: Task of capturing knowledge
- Assuring quality & managing change
  - Quality assurance: Criteria for whether it is 'correct'
  - Evolution Coping with change
  - Regression testing Controlling changes & maintaining Quality

Why its hard (3)

- Confusion of terminology and usage
  - Religious wars over words and assumptions
- The intersection of
  - Linguistics
  - Cognitive science
  - Software engineering
  - Philosophy
  - Human Factors
- A jumble of syntaxes

Vocabulary

- "Class" ≈ "Concept" ≈ "Category" ≈ "Type"
- "Instance" ≈ "Individual"
- "Entity" ≈ "object", Class or individual
- "Property" ≈ "Slot" ≈ "Relation" ≈ "Relationtype" ≈ "Attribute" ≈ Semantic link type" ≈ "Role"
  - but be careful about "role"
    - Means "property" in DL-speak
    - Means "role played" in most ontologies
      - E.g. "doctor_role", "student role" …

Syntaxes

- Three official syntaxes + Protégé-OWL syntax
  - Abstract syntax-- -Specific to OWL
  - *N3* ---------------- *-OWL & RDF* *-used in all SWBP documents*
  - XML/RDF ------- -very verbose, not for human consumption
  - *"German DL"*---- *-very concise, symbolic*
  - First order logic - - complete but more powerful than DL
  - *Protégé-OWL*---- *-Compact, derived from DL syntax*
  - Paraphrase-------- -Verbose but precise
- This tutorial uses simplified abstract syntax
  - someValuesFrom → ***some*** ∃
  - allValuesFrom → ***only*** ∀
  - intersectionOf → AND ⊓
  - unionOf → OR ⊔
  - complementOf → NOT ¬
  - complete ***definition*** ***necessary & sufficient***
  - partial ***description*** ***necessary***
- Protégé/OWL can generate all syntaxes except German

Why its hard (4)

- Clash with vocabulary and practice of related software disciplines

Clash with intuitions of related fields

- Object Oriented Programming
  - Java,a C++, Smalltalk, etc.
    - But OO programming is not knowledge representation
- Object Oriented Design (Databases)
  - But data models are not ontologies either
    - Although UML is often a good starting point
      - Additional a-logical issues
        - » Difference between attributes and relations
        - » Issues of life cycle and handling of aggregation$
        - » Notion of an instance
        - » Implicitly "closed world"
- Frame based systems, Semantic Nets,… Traditional AI
  - Where it all started but real differences
- RDF(S), Topic Maps and other node-and-arc symbolisms
  - "What's in a link?"
  - The battles in standards committees continue

Summary of Approach
Steps in developing an Ontology (1)

1. Establish the purpose
   - Without purpose, no scope, requirements, evaluation,
2. Informal/Semiformal knowledge elicitation
   - Collect the terms
   - Organise terms informally
   - Paraphrase and clarify terms to produce informal concept definitions
   - Diagram informally
3. Refine requirements & tests

Summary of Approach
Steps in implementing an Ontology (2)

4. Implementation
   – Develop normalised schema and skeleton
   – Implement prototype recording the *intention as a paraphrase*
      • Keep track of what you meant to do so you can compare with what happens
         – Implementing logic-based ontologies is programming
   – Scale up a bit
      • Check performance
   – Populate
      • Possibly with help of text mining and language technology
5. Evaluate & quality assure
   – Against
   – Include tests for evolution and change management
   – Design regression tests and “probews”
6. Monitor use and evolve
   – ***Process not product!***

If this were three modules…

1. Knowledge elicitation and analysis
   – A quick overview
2. **Implementation**
   – **A solid introduction**
3. Evolution, ontology alignment, and management
   – Left for another module
      • But a major motivation for the methods taught in this module
         – Normalisation and documentation of intentions

Plan of Labs

- Monday – the mechanics of OWL in Protégé Owl
  – The pizza example
- Tuesday – Ontology building the life cycle
  – A more realistic example
  – Start building the University example
    • On the pattern of the lecture example of animals
- Wednesday
  – Problems and tricks of the trade
  – DL problems (IH)
- Thursday
  – More on patterns and parts and whole
- Friday
  – Upper ontologies and clarification of the mini project