OWL Semantics
COMP62342

Sean Bechhofer
sean.bechhofer@manchester.ac.uk
Uli Sattler
uli.sattler@manchester.ac.uk

General thoughts about ontologies & ontology languages

MANCHESTER 1824

The University of Manchester

Toward Knowledge Formalization

- Acquisition Process
  - Elicit tacit knowledge
  - A set of terms/concepts
- More explicit information
  - Hierarchy and other relations
  - Categorising (modifiers)
  - Constraints and definitions

Nodes/Arcs representing a relationship (default IS-A)

What IS-A Is and Isn't: An Analysis of Taxonomic Links in Semantic Networks (Ron Brachman)

- leading to some form of **knowledge base** or **ontology…**

An *ontology* is

- in Philosophy: the study of the nature of being, becoming, existence, or reality.
- in CS: a *knowledge base*, i.e, an engineering artefact, written in a *formal language* (in contrast to natural language)

An **ontology** is
a **representation** of the **shared knowledge** for a **community**

An *ontology*

e.g., furniture, animals, biology, medicine,…

- is used to
  - provide the **intended meaning** of the **vocabulary**
  - describe a certain **conceptualisation** in a domain of interest
- is usually
  - a **vocabulary** (i.e., terms) plus
  - explicit characterisations of the **assumptions** made
    in interpreting those terms
- is expressed in some *ontology language, e.g. OWL*
  - nearly always includes some notion of hierarchical **classification** (is-a)
- Ontology languages allow
  - the **definition** of classes through description of their characteristics or
  - other axioms or
  - constraints or
  - rules…
  - often based on some **logic**
    - ➡ allows us to use **reasoning** to help in
      management & deployment of the knowledge captured in an ontology!

Ontology, taxonomies, terminologies…?

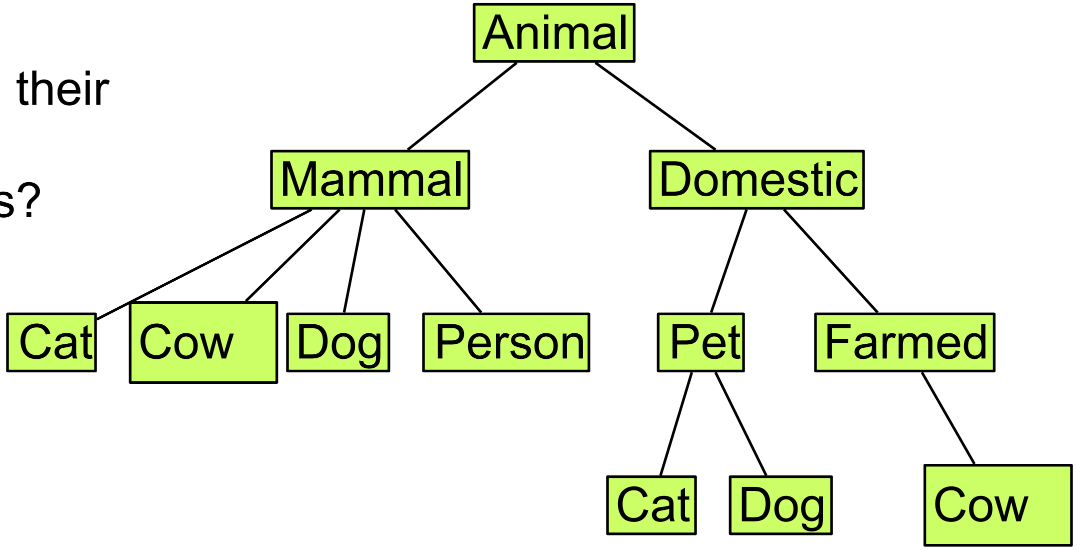
An attempt at clarifying these terms:

Controlled Vocabulary	=	{terms for concepts}
Taxonomy	=	CV + hierarchy
Classification system	=	Taxonomy + principles
Thesaurus	=	Taxonomy + more labels
Terminology	=	… + glossary/explanations
Ontology	=	… + logical axioms + well-defined semantics + reasoning + ….

What is a Taxonomy?

- An organisation of entities
  - typically hierarchical
  - subclass/is-a relationships

- Organisationally Rigid
  - Terms are usually *put* in their proper place
  - Multiple places for terms?



- **Impoverished** descriptions
  - Cats are carnivores
    - Why?
    - What is it to be a Carnivore?
    - What if we say something is a Carnivore and a Herbivore?

MANCHESTER 1824

The University of Manchester

OWL - general

OWL: The Web Ontology Language

“The W3C OWL 2 Web Ontology Language (OWL) is a
Semantic Web language designed to
represent rich and complex knowledge about
things, groups of things, and relations between things.
OWL is a computational logic-based language such that
knowledge expressed in OWL can be reasoned with by computer programs
either to
verify the consistency of that knowledge or to
make implicit knowledge explicit.
OWL documents, known as ontologies, can be published in the
World Wide Web and may refer to or be referred from other OWL ontologies.

OWL is part of the W3C's Semantic Web technology stack,
which includes RDF [RDF Concepts] and SPARQL [SPARQL].”

From http://www.w3.org/TR/owl-primer/

Requirements from this (1)

“The W3C OWL 2 Web Ontology Language (OWL) is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. OWL is a computational logic-based language such that knowledge expressed in OWL can be reasoned with by computer programs either to verify the consistency of that knowledge or to make implicit knowledge explicit. OWL documents, known as ontologies, can be published in the World Wide Web and may refer to or be referred from other OWL ontologies.

Expressive!

OWL is part of the W3C's Semantic Web technology stack, which includes RDF [RDF Concepts] and SPARQL [SPARQL].”

From http://www.w3.org/TR/owl-primer/

MANCHESTER 1824

The University of Manchester

Requirements from this (2)

"The W3C OWL 2 Web Ontology Language (OWL) is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. OWL is a computational logic-based language such that knowledge expressed in OWL can be reasoned with by computer programs either to verify the consistency of that knowledge or to make implicit knowledge explicit. OWL documents, known as ontologies, can be published in the World Wide Web and may refer to or be referred from other OWL ontologies.

OWL is part of the W3C's Semantic Web technology stack, which includes RDF [RDF Concepts] and SPARQL [SPARQL]."

Based on logic - but which?

From http://www.w3.org/TR/owl-primer/

Requirements from this (3)

"The W3C OWL 2 Web Ontology Language (OWL) is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. OWL is a computational logic-based language such that knowledge expressed in OWL can be reasoned with by computer programs either to verify the consistency of that knowledge or to make implicit knowledge explicit. OWL documents, known as ontologies, can be published in the World Wide Web and may refer to or be referred from other OWL ontologies.

OWL is part of the W3C's Semantic Web technology stack, which includes RDF [RDF Concepts] and SPARQL [SPARQL]."

Web compatible syntax

From http://www.w3.org/TR/owl-primer/

"Expressive": Ontologies versus Taxonomies

- Taxonomy: hierarchy of is-a/subsumption relationships
- Ontology can *represent rich and complex knowledge about*
  - **things,** e.g.
    - Bob is a Calf
    - Mary is the mother of Bob
  - **groups of things** and **relations between things**:
    - **Definitions** e.g.,
      - A Herbivore is an an Animal that eats only Plants.
      - A Calf is a Young Cow
      - Cows are Herbivores
    - **Constraints** e.g.,
      - Carnivores are not Herbivores (and vice versa)
      - Calfs are Playful and drink some Milk
      - being-a-daughter-of implies being-a-child-of

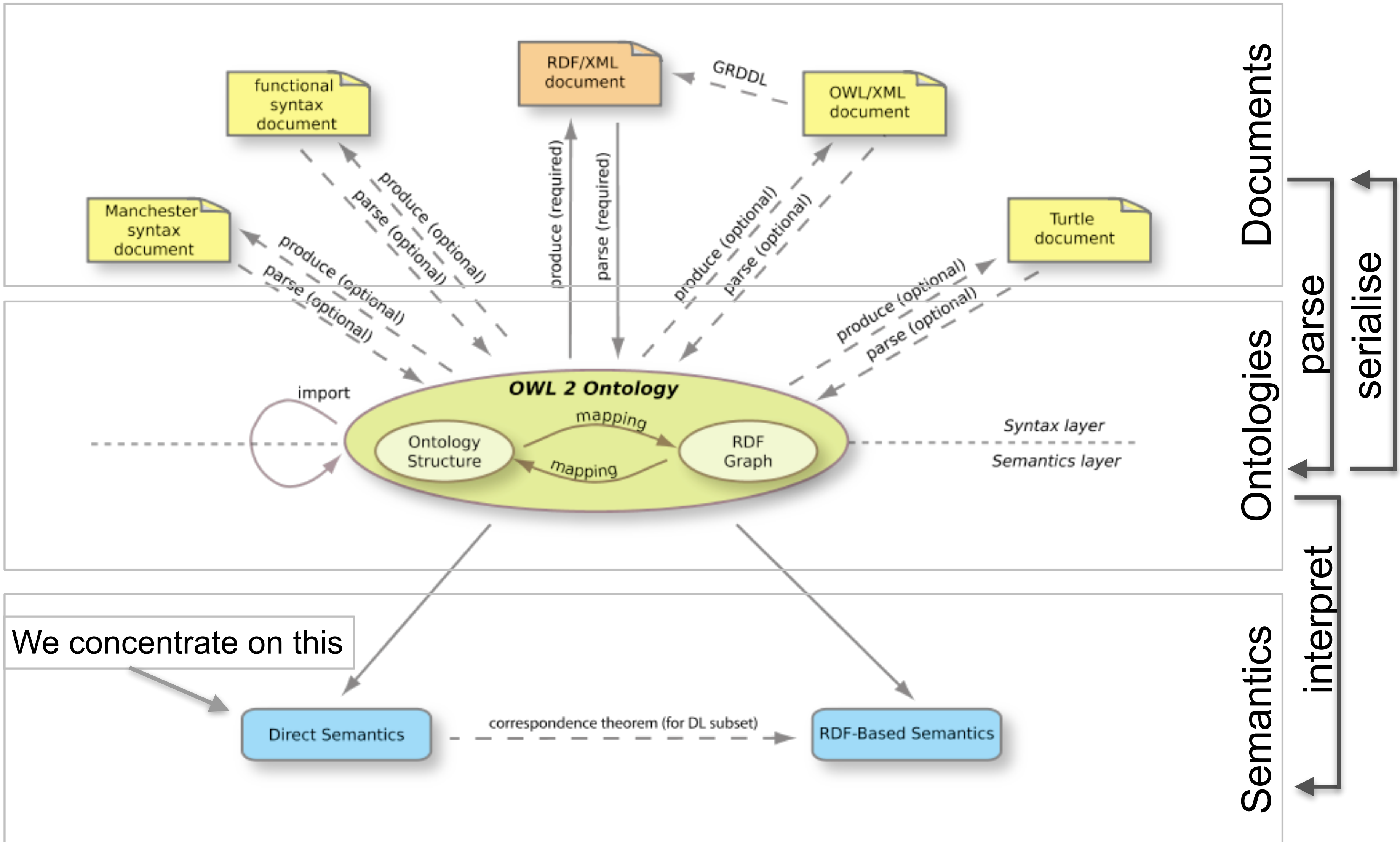
- Implicit knowledge in the above:
  – Herbivores eat only Plants
  – Bob is Playful, Young, and eats only Plants
  – …

OWL - syntax

OWL: Syntax and Semantics

- OWL is a (formal) language, so we consider its
  - **syntax**:
    - what is/isn’t a legal OWL (class/property) expression/axiom/ ontology/…?
    - what can an OWL parser accept?
    - should be web compatible!
    - see COMP60332 for syntax of logics!
  - **semantics**:
    - what does an OWL (class/property) expression/axiom/ ontology… stand for/mean?
    - what can we conclude from an OWL ontology?
    - should be based on logic - but which?

An Overview



From the OWL 2 Primer

OWL Syntax: entities

Find the "Show…Syntax" "Hide …Syntax" buttons!

Entities

- are basic building blocks of an OWL ontology
- check out https://www.w3.org/TR/owl2-primer/
- fall into 3 main categories:
  - **Class Names**:
    - e.g., Animal, Person, Idea, Table, Grass, Water
    - stand for sets of things
  - **Property Names**:
    - e.g., eats, likes, hasPart, hasChild, hasParent, isMarriedTo
    - stand for relations between things
  - **Individual Names**:
    - e.g., Peter, Paul, Mary
    - stand for individual things

OWL Syntax: descriptions

- **Descriptions** (aka **class expressions**) stand for sets of elements
- Examples:

  Animal **that** eats **only** Animal

  eats **some** (**not** Animal)

  not (eats **only** Animal **and** eats **some** Animal)

description ::= **conjunction** 'or' **conjunction** { 'or' **conjunction** }
| **conjunction**
conjunction ::= **classIRI** 'that' ['not'] **restriction**
{ 'and' ['not'] **restriction** }
| **primary** 'and' **primary** { 'and' **primary** }
| **primary**
primary ::= ['not'] (**restriction** | **atomicClass**)
restriction ::= **Property** 'some' **primary**
| **Property** 'only' **primary**
atomicClass ::= [A-Z][a-zA-Z]* *(in camel case)*
Property ::= [a-z][a-zA-Z]* *(in camel case)*

Grammar is a slightly modified subset of the one given in: http://www.w3.org/TR/owl2-manchester-syntax/

OWL Syntax: axioms

- **Axioms** (aka propositions, statements)
  – can be true or false
  – are often formulated in a **frame**

e.g. in Protégé or Manchester syntax

- Examples

  Class: CarnivorousAnimal **EquivalentTo:**
  Animal **that** eats **only** Animal

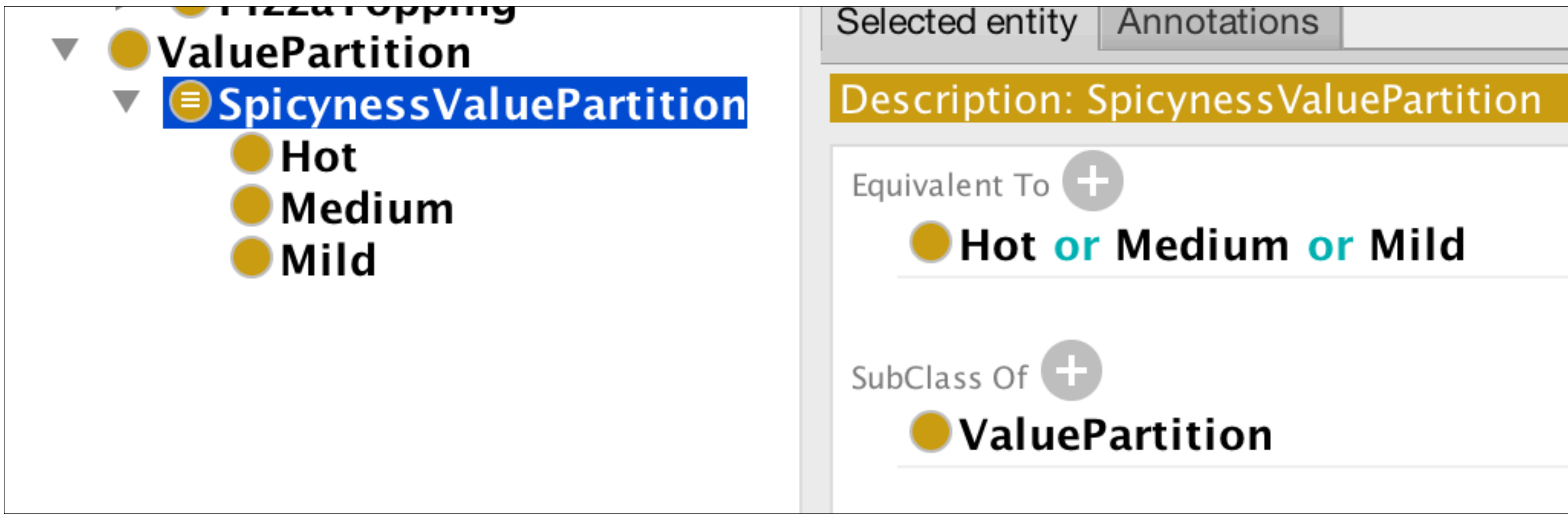
  Class: Cow **SubClassOf:** eats **some** (**not** Animal)

  Class: ConfusedCow **SubClassOf:**
  (eats **only** Animal **and** eats **some not** Animal)

- What does it all mean!?

classFrame ::= 'Class:' **atomicClass**
{ 'Annotations:' **annotation** { ',' annotation }
| 'SubClassOf:' **description** { ',' annotation }
| 'EquivalentTo:' **description** { ',' annotation } }

Axioms in Protégé?

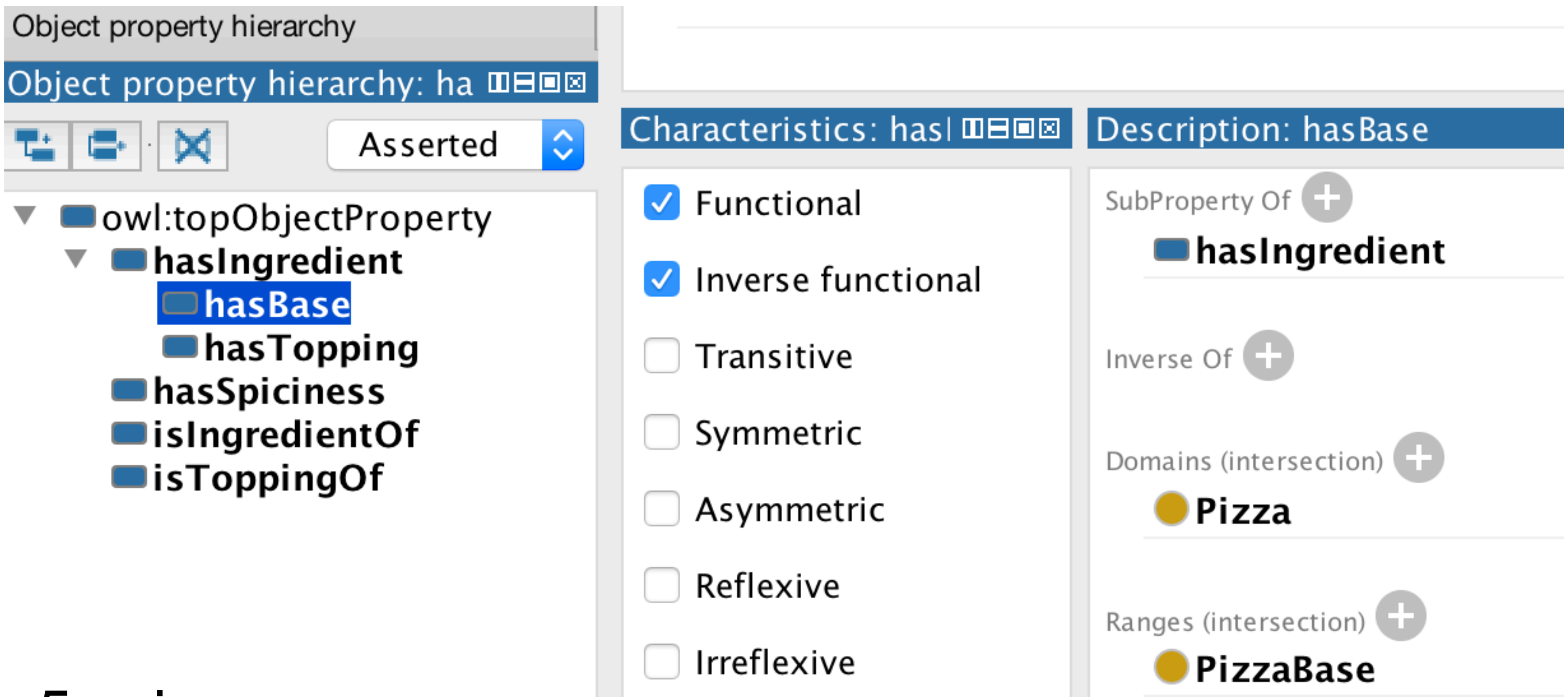


2 axioms:

SpicynessValuePartition **EquivalentTo:**
(Hot **or** Medium **or** Mild)

SpicynessValuePartition **SubClassOf:** ValuePartition

Axioms in Protégé?



5 axioms:

hasBase **SubPropertyOf:** hasIngredient
Domain: Pizza
Range: PizzaBase
Characteristics: Functional, InverseFunctional

The University of Manchester

MANCHESTER 1824

OWL Syntax: ontology

OWL doesn't make this TBox/ABox distinction, but Protégé & DL do and I like it

- An **OWL ontology** is a collection of axioms,
  - which is the **imports closure** of an OWL document
  - which is in one of the OWL syntaxes https://www.w3.org/TR/owl2-syntax/

- An OWL **axiom** takes one of the following forms:
  - Class Frame (see above)
  - C SubClassOf: D (subclass)
  - C EquivalentTo: D (class equivalence)
  - R SubPropertyOf: S (subproperty)
  - R EquivalentTo: S (property equivalence)
  - ...
  - x Type: C (class instantiation)
  - x R y (property instantiation)

TBox

ABox

- where
  - C, D are class **expressions**
  - R is a property **expression**

built using OWL's constructors (see above)

MANCHESTER 1824

The University of Manchester

OWL - semantics & reasoning

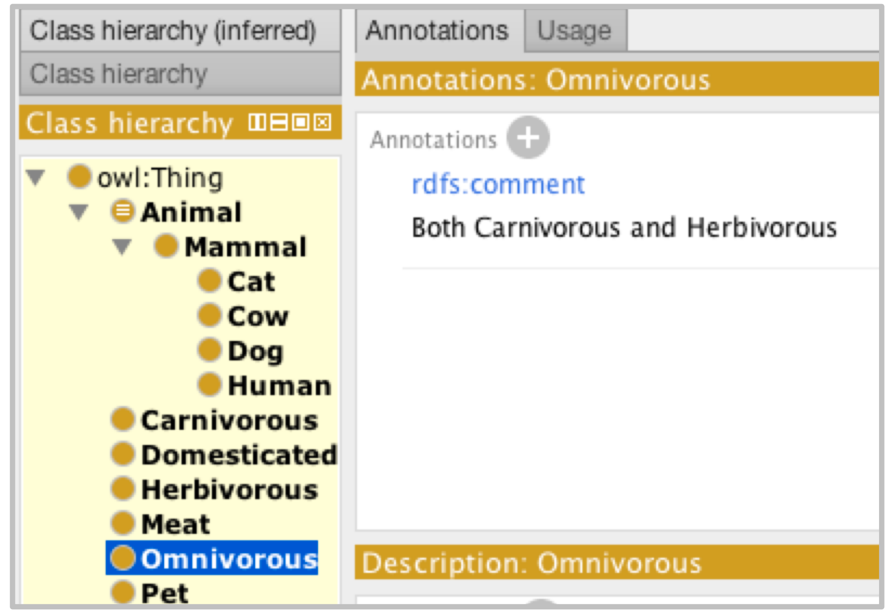
Reasoning…

- [here] is the process of determining **logical consequences** from a set of assumptions/statements/axioms
- sometimes is used more broadly
  - as the process of thinking about something in a logical way
- aka as
  - making inferences or
  - inferring (please, not inferencing)
- requires **semantic** so that we can agree on what **are** the consequences of (any set of) axioms
- but first: why reasoning?

Exploring Benefits of Axioms

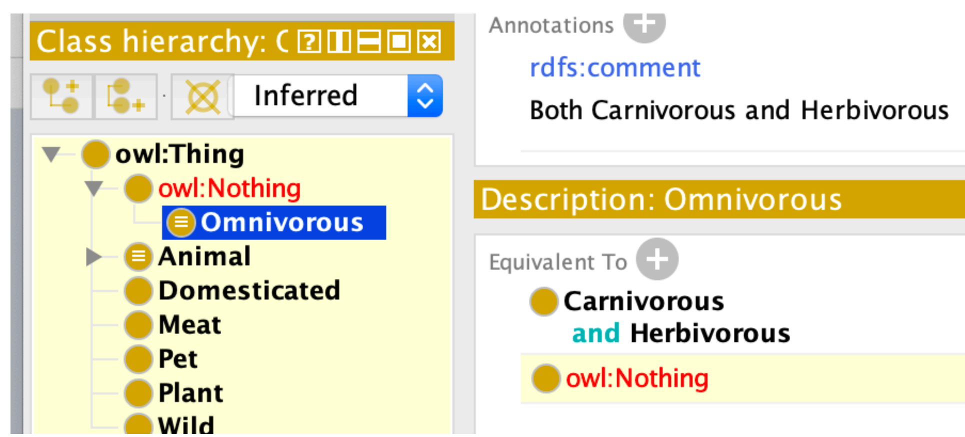
E.g., Omnivorous

- Annotations: comment "Carnivorous and Herbivorous" has no meaning



- Add **definition** in class description
  - run **reasoner**
  - check **inferred** class hierarchy

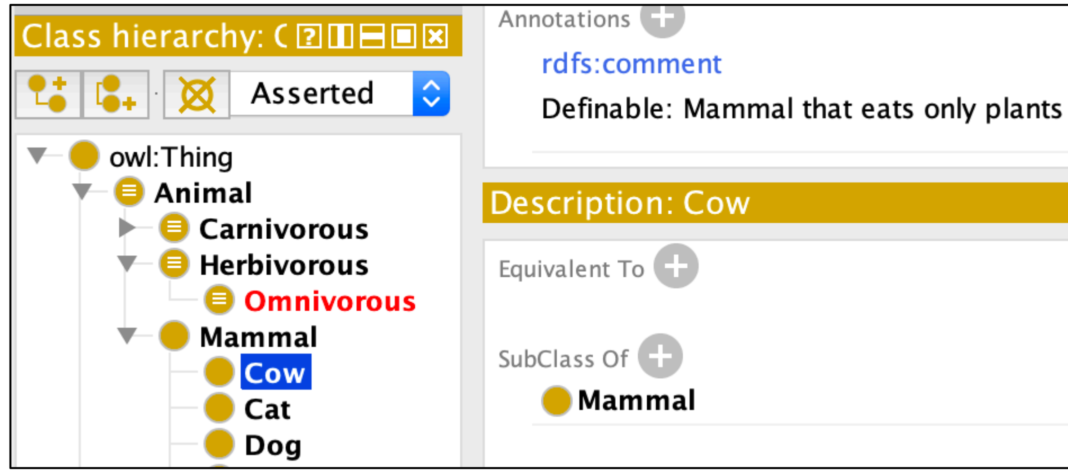
➡ our definition was wrong!



Exploring Benefits of Axioms II

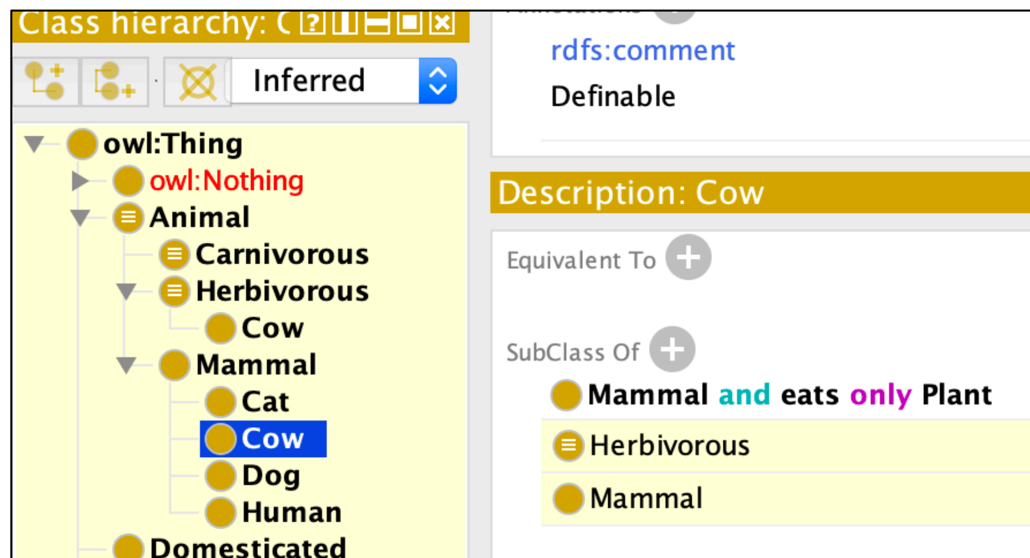
E.g., Cows

- Annotations: comment “Animal that eats only Plants” has no meaning



- Add definition in class description
  - run **reasoner**
  - check **inferred** class hierarchy

➡ our class hierarchy is **improved**: Cows are indeed herbivores!



First Benefits of Axioms & Reasoner

- Links/Sub-Super-class relations/Taxonomy for "free"
  - Tools make **implicit** links **explicit**
  - We don't have to encode every link ourselves
  - Different modality
    - Instead of is-a/subsumption relations...focus on **meanings**
    - …we can think **local** rather than **global**

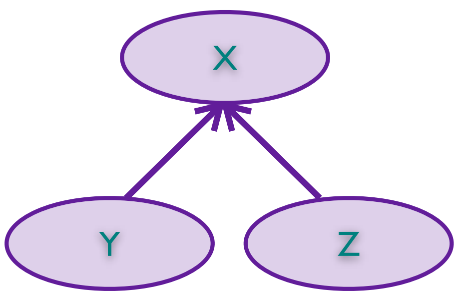
Meaning of term

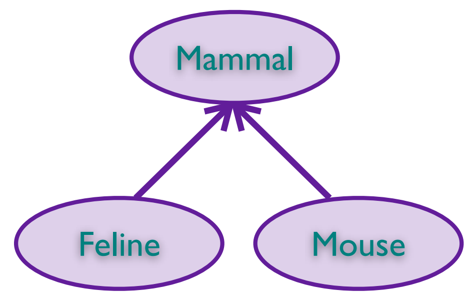
Place in Class Hierarchy/ Taxonomy

- Verification
  - Definitions have **consequences**
    - May change links:
      - wrong definitions may cause wrong links
      - links can be so wrong they are **obviously** wrong

Why Semantics? Isn't meaning obvious?

- **Semantics** of a language says **precisely** how to interpret a complex expression.
- Well defined semantics is **vital** to support machine interpretability
  - it removes ambiguities in the interpretation of the descriptions
  - i.e., all **tools** agree on their behaviour/give the same results & answers
  - …semantics acts as partial *specification* for tool developers





Is every Y and X (or only most/normally)?
Can a Y be a Z?
Can there be an X that's neither a Y nor a Z?
…

OWL 2 Semantics

- …here we concentrate on "Direct Semantics", "semantics" for short
- Is defined in terms of an **interpretation**
  - like in First Order Logic
- and comes in 3 stages:
  1. what do **classes/properties/individuals** stand for
     a. for names
     b. for expressions
  2. what does it mean for an interpretation to **satisfy** an
     - axiom
     - ontology
  3. what does it mean for an
     - ontology to **entail** an axiom
     - ontology to be **consistent**
     - ontology to be **coherent**
     - …or what is the inferred class hierarchy

OWL 2 Semantics: an interpretation (1a)

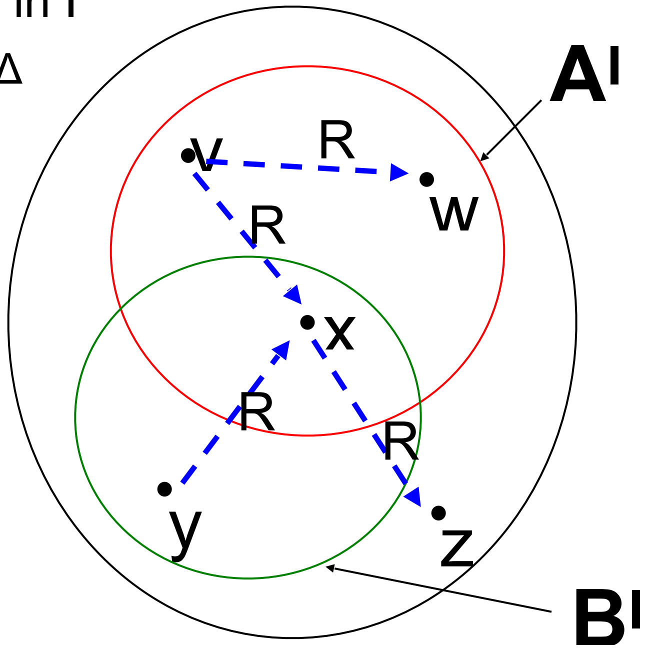
Like in FOL!

- An ***interpretation*** is a pair <Δ, I>, where
  - Δ is the ***domain*** (a non-empty set)
  - I is an ***interpretation function*** that maps each
    - **class name** A to a set $A^I \subseteq \Delta$

      ...we call A^I the ***extension*** of A in I
    - **property name** R to a binary relation $R^I \subseteq \Delta \times \Delta$

      ...if $(e,f) \in R^I$ we call f an R-***filler*** of e in I
    - **individual name** i to an element $i^I \in \Delta$

      ...if $i^I \in A^I$ we say that i is an ***instance of*** A in I
- ...and we can draw interpretations!
  - $\Delta = \{v, w, x, y, z\}$
  - $A^I = \{v, w, x\}$
  - $B^I = \{x, y\}$
  - $C^I = \{w, y\}$
  - $R^I = \{(v, w), (v, x), (y, x), (x, z)\}$



OWL 2 Semantics: an interpretation (1a)

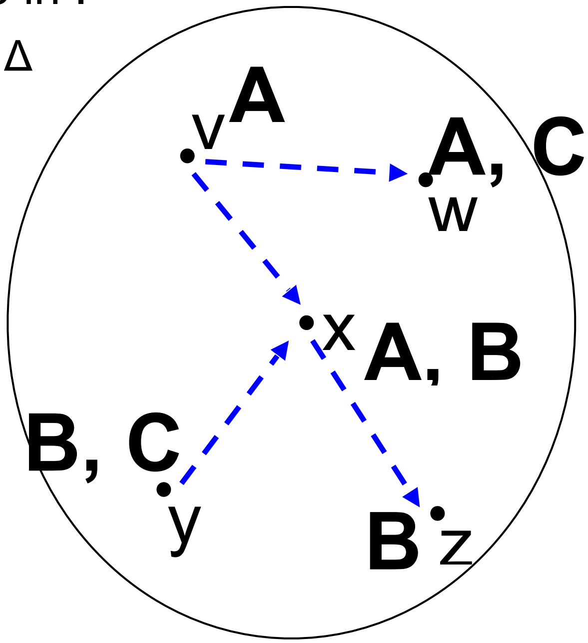
- An ***interpretation*** is a pair <Δ, I>, where
  - Δ is the ***domain*** (a non-empty set)
  - I is an ***interpretation function*** that maps each
    - **class name** A to a set $A^I \subseteq \Delta$

      ...we call A^I the ***extension*** of A in I
    - **property name** R to a binary relation $R^I \subseteq \Delta \times \Delta$

      ...if $(e,f) \in R^I$ we call f an R-***filler*** of e in I
    - **individual name** i to an element $i^I \in \Delta$

      ...if $i^I \in A^I$ we say that i is an

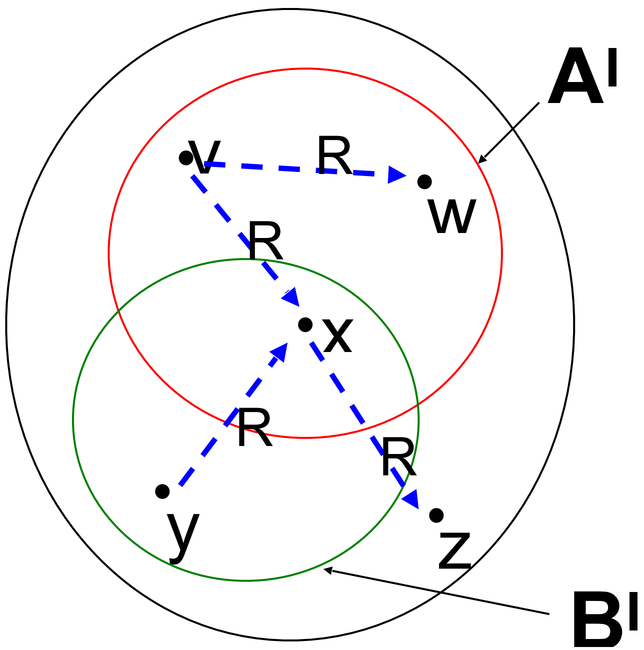
      instance of A in I
- ...and we can draw interpretations!
  - $\Delta = \{v, w, x, y, z\}$
  - $A^I = \{v, w, x\}$
  - $B^I = \{x, y\}$
  - $C^I = \{w, y\}$
  - $R^I = \{(v, w), (v, x), (y, x), (x, z)\}$

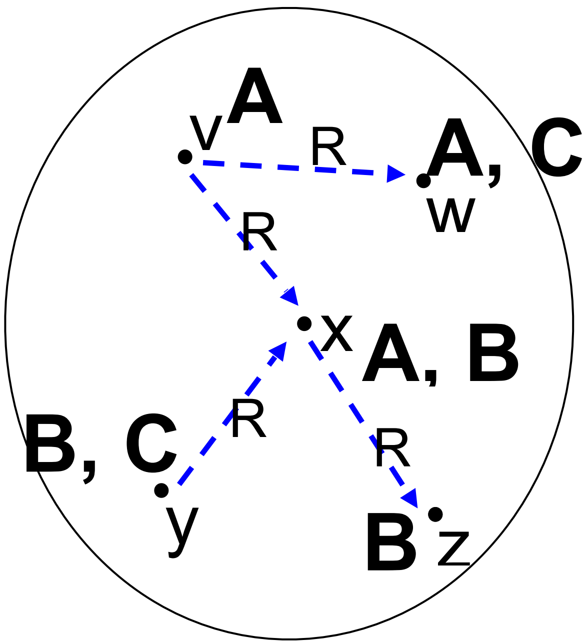


Interlude: drawing interpretations

- An interpretation is a pair <Δ, I>, where
  - Δ is the domain (a non-empty set)
  - I is an interpretation function that maps each
    - class name A to a set $A^I \subseteq \Delta$
    - property name R to a binary relation $R^I \subseteq \Delta \times \Delta$
    - individual name i to an element $i^I \in \Delta$

- We can draw interpretations
  - in 2 different ways
  - take your pick
  - but don't forget arrow heads!

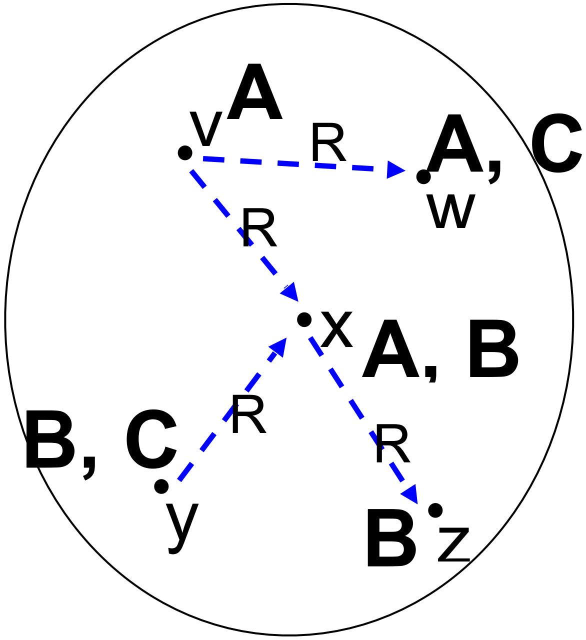




Interlude 2: Reading Definitions

- is really important for understanding
  - interpretations and hence
  - semantics of OWL
- make sure you understand that
  - you need **arrows** (not just lines)
  - possibly with **labels** for property names
  - what nodes and their labels mean
- check/re-read the definition:
  - what size can the domain have?
  - what size are extensions?
  - which restrictions are on them?
  - what's a really small interpretation?
  - what's a really big interpretation?

- An interpretation is a pair <Δ, I>, where
  - Δ is the domain (a non-empty set)
  - I is an interpretation function that maps each
    - class name A to a set $A^I \subseteq \Delta$
    - property name R to a binary relation $R^I \subseteq \Delta \times \Delta$
    - individual name i to an element $i^I \in \Delta$



OWL 2 Semantics: an interpretation (1b)

Interpretation of class expressions:

Constructor	**Example**	**Interpretation**
Class name	*Human*	$Human^I \subseteq \Delta$
Thing	n/a	Δ
Nothing	n/a	$\emptyset$
and	*Human* **and** *Male*	$Human^I \cap Male^I$
or	*Doctor* **or** *Lawyer*	$Doctor^I \cup Lawyer^I$
not	**not** *Male*	$\Delta \setminus Male^I$

OWL 2 Semantics: an interpretation (1b)

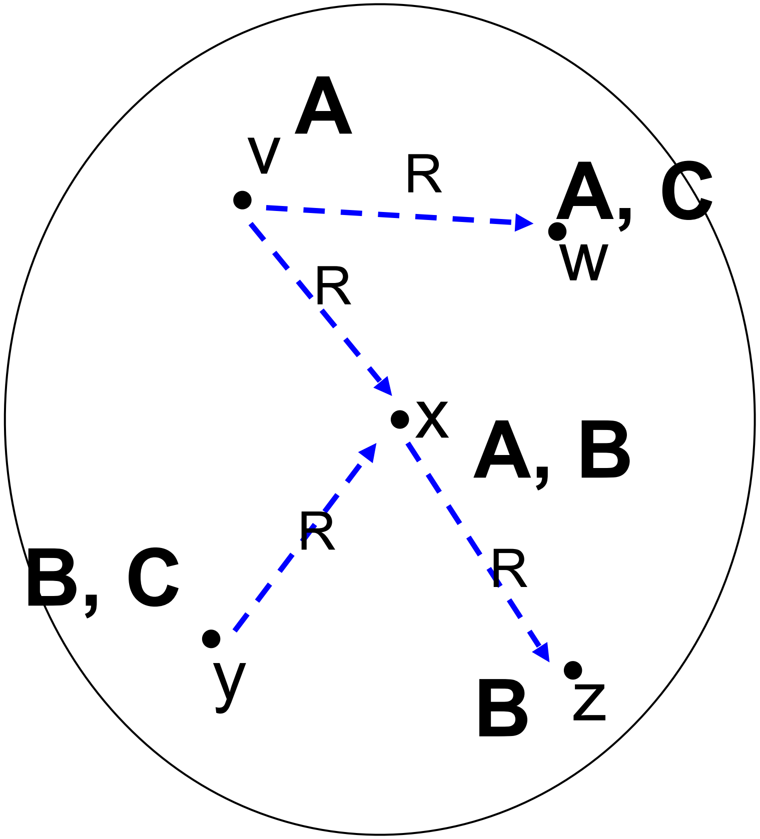
Interpretation of more class expressions:

Con-structor	Example	Interpretation
some	*hasChild* **some** *Lawyer*	{e ∈ Δ | there is some f: (e,f) ∈ $hasChild^I$ and f ∈ $Lawyer^I$}
only	*hasChild* **only** *Doctor*	{e ∈ Δ | for all f ∈ Δ: if (e,f) ∈ $hasChild^I$ then f ∈ $Doctor^I$}
min	*hasChild* **min** 2 *Tall*	{e ∈ Δ | there are at least 2 f ∈ Δ with (e,f) ∈ $hasChild^I$ and f ∈ $Tall^I$ }
max	*hasChild* **max** 2 *Tall*	{e ∈ Δ | there are at most 2 f ∈ Δ with (e,f) ∈ $hasChild^I$ and f ∈ $Tall^I$ }

Interpretation of Classes - Examples

- $\Delta = \{v, w, x, y, z\}$
- $A^I = \{v, w, x\}$
- $B^I = \{x, y\}$
- $R^I = \{(v, w), (v, x), (y, x), (x, z)\}$

- $(\text{not } B)^I =$
- $(A \text{ and } B)^I =$
- $((\text{not } A) \text{ or } B)^I =$
- $(R \text{ some } B)^I =$
- $(R \text{ only } B)^I =$
- $(R \text{ some } (R \text{ some } A))^I =$
- $(R \text{ some not}(A \text{ or } B))^I =$
- $(R \text{ min } 1.\text{Thing})^I =$
- $(R \text{ max } 1.\text{Thing})^I =$



OWL 2 Semantics: an interpretation satisfying … (2)

- An interpretation I **satisfies an axiom**
  - C **SubClassOf**: D if $C^I \subseteq D^I$
  - C **EquivalentTo**: D if $C^I = D^I$
  - P **SubPropertyOf**: S if $P^I \subseteq S^I$
  - P **EquivalentTo**: S if $P^I = S^I$
  - …
  - x **Type**: C if $x^I \in C^I$
  - x R y if $(x^I, y^I) \in R^I$

Check
OWL 2 Direct Semantics
for more!!!

- I **satisfies an ontology O** if I satisfies every axiom A in O
  - If I satisfies O, we call I a **model of** O

- See how the axioms in O *constrain* interpretations:
  - ✓ the more axioms you add to O, the fewer models O has
- …they do/don't hold/are(n't) satisfied in an ontology
  - in contrast, a class expression C **describes a set** C^I in I

OWL 2 is a decidable fragment of FOL

- For example,

Endocarditis *SubClassOf* Inflammation *and*
hasLoc *some* Endocardium
HeartDisease *EquivalentClass* Disease *and*
hasLoc *some* Heart

- is equivalent to

$$\forall x.\mathrm{Endocarditis}(x) \Rightarrow \mathrm{Inflammation}(x) \wedge \exists y.(\mathrm{hasLoc}(x,y) \wedge \mathrm{Endocardium}(y))$$

$$\forall x.\mathrm{HeartDisease}(x) \Leftrightarrow \mathrm{Disease}(x) \wedge \exists y.(\mathrm{hasLoc}(x,y) \wedge \mathrm{Heart}(y))$$

MANCHESTER 1824

The University of Manchester

Draw & Match Models to Ontologies!

O1 = {}

O2 = {a:C, b:D, c:C, d:C}

O3 = {a:C, b:D, c:C, b:C, d:E}

O4 = {a:C, b:D, c:C, b:C, d:E
D SubClassOf C}

O5 = {a:C, b:D, c:C, b:C, d:E
a R d,
D SubClassOf C,
D SubClassOf
S some C}

O6 = {a:C, b:D, c:C, b:C, d:E
a R d,
D SubClassOf C,
D SubClassOf
S some C,
C SubClassOf R only C }

I_1:
Δ = {v, w, x, y, z}

C^I = {v, w, y}
D^I = {x, y} E^I = {}

R^I = {(v, w), (v, y)}
S^I = {}

a^I = v b^I = x
c^I = w d^I = y

I_2:
Δ = {v, w, x, y, z}

C^I = {v, w, y}
D^I = {x, y} E^I = {y}

R^I = {(v, w), (v, y)}
S^I = {}

a^I = v b^I = x
c^I = w d^I = y

I_3:
Δ = {v, w, x, y, z}

C^I = {x, v, w, y}
D^I = {x, y} E^I = {y}

R^I = {(v, w), (v, y)}
S^I = {}

a^I = v b^I = x
c^I = w d^I = y

I_4:
Δ = {v, w, x, y, z}

C^I = {x, v, w, y}
D^I = {x, y} E^I = {y}

R^I = {(v, w), (v, y)}
S^I = {(x,x), (y,x)}

a^I = v b^I = x
c^I = w d^I = y

The world in an ontology: ontology as surrogate

Should agree with our view

World

Our view of our domain

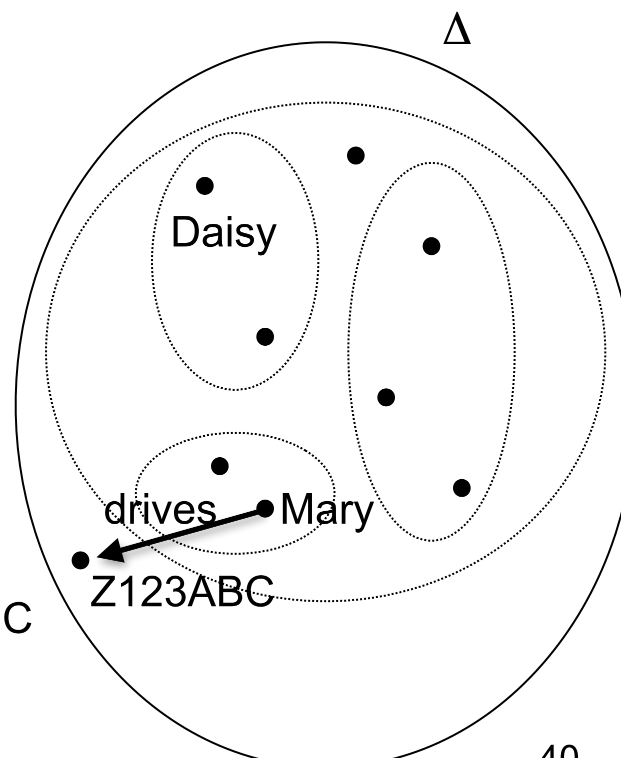
Ontology O

Daisy:Cow
Cow SubClassOf Animal

Mary: Person
Person SubClassOf Animal

Z123ABC: Car
Mary drives Z123ABC

Model of O



OWL 2 Semantics: Entailments etc. (3)

Let O be an ontology, α an axiom, and A, B classes, b an individual name:

- O is **consistent** if there exists some model I of O
  - i.e., there is an interpretation that satisfies all axioms in O
  - i.e., O isn't self contradictory
- O **entails** α (written O ⊨ α) if α is satisfied in all models of O
  - i.e., α is a consequence of the axioms in O
- A is **satisfiable** w.r.t. O if O ⊭ A SubClassOf Nothing
  - i.e., there is a model I of O with $A^I \neq \{\}$
- b is an **instance of** A w.r.t. O (written O ⊨ b:A) if $b^I \subseteq A^I$ in every model I of O

Theorem:

1. O is consistent iff O ⊭ Thing SubClassOf Nothing
2. A is satisfiable w.r.t. O iff O ∪ {n:A} is consistent (where n doesn't occur in O)
3. b is an instance of A in O iff O ∪ {b:not(A)} is not consistent
4. O entails A SubClassOf B iff O ∪ {n:A and not(B)} is inconsistent

OWL 2 Semantics: Entailments etc. (3) ctd

Let O be an ontology, α an axiom, and A, B classes, b an individual name:

- O is **coherent** if every class name that occurs in O is satisfiable w.r.t O

- **Classifying O** is a reasoning service consisting of
  1. testing whether O is consistent; if yes, then
  2. checking, for each pair A,B of class names in O plus Thing, Nothing
     O ⊧ A SubClassOf B
  3. checking, for each individual name b and class name A in O, whether O ⊧ b:A

  …and returning the result in a suitable form: O's **inferred class hierarchy**

OWL - tools & resources

OWL Reasoners and Protégé

- OWL **reasoners**
  - implement **decision procedures** for consistency/entailments, and classify ontologies
- **Protégé**
  - interacts with reasoners via the OWL API
  - shows results as
    - inferred class hierarchy where
    - unsatisfiable classes are red and you get a
    - warning (red triangle) if O is inconsistent
- OWL **reasoners**
  - implement highly optimised algorithms which decide
    - complex logical decision problems:
      - between PTime for OWL 2 EL profile to
      - N2ExpTime-hard for OWL 2…
    - via (hyper)-tableau algorithm or other
    - …later more

Complete details about OWL

- here, we have concentrated on some **core** features of OWL, e.g., no
  - domain, range axioms
  - SubPropertyOf, InverseOf
  - datatype properties
  - …
- we expect you to look these up!

- OWL is defined via a **Structural Specification**
- http://www.w3.org/TR/owl2-syntax/
- Defines language independently of concrete syntaxes
- Conceptual structure and abstract syntax
  - UML diagrams and functional-style syntax used to define the language
  - Mappings to concrete syntaxes then given.
- The structural specification provides the foundation for implementations (e.g. OWL API as discussed later)

OWL Resources

- The OWL Technical Documentation is all available online from the W3C site.

  http://www.w3.org/TR/owl2-overview/

  All the OWL documents are relevant; we recommend in particular the
  - Overview
  - Primer
  - Reference Guide and
  - Manchester Syntax Guide

- An introduction to OWL for people who know logic at
  http://owl.cs.manchester.ac.uk/about/orientation/a-logics-perspective/

- Our Ontogenesis Blog at
  http://www.sciencedirect.com/science/article/pii/S1570826808000413

Should agree with our view

World

Our view of our domain

Ontology O

Models of O

Chair SubClassOf
Furniture and
hasLeg at-least 3

Stool EquivalentTo
Chair and
not(hasPart some BackRest)

Z123: Chair
Z123 madeFrom W123
W123: Wood

I_1

I_2

....

Assumption: you are knowledge engineers, but not domain experts!