Requirements for Ontology Languages

Some hard requirements are as in software engineering:

- documentation / comments
- organisation of ontologies into separate modules / files

Others are central to capturing *shared conceptualisations*

- unique naming of entities
  - Cow in Ontology 1 should refer to the same concept as in Ontology 2
- multi-language support
  - Cow means “cow” in English, “Kuh” in German, “vache” in French

Again others are important for authorship and trust

- Who created/modified an ontology/axiom?
- When was an axiom modified?

OWL 2: Overview

OWL 2 is a standardized ontology language based on XML and RDF that was developed for the Semantic Web, and is used in many applications.

`https://www.w3.org/TR/owl2-overview/`

- **W**eb **O**ntology **L**anguage
- OWL 1 was specified in 2004 by the World Wide Web Consortium (W3C)
- OWL 2 was specified in 2009 by the W3C
- The final standard encompasses different sublanguages
  - more on this later

OWL 2 Terminology

OWL 2 uses different terminology than description logics:

DL	$\rightsquigarrow$	OWL
name	$\rightsquigarrow$	entity
concept name	$\rightsquigarrow$	(named) class
role name	$\rightsquigarrow$	(named) object property
individual name	$\rightsquigarrow$	(named) individual
concept (description)	$\rightsquigarrow$	class expression
axiom	$\rightsquigarrow$	axiom

Class Hierarchy

- Before adding complex axioms, first define the class hierarchy ($\sqsubseteq$-axioms).
- Flesh out the hierarchy with common superconcepts, missing siblings.
- Ideally, much of this information was already elicited, otherwise we have to ask the domain experts again.
- Once the class hierarchy is fixed, we can add definitions.

Organism

- Animal
  - Mammal
    - Cat
    - ...
  - Fish
    - Trout
    - ...
  - Carnivore
  - Herbivore
  - Omnivore
- Plant
  - Tree
  - Grass
  - Wheat

Definitions

Identify which terms should be defined:

- Depends on the goals of the ontology
- General terms like “Organism” probably don’t need a definition
- Some terms are easier to define than others, e.g. “Cat” vs. “Carnivore”.
- For many terms, the information about their place in the class hierarchy is enough.

Intensional definitions consist of the superclass(es) and any distinguishing characteristics.

> A cat is a mammal that has paws, legs, and a tail.
>
> A carnivore is an animal that eats only meat.
>
> A pet is a domesticated animal that lives with humans.

Definitions (II)

Distinguish between full definitions ($\equiv$) and partial definitions ($\sqsubseteq$)!

$$\mathit{Carnivore} \equiv \mathit{Animal} \sqcap \forall \mathit{eats}.\mathit{Meat}$$
$$\mathit{Herbivore} \equiv \mathit{Animal} \sqcap \forall \mathit{eats}.\mathit{Plant}$$
$$\mathit{Pet} \equiv \mathit{Animal} \sqcap \exists \mathit{livesWith}.\mathit{Human}$$
$$\mathit{Animal} \equiv \mathit{Organism} \sqcap \exists \mathit{eats}.\mathit{Organism}$$

$$\mathit{Cat} \sqsubseteq \mathit{Mammal} \sqcap \exists \mathit{bodyPart}.\mathit{Paw} \sqcap \exists \mathit{bodyPart}.\mathit{Leg} \sqcap \exists \mathit{bodyPart}.\mathit{Tail}$$
$$\mathit{Cow} \sqsubseteq \mathit{Mammal} \sqcap \forall \mathit{eats}.\mathit{Grass}$$

- Often, not everything can be fully defined, due to the restrictions of the ontology language.
- We will later see a more expressive DL, but this one will be restricted too.

Class Hierarchy (II)

In general, the class hierarchy is not a tree, but a directed acyclic graph with multiple inheritance.

$Cow \sqsubseteq Mammal \qquad Cow \sqsubseteq Herbivore$

($Mammal$ and $Herbivore$ are unrelated)

Instead of specifying all subclass-superclass relationships, it is easier to specify only a tree and let the reasoner infer the implicit ones.

$Grass \sqsubseteq Plant \qquad Mammal \sqsubseteq Animal$

$Herbivore \equiv Animal \sqcap \forall eats.Plant$

$Cow \sqsubseteq Mammal \sqcap \forall eats.Grass$

This entails $Cow \sqsubseteq Herbivore$, so we do not have to explicitly add this axiom to the ontology.

Note: "Some" Does Not Mean "Only"

When writing definitions, it is not trivial to find the right one.

A common modeling error is to swap $\forall$ and $\exists$:

$$Grass \sqsubseteq Plant$$
$$Herbivore \equiv Animal \sqcap \forall eats.Plant$$
$$Cow \sqsubseteq Mammal \sqcap \exists eats.Grass$$

Cow is not subsumed by Herbivore!

(A cow must eat "at least 1 Grass", but could eat other things.)

Note: "Only" Does Not Mean "Some"

$$Cow \sqsubseteq \forall eats.Grass$$

Cow is not subsumed by $\exists eats.Grass$, not even $\exists eats.\top$.

(A cow can eat only Grass, but does not have to eat anything.)

$$Animal \equiv Organism \sqcap \exists eats.Organism$$
$$Mammal \sqsubseteq Animal$$
$$Cow \sqsubseteq Mammal \sqcap \forall eats.Grass$$

entails $Cow \sqsubseteq \exists eats.Grass$.

Note: “And” Does Not Mean “Or”

“Cows eat grass and grain.”

$$Cow \sqsubseteq \forall eats.(Grass \sqcap Grain) \qquad Grass \sqsubseteq \neg Grain$$

Cow and $\exists eats.\top$ are disjoint!

(A cow can eat only things that are at the same time Grass and Grain, which do not exist.)

Better:

$$Cow \sqsubseteq \forall eats.(Grass \sqcup Grain) \qquad Grass \sqsubseteq \neg Grain$$

- Axioms like $Cow \sqsubseteq \forall eats.(Grass \sqcup Grain)$ are called closure axioms, because they define the range of $eats$ for all cows.

Closure Axioms

Often, closure axioms are used in combination with existential restrictions:

$$\mathit{Book} \sqsubseteq \exists \mathit{hasPart}.\mathit{Cover}$$
$$\mathit{Book} \sqsubseteq \exists \mathit{hasPart}.\mathit{Page}$$
$$\mathit{Book} \sqsubseteq \forall \mathit{hasPart}.(\mathit{Cover} \sqcup \mathit{Page})$$

This is like saying *"A book has those parts and no other."*

Covering Axioms

- Closure axioms are one way of *closing* descriptions under the open world assumption
- Another such technique is using covering axioms
- Covering axioms work well in combination with disjointness axioms

$$
\begin{aligned}
Herbivore &\equiv Animal \sqcap \forall eats.Plant \\
Carnivore &\equiv Animal \sqcap \forall eats.Animal \\
Animal &\equiv Herbivore \sqcup Carnivore \sqcup Omnivore \\
Animal &\sqsubseteq \neg Plant
\end{aligned}
$$

- These axioms entail:

$$Animal \sqcap \exists eats.Animal \sqcap \exists eats.Plant \sqsubseteq Omnivore$$

Disjointness Axioms

- Covering axioms work well in combination with disjointness axioms
- Disjointness axioms furthermore help finding bugs via incoherence
- OWL (and Protégé) offers convenient syntactic sugar:
  - Make a set of class expressions pair-wise disjoint
  - Define a class as disjoint union

Note: Value Restrictions on the Left-Hand Side

Value restrictions can behave strangely on the left-hand side of GCIs (and therefore also in full definitions).

$$\forall eats.Grass \sqsubseteq Cow$$

$$Cow \equiv \forall eats.Grass$$

With one of these axioms, anything that does not eat is a cow.

$$Tornado \sqsubseteq \neg Organism$$

$$\exists eats.\top \sqsubseteq Organism$$

$$\forall eats.Grass \sqsubseteq Cow$$

entail $Tornado \sqsubseteq \forall eats.\bot \sqsubseteq Cow$.