

Lecture 3: Ontologies and Description Logics

2-AIN-108 Computational Logic

Martin Baláž, Martin Homola

Department of Applied Informatics
Faculty of Mathematics, Physics and Informatics
Comenius University in Bratislava



9 Oct 2012

Ontology (Definition)

Definition (Philosophy)

Ontology a study of the nature of being, existence, or reality, as well as of basic categories of being and their relations.

Note: It is a philosophical study of entities that exist.

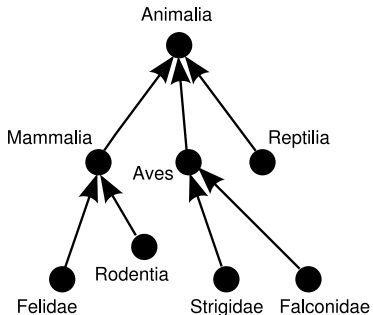
Definition (Computer Science)

Ontology is a formal conceptualization of a domain.

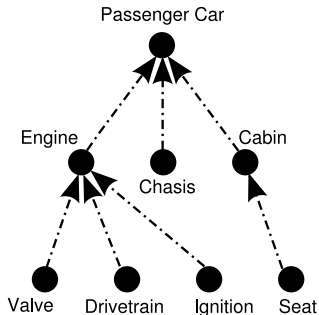
Note: It is a description of entities and their relations in a given domain, recorded in a formal language.

Note: In knowledge representation and computational logic we consider a formal language with logical semantics.

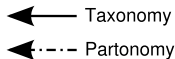
Example Ontologies



(a)



(b)



Is my ontology consistent? Are all concepts meaningful?
Is subsumption implied by the ontology?
Is a given object an instance of a given class?

Definition (Vocabulary)

A DL **vocabulary** consists of three mutually disjoint countably infinite sets:

- 1 set of **individuals** $N_I = \{a, b, \dots\}$;
- 2 set of **atomic concepts** $N_C = \{A, B, \dots\}$;
- 3 set of **roles** $N_R = \{R, S, \dots\}$.

Definition (Vocabulary)

A DL **vocabulary** consists of three mutually disjoint countably infinite sets:

- 1 set of **individuals** $N_I = \{a, b, \dots\}$;
- 2 set of **atomic concepts** $N_C = \{A, B, \dots\}$;
- 3 set of **roles** $N_R = \{R, S, \dots\}$.

Note: from now on, we always assume that some suitable vocabulary is given, containing all the symbols we use in our concepts and knowledge bases.

Definition (Complex concepts)

Complex concepts are recursively constructed as a smallest set of expressions of the forms:

$$C, D ::= A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C$$

where $A \in N_C$, $R \in N_R$, and C, D are complex concepts.

Definition (Complex concepts)

Complex concepts are recursively constructed as a smallest set of expressions of the forms:

$$C, D ::= A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C$$

where $A \in N_C$, $R \in N_R$, and C, D are complex concepts.

Note: A concept is any atomic or complex concept constructed on top of a given vocabulary.

Definition (Complex concepts)

Complex concepts are recursively constructed as a smallest set of expressions of the forms:

$$C, D ::= A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C$$

where $A \in N_C$, $R \in N_R$, and C, D are complex concepts.

Note: A concept is any atomic or complex concept constructed on top of a given vocabulary.

Note: Concept constructors of \mathcal{ALC} : complement (\neg), intersection (\sqcap), union (\sqcup), existential restriction (\exists), and value restriction (\forall).

Note: Other DL different from \mathcal{ALC} use different sets of constructors.

Definition (TBox)

A **TBox** \mathcal{T} is a finite set of GCI axioms ϕ of the form:

$$\phi ::= C \sqsubseteq D$$

where C, D are any concepts.

Definition (TBox)

A **TBox** \mathcal{T} is a finite set of GCI axioms ϕ of the form:

$$\phi ::= C \sqsubseteq D$$

where C, D are any concepts.

Definition (ABox)

A **ABox** \mathcal{T} is a finite set of assertion axioms ϕ of the form:

$$\phi ::= a : C \mid a, b : R$$

where $a, b \in N_I$, $R \in N_R$, and C is any concept.

Definition (TBox)

A **TBox** \mathcal{T} is a finite set of GCI axioms ϕ of the form:

$$\phi ::= C \sqsubseteq D$$

where C, D are any concepts.

Definition (ABox)

A **ABox** \mathcal{T} is a finite set of assertion axioms ϕ of the form:

$$\phi ::= a : C \mid a, b : R$$

where $a, b \in N_I, R \in N_R$, and C is any concept.

Note: GCI stands for General Concept Inclusions, they are general subsumption axioms. The two types of assertions are concept assertion and role assertion, respectively.

Definition (DL Knowledge Base)

A DL knowledge base (KB) $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ is a pair consisting of a TBox and an ABox.

Note: TBox contains the **intentional** part of the KB: the descriptions of all concepts and their relations. ABox contains the **extensional** part: empirical evidence, facts.

Note: Ontologies can be represented by DL KB. But ontologies can also be represented in other languages (including FOL).

Example (cont.)

\mathcal{T} :

$\text{Carnivore} \sqcup \text{Herbivore} \sqsubseteq \text{Animal}$
 $\text{Carnivore} \sqsubseteq \forall \text{eats} . (\text{Animal} \sqcup \text{AnimalPart})$
 $\text{Herbivore} \sqsubseteq \forall \text{eats} . \neg (\text{Animal} \sqcup \exists \text{partOf} . \text{Animal})$
 $\text{Cow} \sqsubseteq \text{Herbivore}$
 $\text{Brain} \sqsubseteq \exists \text{partOf} . \text{Animal}$
 $\text{CowBrain} \sqsubseteq \text{Brain} \sqcap \exists \text{partOf} . \text{Cow}$
 $\text{Plant} \sqsubseteq \neg \text{Animal}$
 $\text{Grass} \sqsubseteq \text{Plant}$

\mathcal{A} :

daisy : Cow
g3457 : Grass
daisy, g3457 : eats

Definition (Interpretation)

An **interpretation** of a given a DL KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ which contains:

- a **domain** $\Delta^{\mathcal{I}} \neq \emptyset$;
- an **interpretation function** $\cdot^{\mathcal{I}}$ s.t.:
 - $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ for all $a \in N_I$;
 - $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ for all $A \in N_C$;
 - $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ for all $R \in N_R$;
- and for any C , D and R , the **interpretation of complex concepts** is recursively defined as follows:

$$\neg C^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}};$$

$$C \sqcap D^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$

$$C \sqcup D^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$$

$$\exists R.C^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : \langle x, y \rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$$

$$\forall R.C^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}} : \langle x, y \rangle \in R^{\mathcal{I}} \implies y \in C^{\mathcal{I}}\}$$

Definition (Satisfaction \models)

Given an axiom ϕ , an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ satisfies ϕ depending on its type:

$$C \sqsubseteq D: \mathcal{I} \models C \sqsubseteq D \text{ iff } C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$$

$$a : C: \mathcal{I} \models a : C \text{ iff } a^{\mathcal{I}} \in C^{\mathcal{I}}$$

$$a, b : R: \mathcal{I} \models a, b : R \text{ iff } \langle a^{\mathcal{I}}, b^{\mathcal{I}} \rangle \in R^{\mathcal{I}}$$

Definition (Satisfaction \models)

Given an axiom ϕ , an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ satisfies ϕ depending on its type:

$$C \sqsubseteq D: \mathcal{I} \models C \sqsubseteq D \text{ iff } C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$$

$$a : C: \mathcal{I} \models a : C \text{ iff } a^{\mathcal{I}} \in C^{\mathcal{I}}$$

$$a, b : R: \mathcal{I} \models a, b : R \text{ iff } \langle a^{\mathcal{I}}, b^{\mathcal{I}} \rangle \in R^{\mathcal{I}}$$

Definition (Model)

An interpretation $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ is a **model** of a DL KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ if it satisfies every axiom in \mathcal{T} and \mathcal{A} .

Definition (Decision Problems)

Given a DL KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, and two concepts C, D , we say that:

- C is **satisfiable** w.r.t. \mathcal{K} iff there is a model \mathcal{I} of \mathcal{K} s.t. $C^{\mathcal{I}} \neq \emptyset$;
- C is **subsumed** by D w.r.t. \mathcal{K} (denoted $\mathcal{K} \models C \sqsubseteq D$) iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ in every model \mathcal{I} of \mathcal{K} ;
- C and D are **equivalent** w.r.t. \mathcal{K} (denoted $\mathcal{K} \models C \equiv D$) iff $C^{\mathcal{I}} = D^{\mathcal{I}}$ in every model \mathcal{I} of \mathcal{K} ;
- C and D are **disjoint** w.r.t. \mathcal{K} iff $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$ in every model \mathcal{I} of \mathcal{K} .

Definition (Decision Problems)

Given a DL KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, and two concepts C, D , we say that:

- C is **satisfiable** w.r.t. \mathcal{K} iff there is a model \mathcal{I} of \mathcal{K} s.t. $C^{\mathcal{I}} \neq \emptyset$;
- C is **subsumed** by D w.r.t. \mathcal{K} (denoted $\mathcal{K} \models C \sqsubseteq D$) iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ in every model \mathcal{I} of \mathcal{K} ;
- C and D are **equivalent** w.r.t. \mathcal{K} (denoted $\mathcal{K} \models C \equiv D$) iff $C^{\mathcal{I}} = D^{\mathcal{I}}$ in every model \mathcal{I} of \mathcal{K} ;
- C and D are **disjoint** w.r.t. \mathcal{K} iff $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$ in every model \mathcal{I} of \mathcal{K} .

Note: If \mathcal{K} is empty, then satisfiability, subsumption, equivalence, and disjointness of concepts is defined in general by the definition. In such a case we omit “ $\mathcal{K} \models$ ” from the notation.

$\text{Carnivore} \sqcup \text{Herbivore} \sqsubseteq \text{Animal}$

$\text{Carnivore} \sqsubseteq \forall \text{eats} . (\text{Animal} \sqcup \text{AnimalPart})$

$\text{Herbivore} \sqsubseteq \forall \text{eats} . \neg (\text{Animal} \sqcup \exists \text{partOf} . \text{Animal})$

$\text{Cow} \sqsubseteq \text{Herbivore}$

$\text{Brain} \sqsubseteq \exists \text{partOf} . \text{Animal}$

$\text{CowBrain} \sqsubseteq \text{Brain} \sqcap \exists \text{partOf} . \text{Cow}$

$\text{Carnivore} \sqcup \text{Herbivore} \sqsubseteq \text{Animal}$

$\text{Carnivore} \sqsubseteq \forall \text{eats} . (\text{Animal} \sqcup \text{AnimalPart})$

$\text{Herbivore} \sqsubseteq \forall \text{eats} . \neg (\text{Animal} \sqcup \exists \text{partOf} . \text{Animal})$

$\text{Cow} \sqsubseteq \text{Herbivore}$

$\text{Brain} \sqsubseteq \exists \text{partOf} . \text{Animal}$

$\text{CowBrain} \sqsubseteq \text{Brain} \sqcap \exists \text{partOf} . \text{Cow}$

$\text{MadCow} \sqsubseteq \text{Cow} \sqcap \exists \text{eats} . \text{CowBrain}$

Let us introduce some syntactic sugar:

Definition (Top and bottom concepts)

The top (\top) and bottom (\perp) concepts are defined as syntactic shorthands:

- \top is a placeholder for $A \sqcup \neg A$;
- \perp is a placeholder for $A \sqcap \neg A$;

where A is a new atomic concept not appearing elsewhere in the given KB or a any given concept.

Lemma (Top and bottom semantics)

In any interpretation \mathcal{I} , $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$ and $\perp^{\mathcal{I}} = \emptyset$.

Basic Decision Problems (cont.)

Reduction lemmata:

Lemma

Given a DL KB \mathcal{K} and a concept C : C is satisfiable w.r.t. \mathcal{K} iff $\mathcal{K} \not\models C \sqsubseteq \perp$.

Basic Decision Problems (cont.)

Reduction lemmata:

Lemma

Given a DL KB \mathcal{K} and a concept C : C is satisfiable w.r.t. \mathcal{K} iff $\mathcal{K} \not\models C \sqsubseteq \perp$.

Lemma

*Given a DL KB \mathcal{K} and concepts C, D : $\mathcal{K} \models C \sqsubseteq D$ iff $C \sqcap \neg D$ is *unsatisfiable w.r.t. \mathcal{K}* .*

Basic Decision Problems (cont.)

Reduction lemmata:

Lemma

Given a DL KB \mathcal{K} and a concept C : C is satisfiable w.r.t. \mathcal{K} iff $\mathcal{K} \not\models C \sqsubseteq \perp$.

Lemma

Given a DL KB \mathcal{K} and concepts C, D : $\mathcal{K} \models C \sqsubseteq D$ iff $C \sqcap \neg D$ is unsatisfiable w.r.t. \mathcal{K} .

Lemma

Given a DL KB \mathcal{K} and concepts C, D : $\mathcal{K} \models C \equiv D$ iff both $\mathcal{K} \models C \sqsubseteq D$ and $\mathcal{K} \models D \sqsubseteq C$.

Basic Decision Problems (cont.)

Reduction lemmata:

Lemma

Given a DL KB \mathcal{K} and a concept C : C is satisfiable w.r.t. \mathcal{K} iff $\mathcal{K} \not\models C \sqsubseteq \perp$.

Lemma

Given a DL KB \mathcal{K} and concepts C, D : $\mathcal{K} \models C \sqsubseteq D$ iff $C \sqcap \neg D$ is unsatisfiable w.r.t. \mathcal{K} .

Lemma

Given a DL KB \mathcal{K} and concepts C, D : $\mathcal{K} \models C \equiv D$ iff both $\mathcal{K} \models C \sqsubseteq D$ and $\mathcal{K} \models D \sqsubseteq C$.

Lemma

Given a DL KB \mathcal{K} and concepts C, D : C and D are disjoint w.r.t. \mathcal{K} iff $C \sqcap D$ is unsatisfiable w.r.t. \mathcal{K} .

Definition (ABox consistency)

A DL KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ is **consistent** (also, \mathcal{A} is consistent w.r.t. \mathcal{T}) iff it has at least one model.

Definition (Instance checking)

An individual a is an **instance** of a concept C w.r.t. a DL KB \mathcal{K} (denoted $\mathcal{K} \models a : C$) iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$ in all models \mathcal{I} of \mathcal{K} .

Some more reduction lemmata:

Lemma

Given a DL KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, an individual a and a concept C :
 $\mathcal{K} \models a : C$ iff $\mathcal{K}' = (\mathcal{T}, \mathcal{A} \cup \{a : \neg C\})$ is inconsistent.

Some more reduction lemmata:

Lemma

Given a DL KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, an individual a and a concept C :
 $\mathcal{K} \models a : C$ iff $\mathcal{K}' = (\mathcal{T}, \mathcal{A} \cup \{a : \neg C\})$ is inconsistent.

Lemma

Given a DL KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, and some concept C : *C is satisfiable w.r.t. \mathcal{K} iff $\mathcal{K}' = (\mathcal{T}, \mathcal{A} \cup \{a : C\})$ is consistent*, for some new individual a not appearing in \mathcal{K} .

Example (cont.)

\mathcal{T} :

$\text{Carnivore} \sqcup \text{Herbivore} \sqsubseteq \text{Animal}$

$\text{Carnivore} \sqsubseteq \forall \text{eats} . (\text{Animal} \sqcup \text{AnimalPart})$

$\text{Herbivore} \sqsubseteq \forall \text{eats} . \neg(\text{Animal} \sqcup \exists \text{partOf} . \text{Animal})$

$\text{Cow} \sqsubseteq \text{Herbivore}$

$\text{Brain} \sqsubseteq \exists \text{partOf} . \text{Animal}$

$\text{CowBrain} \sqsubseteq \text{Brain} \sqcap \exists \text{partOf} . \text{Cow}$

$\text{Grass} \sqsubseteq \text{Plant}$

$\text{DaisyFlower} \sqsubseteq \text{Plant}$

$\text{DaisyFlower} \sqsubseteq \neg \text{Grass}$

\mathcal{A} :

daisy : Cow

g3457 : Grass

daisy : $\forall \text{eats} . \text{DaisyFlower}$

daisy, g3457 : eats