Lecture 4: 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

October 16, 2013

Ontology (Definition)

Definition (Philosophy)

Ontology is 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.

Ontology (Definition)

Definition (Philosophy)

Ontology is 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.

Ontology (Definition)

Definition (Philosophy)

Ontology is 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.

Ontology (Definition)

Definition (Philosophy)

Ontology is 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.

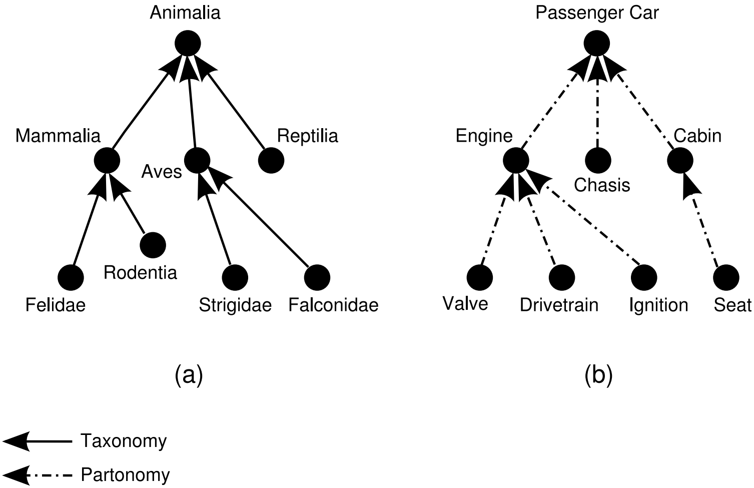
Definition (Computer Science)

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



Computational Problems

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

Definition (Vocabulary)

A DL vocabulary consists of three countable mutually disjoint sets:

1. set of individuals $N_{\mathsf{I}} = \{a, b, \dots\}$;
2. set of atomic concepts $N_{\mathsf{C}} = \{A, B, \dots\}$;
3. set of roles $N_{\mathsf{R}} = \{R, S, \dots\}$.

Definition (Vocabulary)

A DL vocabulary consists of three countable mutually disjoint sets:

1. set of individuals $N_{\mathsf{I}} = \{a, b, \dots\}$;
2. set of atomic concepts $N_{\mathsf{C}} = \{A, B, \dots\}$;
3. set of roles $N_{\mathsf{R}} = \{R, S, \dots\}$.

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.

$\mathcal{ALC}$ DL: Syntax (cont.)

Definition (Complex concepts)

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 concepts.

$\mathcal{ALC}$ DL: Syntax (cont.)

Definition (Complex concepts)

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 $A \in N_{\mathrm{C}}$, $R \in N_{\mathrm{R}}$, and C, D are concepts.

Note: Non-atomic concepts are often called complex concepts.

$\mathcal{ALC}$ DL: Syntax (cont.)

Definition (Complex concepts)

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 $A \in N_{\mathrm{C}}$, $R \in N_{\mathrm{R}}$, and C, D are concepts.

Note: Non-atomic concepts are often called complex concepts.

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.

$\mathcal{ALC}$ DL: Syntax (cont.)

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.

$\mathcal{ALC}$ DL: Syntax (cont.)

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 $a, b \in N_{\mathsf{I}}$, $R \in N_{\mathsf{R}}$, and C is any concept.

$\mathcal{ALC}$ DL: Syntax (cont.)

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 $a, b \in N_{\mathrm{I}}$, $R \in N_{\mathrm{R}}$, 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.

$\mathcal{ALC}$ DL: Syntax (cont.)

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.)

$\mathcal{T}$:

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

$\mathcal{A}$:

$$\mathsf{daisy} : \mathsf{Cow}$$
$$\mathsf{g3457} : \mathsf{Grass}$$
$$\mathsf{daisy}, \mathsf{g3457} : \mathsf{eats}$$

$\mathcal{ALC}$ DL: Semantics

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 $a \in N_{\mathrm{I}}$;
  $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ for all $A \in N_{\mathrm{C}}$;
  $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ for all $R \in N_{\mathrm{R}}$;
- and for any C, D and R, the interpretation of complex concepts is recursively defined as follows:
  $\neg C^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$;
  $C \sqcap D^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
  $C \sqcup D^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$
  $\exists R.C^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : \langle x, y\rangle \in R^{\mathcal{I}} \land y \in C^{\mathcal{I}}\}$
  $\forall R.C^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}} : \langle x, y\rangle \in R^{\mathcal{I}} \implies y \in C^{\mathcal{I}}\}$

$\mathcal{ALC}$ DL: Semantics

Definition (Satisfaction $\models$)

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

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

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

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

$\mathcal{ALC}$ DL: Semantics

Definition (Satisfaction $\models$)

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

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

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

$a, b : R$: $\mathcal{I} \models a, b : R$ 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 $\mathcal{A}$.

Basic Decision Problems

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 $\mathcal{K}$.

Basic Decision Problems

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 are defined in general by the definition. In such a case we omit "$\mathcal{K} \models$" from the notation.

Example (cont.)

$$\mathsf{Carnivore} \sqcup \mathsf{Herbivore} \sqsubseteq \mathsf{Animal}$$
$$\mathsf{Carnivore} \sqsubseteq \forall \mathsf{eats}.(\mathsf{Animal} \sqcup \mathsf{AnimalPart})$$
$$\mathsf{Herbivore} \sqsubseteq \forall \mathsf{eats}.\neg(\mathsf{Animal} \sqcup \exists \mathsf{partOf}.\mathsf{Animal})$$
$$\mathsf{Cow} \sqsubseteq \mathsf{Hebivore}$$
$$\mathsf{Brain} \sqsubseteq \exists \mathsf{partOf}.\mathsf{Aminal}$$
$$\mathsf{CowBrain} \sqsubseteq \mathsf{Brain} \sqcap \exists \mathsf{partOf}.\mathsf{Cow}$$

Example (cont.)

$$\mathsf{Carnivore} \sqcup \mathsf{Herbivore} \sqsubseteq \mathsf{Animal}$$
$$\mathsf{Carnivore} \sqsubseteq \forall \mathsf{eats}.(\mathsf{Animal} \sqcup \mathsf{AnimalPart})$$
$$\mathsf{Herbivore} \sqsubseteq \forall \mathsf{eats}.\neg(\mathsf{Animal} \sqcup \exists \mathsf{partOf}.\mathsf{Animal})$$
$$\mathsf{Cow} \sqsubseteq \mathsf{Hebivore}$$
$$\mathsf{Brain} \sqsubseteq \exists \mathsf{partOf}.\mathsf{Aminal}$$
$$\mathsf{CowBrain} \sqsubseteq \mathsf{Brain} \sqcap \exists \mathsf{partOf}.\mathsf{Cow}$$
$$\mathsf{MadCow} \sqsubseteq \mathsf{Cow} \sqcap \exists \mathsf{eats}.\mathsf{CowBrain}$$

$\mathcal{ALC}$ DL: Syntax (cont.)

Let us introduce some syntactic sugar:

Definition (Top and bottom concepts)

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

- $\top$ is a placeholder for $A \sqcup \neg A$;
- $\bot$ 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.

$\mathcal{ALC}$ DL: Syntax (cont.)

Let us introduce some syntactic sugar:

Definition (Top and bottom concepts)

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

- $\top$ is a placeholder for $A \sqcup \neg A$;
- $\bot$ 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* $\bot^{\mathcal{I}} = \emptyset$.

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 $\mathcal{K} \not\models C \sqsubseteq \bot$.

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 $\mathcal{K} \not\models C \sqsubseteq \bot$.

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 $\mathcal{K} \not\models C \sqsubseteq \bot$.

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 $\mathcal{K} \not\models C \sqsubseteq \bot$.

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. $\mathcal{K}$.

Decision Problems for ABoxes

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 $\mathcal{I}$ if $\mathcal{K}$.

Decision Problems for ABoxes (cont.)

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.)

$\mathcal{T}$:

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

$\mathcal{A}$:

$$\mathsf{daisy} : \mathsf{Cow} \qquad \mathsf{g3457} : \mathsf{Grass}$$
$$\mathsf{daisy} : \forall \mathsf{eats}.\mathsf{DaisyFlower} \qquad \mathsf{daisy}, \mathsf{g3457} : \mathsf{eats}$$