Ontology Languages: Description Logics

Oscar Romero
Facultat d'Informàtica de Barcelona
Universitat Politècnica de Catalunya

Examples in this section are based on:

- D. Calvanese and D. Lembo (tutorial on DL @ISCW'07)
- F. Baader et al. The Description Logic Handbook

DESCRIPTION LOGICS

Logic-Based Ontology Languages

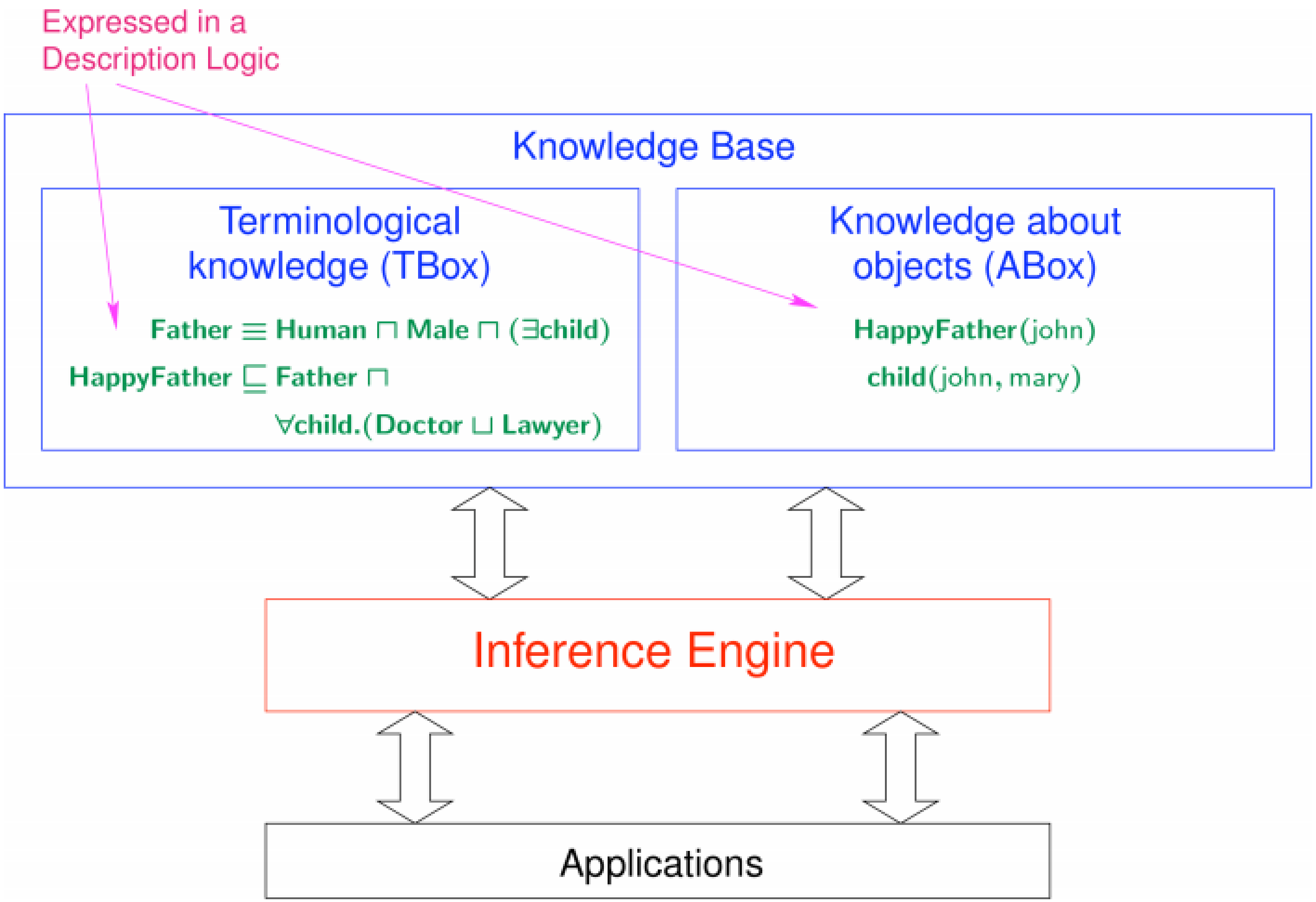
- FOL
  - Suitable for knowledge representation
    - Classes as unary predicates
    - Properties / relationships as binary predicates
    - Constraints as logical formulas using those predicates
  - Undecidability
    - In the general case, there is no algorithm that determines if a FOL formula implies another
- Decidable Fragments of FOL
  - Description Logics (binary predicates bounded number of variables)
  - Datalog (Horn-clauses)

Expressive Ontology Languages

Decidable subsets of FOL

	Datalog	Description Logics
Focus	Instances	Knowledge
Approach	Centralized	Decentralized
Reasoning	Closed-world assumption	Open-world assumption
Unique name	Unique name assumption	Non-unique name assumption

Description Logic KB



Description Logics: TBOX

- A DL TBOX is characterized by a set of constructs for building complex concepts and roles from atomic ones:
  - Concepts correspond to classes
  - Roles correspond to relationships
- It defines the terminology (of the domain)
- Formal semantics are given in terms of interpretations:

An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of:

- a nonempty set $\Delta^{\mathcal{I}}$, the domain of $\mathcal{I}$
- an interpretation function $\cdot^{\mathcal{I}}$, which maps
  - each individual a to an element $a^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$
  - each atomic concept A to a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$
  - each atomic role P to a subset $P^{\mathcal{I}}$ of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$

Concept Constructs

Construct	Syntax	Example	Semantics
atomic concept	A	Doctor	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
atomic role	P	hasChild	$P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
atomic negation	$\neg A$	¬Doctor	$\Delta^{\mathcal{I}} \setminus A^{\mathcal{I}}$
conjunction	$C \sqcap D$	Hum ⊓ Male	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
(unqual.) exist. res.	$\exists R$	∃hasChild	$\{\ a \mid \exists b.\,(a,b) \in R^{\mathcal{I}}\ \}$
value restriction	$\forall R.C$	∀hasChild.Male	$\{a \mid \forall b.\,(a,b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$
bottom	$\bot$		$\emptyset$

(C, D denote arbitrary concepts and R an arbitrary role)

The above constructs form the basic language $\mathcal{AL}$ of the family of $\mathcal{AL}$ languages.

Additional Concept and Role Constructs

Construct	$\mathcal{AL}\cdot$	Syntax	Semantics
disjunction	$\mathcal{U}$	$C \sqcup D$	$C^\mathcal{I} \cup D^\mathcal{I}$
top		$\top$	$\Delta^\mathcal{I}$
qual. exist. res.	$\mathcal{E}$	$\exists R.C$	$\{ a \mid \exists b. (a,b) \in R^\mathcal{I} \wedge b \in C^\mathcal{I} \}$
(full) negation	$\mathcal{C}$	$\neg C$	$\Delta^\mathcal{I} \setminus C^\mathcal{I}$
number	$\mathcal{N}$	$(\geq k\, R)$	$\{ a \mid \#\{b \mid (a,b) \in R^\mathcal{I}\} \geq k \}$
restrictions		$(\leq k\, R)$	$\{ a \mid \#\{b \mid (a,b) \in R^\mathcal{I}\} \leq k \}$
qual. number	$\mathcal{Q}$	$(\geq k\, R.C)$	$\{ a \mid \#\{b \mid (a,b) \in R^\mathcal{I} \wedge b \in C^\mathcal{I}\} \geq k \}$
restrictions		$(\leq k\, R.C)$	$\{ a \mid \#\{b \mid (a,b) \in R^\mathcal{I} \wedge b \in C^\mathcal{I}\} \leq k \}$
inverse role	$\mathcal{I}$	R^-	$\{ (a,b) \mid (b,a) \in R^\mathcal{I} \}$
role closure	$_{reg}$	$\mathcal{R}^*$	$(R^\mathcal{I})^*$

Further Examples of DL Constructs

- What is the meaning of these axioms?

$$\forall \mathsf{hasChild}.(\mathsf{Doctor} \sqcup \mathsf{Lawyer})$$

$$\exists \mathsf{hasChild}.\mathsf{Doctor}$$

$$\neg(\mathsf{Doctor} \sqcup \mathsf{Lawyer})$$

$$(\geq 2\,\mathsf{hasChild}) \sqcap (\leq 1\,\mathsf{sibling})$$

$$(\geq 2\,\mathsf{hasChild}.\,\mathsf{Doctor})$$

$$\forall \mathsf{hasChild}^{-}.\mathsf{Doctor}$$

$$\exists \mathsf{hasChild}^{*}.\mathsf{Doctor}$$

Formal Semantics

- A DL TBOX only includes terminological axioms of the following form
  - Inclusion $C_1 \sqsubseteq C_2$ is satisfied by $\mathcal{I}$ if $C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$
    $R_1 \sqsubseteq R_2$ is satisfied by $\mathcal{I}$ if $R_1^{\mathcal{I}} \subseteq R_2^{\mathcal{I}}$

    Example: $\text{PhDStudent} \sqsubseteq \text{Student} \sqcap \text{Researcher}$
  - Equivalence $C_1 \sqsubseteq C_2,\ C_2 \sqsubseteq C_1$

    Example: $\text{PhDStudent} \equiv \text{Student} \sqcap \text{Researcher}$

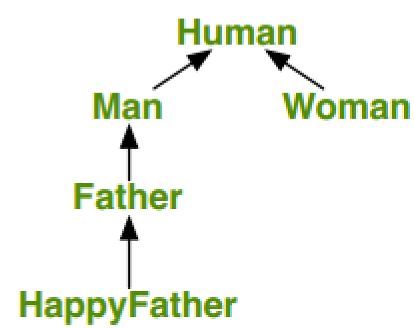
Reasoning on Concept Expressions

An interpretation $\mathcal{I}$ is a model of a concept C if $C^{\mathcal{I}} \neq \emptyset$.

Basic reasoning tasks:

1. Concept satisfiability: does C admit a model?
2. Concept subsumption $C \sqsubseteq D$: does $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ hold for all interpretations $\mathcal{I}$?

Subsumption used to build the concept hierarchy:



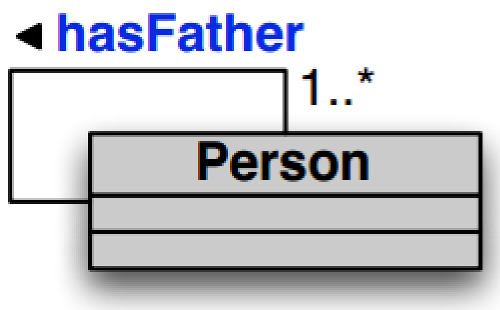
TBOX Example

- Example of TBOX:

$$
\begin{aligned}
\mathsf{Woman} &\equiv \mathsf{Person} \sqcap \mathsf{Female}\\
\mathsf{Man} &\equiv \mathsf{Person} \sqcap \neg\mathsf{Woman}\\
\mathsf{Mother} &\equiv \mathsf{Woman} \sqcap \exists\mathsf{hasChild}.\mathsf{Person}\\
\mathsf{Father} &\equiv \mathsf{Man} \sqcap \exists\mathsf{hasChild}.\mathsf{Person}\\
\mathsf{Parent} &\equiv \mathsf{Father} \sqcup \mathsf{Mother}\\
\mathsf{Grandmother} &\equiv \mathsf{Mother} \sqcap \exists\mathsf{hasChild}.\mathsf{Parent}\\
\mathsf{MotherWithManyChildren} &\equiv \mathsf{Mother} \sqcap \geqslant 3\ \mathsf{hasChild}\\
\mathsf{MotherWithoutDaughter} &\equiv \mathsf{Mother} \sqcap \forall\mathsf{hasChild}.\neg\mathsf{Woman}\\
\mathsf{Wife} &\equiv \mathsf{Woman} \sqcap \exists\mathsf{hasHusband}.\mathsf{Man}
\end{aligned}
$$

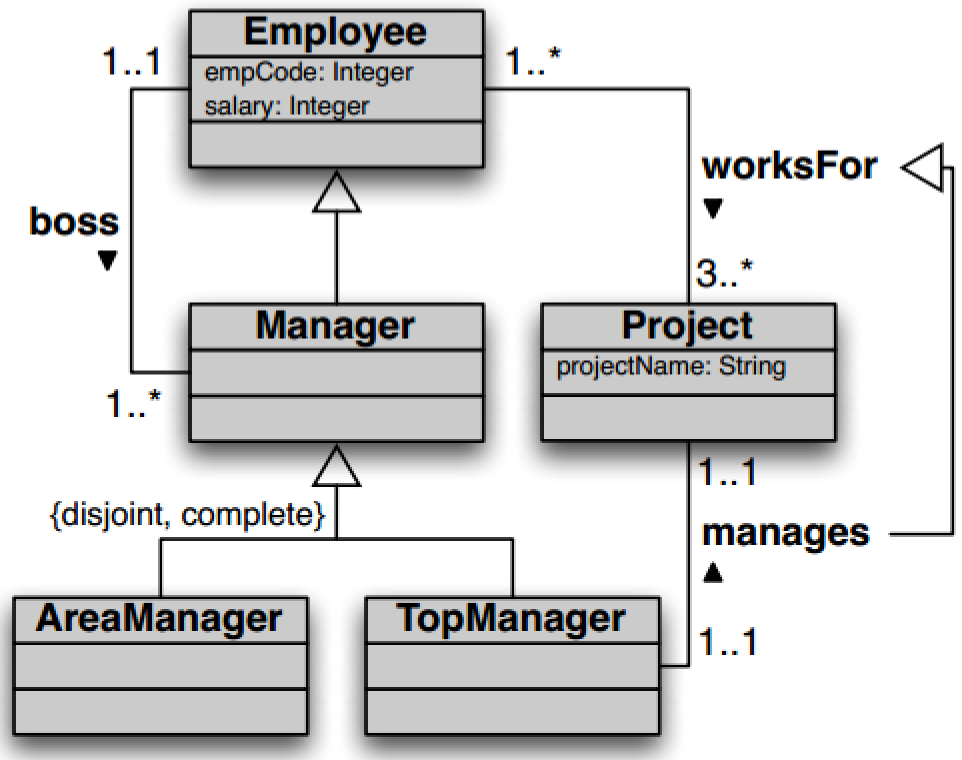
Exercise

- Represent as concept expressions the following UML diagram



Exercise II

- Represent as concept expressions the following UML diagram



Reasoning Complexity

Complexity of concept satisfiability: [DLNN97]

$\mathcal{AL}$, $\mathcal{ALN}$	PTIME
$\mathcal{ALU}$, $\mathcal{ALUN}$	NP-complete
$\mathcal{ALE}$	coNP-complete
$\mathcal{ALC}$, $\mathcal{ALCN}$, $\mathcal{ALCI}$, $\mathcal{ALCQI}$	PSPACE-complete

Observations:

- Two sources of complexity:
  - union ($\mathcal{U}$) of type NP,
  - existential quantification ($\mathcal{E}$) of type coNP.

  When they are combined, the complexity jumps to PSPACE.
- Number restrictions ($\mathcal{N}$) do not add to the complexity.

Description Logics: ABOX

- Defines instances in terms of the terminological axioms defined in the TBOX
  - Concept assertions
    - Student(Pere)
  - Role assertions
    - Teaches (Oscar, Pere)

Example of DL Knowledge Base

TBox assertions:

- Inclusion assertions on concepts:

$$\begin{array}{rcl} \mathsf{Father} & \equiv & \mathsf{Human} \sqcap \mathsf{Male} \sqcap \exists \mathsf{hasChild} \\ \mathsf{HappyFather} & \sqsubseteq & \mathsf{Father} \sqcap \forall \mathsf{hasChild}.(\mathsf{Doctor} \sqcup \mathsf{Lawyer} \sqcup \mathsf{HappyPerson}) \\ \mathsf{HappyAnc} & \sqsubseteq & \forall \mathsf{descendant}.\mathsf{HappyFather} \\ \mathsf{Teacher} & \sqsubseteq & \neg \mathsf{Doctor} \sqcap \neg \mathsf{Lawyer} \end{array}$$

- Inclusion assertions on roles:

$$\mathsf{hasChild} \sqsubseteq \mathsf{descendant} \qquad \mathsf{hasFather} \sqsubseteq \mathsf{hasChild}^-$$

ABox membership assertions:

- $\mathsf{Teacher}(\mathtt{mary}), \quad \mathsf{hasFather}(\mathtt{mary}, \mathtt{john}), \quad \mathsf{HappyAnc}(\mathtt{john})$

Models of a DL Ontology

Model of a DL knowledge base

An interpretation $\mathcal{I}$ is a model of $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ if it satisfies all assertions in $\mathcal{T}$ and all assertions in $\mathcal{A}$.

$\mathcal{O}$ is said to be satisfiable if it admits a model.

The fundamental reasoning service from which all other ones can be easily derived is . . .

Logical implication

$\mathcal{O}$ logically implies and assertion α, written $\mathcal{O} \models \alpha$, if α is satisfied by all models of $\mathcal{O}$.

TBOX Reasoning

- Concept Satisfiability: C is satisfiable wrt $\mathcal{T}$, if there is a model $\mathcal{I}$ of $\mathcal{T}$ such that $C^{\mathcal{I}}$ is not empty, i.e., $\mathcal{T} \not\models C \equiv \bot$.
- Subsumption: C_1 is subsumed by C_2 wrt $\mathcal{T}$, if for every model $\mathcal{I}$ of $\mathcal{T}$ we have $C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$, i.e., $\mathcal{T} \models C_1 \sqsubseteq C_2$.
- Equivalence: C_1 and C_2 are equivalent wrt $\mathcal{T}$ if for every model $\mathcal{I}$ of $\mathcal{T}$ we have $C_1^{\mathcal{I}} = C_2^{\mathcal{I}}$, i.e., $\mathcal{T} \models C_1 \equiv C_2$.
- Disjointness: C_1 and C_2 are disjoint wrt $\mathcal{T}$ if for every model $\mathcal{I}$ of $\mathcal{T}$ we have $C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}} = \emptyset$, i.e., $\mathcal{T} \models C_1 \sqcap C_2 \equiv \bot$.
- Functionality implication: A functionality assertion $(\mathbf{funct}\ R)$ is logically implied by $\mathcal{T}$ if for every model $\mathcal{I}$ of $\mathcal{T}$, we have that $(o, o_1) \in R^{\mathcal{I}}$ and $(o, o_2) \in R^{\mathcal{I}}$ implies $o_1 = o_2$, i.e., $\mathcal{T} \models (\mathbf{funct}\ R)$

Ontology Reasoning

- Ontology Satisfiability: Verify whether an ontology $\mathcal{O}$ is satisfiable, i.e., whether $\mathcal{O}$ admits at least one model.
- Concept Instance Checking: Verify whether an individual c is an instance of a concept C in $\mathcal{O}$, i.e., whether $\mathcal{O} \models C(c)$.
- Role Instance Checking: Verify whether a pair (c_1, c_2) of individuals is an instance of a role R in $\mathcal{O}$, i.e., whether $\mathcal{O} \models R(c_1, c_2)$.
- Query Answering:

The certain answers to $q(\vec{x})$ over $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, denoted $cert(q, \mathcal{O})$, ...

are the tuples $\vec{c}$ of constants of $\mathcal{A}$ such that $\vec{c} \in q^{\mathcal{I}}$, for every model $\mathcal{I}$ of $\mathcal{O}$.

Exercise

- TBOX:

$$Researcher \sqsubseteq \neg Professor$$
$$Researcher \sqsubseteq \neg Lecturer$$
$$\exists TeachesTo^{-} \sqsubseteq Student$$

$$Student \sqcap \neg Undergrad \sqsubseteq GraduateStudent$$
$$\exists TeachesTo.Undergrad \sqsubseteq Professor \sqcup Lecturer$$

TBOX Inferences:

- ABOX:

$$TeachesTo(dupond, pierre)$$
$$\neg GraduateStudent(pierre)$$
$$\neg Professor(dupond)$$

Ontology Inferences:

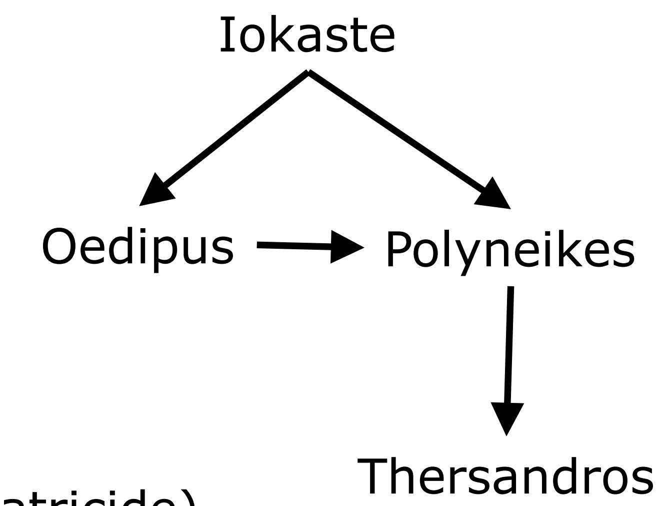
Modeling with Description Logics

- It is hard to build good ontologies with DL
  - The names of the classes are irrelevant
  - Classes are overlapping by default
  - Domain and range definitions are axioms, not constraints
  - Open world assumption
    - Anything might be true unless explicit asserted knowledge contradicts it (negation)
  - Non-unique name assumption
    - Although families such as the DL-Lite family assume the unique name assumption

Open-World Assumption

- Something evaluates false only if it contradicts other information in the ontology

hasSon(Iokaste,Oedipus)
hasSon(Iokaste,Polyneikes)
hasSon(Oedipus,Polyneikes)
hasSon(Polyneikes,Thersandros)
patricide(Oedipus)
¬patricide(Thersandros)



$Query \equiv \exists hasSon.(patricide \sqcap \exists hasSon.\neg patricide)$

$ABox \models Query(Iokaste)$?

Summary

- Description Logics
  - TBOX
    - Constructs
    - Formal Semantics
  - ABOX
  - Reasoning
    - Open-World Assumption