Master in Human Language Technology and Interfaces
Course on Languages Resources and Ontologies

Formal Ontology and Ontologies

1- A Logic Primer

Laure Vieu
Laboratory for Applied Ontology, ISTC-CNR & IRIT-CNRS, France
vieu@irit.fr

Trento, March 2011

Outline of this part of the course

- Lexical resources
- Logical bases of knowledge representation and ontologies, **today**
- Ontology and Ontologies (March 29)
- Ontologies and lexical resources; Methodological issues (April 5)
- Tools for ontology building (April 12)
- Annotation

1 Why logic in this course?

- Ontologies are knowledge systems, formally studied in Artificial Intelligence though logic, for both analyzing their expressivity and calculability
- Ontologies are standardly represented in OWL, a language backed by a Description Logic
- Ontologies are conceptually ground in “Formal Ontology” a branch of philosophy between metaphysics and analytic philosophy, using logic as a common language to express theories

2 Logic: Representation and Reasoning

Reliable tool to represent and to reason about explicit knowledge
Precise (no ambiguity) and **general** (not context dependent)
First-Order Logic is a standard in knowledge representation

- language
  - vocabulary: atomic symbols;
  - syntax: complex formulas;
  - semantics: truth;
  - true formulas *represent* facts.

- inference rules
  - *reasoning*: syntactically deduce new formulas (consequences) from given formulas (premises);
  - soundness and completeness: deduction preserves truth.

3 Outline of today's lesson

Focus on representation, not reasoning
Simple introduction to become familiar with the use of the language

- Propositional Logic
- First-order Logic (FOL) / Predicate Logic
- Overview of other logics, Introduction to Description Logics

4 Propositional Logic - Vocabulary

- *propositional letters*: $A, B, C, \dots$
  symbols for the *atomic propositions* of the language, i.e., simple statements
  e.g. A could mean "The weather is cold", B could mean "Michael eats an apple" etc.

- *connectives*:

$\neg$	*not*	negation
$\wedge$	*and*	conjunction
$\vee$	*or*	disjunction
$\rightarrow$	*if ... then*	material implication
$\leftrightarrow$	*if and only if*	bi-conditional (equivalence)

  ⋆ some other symbols in textbooks: $\sim$ for $\neg$, & for $\wedge$, $\supset$ for $\rightarrow$

- *parenthesis*: (,)

5 Propositional Logic - Syntax 1

- Using atomic propositions and connectives, we can build complex propositions:
  - "The weather is **not** cold" becomes: $\neg A$
  - "**If** the weather is cold **then** Michael eats an apple": $A \to B$
  - "The weather is cold **and** Michael eats an apple": $A \wedge B$
- In the same way, we can combine complex propositions using connectives to obtain even more complex propositions.

⋆ The set of all atomic and complex propositions is the set of *propositional formulas* or, simply, *propositions*.

6 Propositional Logic - Syntax 2

- Just some combinations of symbols (expressions) make sense.
  These expressions are called *well-formed formulas* (wffs).
- Rule for the generation of all the wffs (inductive definition):
  - Each propositional letter is a wff
  - If ϕ is a wff then $(\neg\phi($ is a wff
  - If ϕ and ψ are wffs then $(\phi \wedge \psi), (\phi \vee \psi), (\phi \to \psi), (\phi \leftrightarrow \psi)$ are wffs
  - Nothing else is a wff

⋆ The use of parentheses is crucial to disambiguate scope:
$(A \vee (B \wedge C)) \neq ((A \vee B) \wedge C)$
$A \vee (B \wedge C) \neq (A \vee B) \wedge C$

7 Examples

- Well-formed formulas (wffs)

$$(((\neg A) \vee B) \to C) \tag{1}$$

$$((A \wedge B) \leftrightarrow \neg(C \vee \neg A)) \tag{2}$$

$$(\neg A \wedge A) \tag{3}$$

- Expressions which are not wff's

$$((A \vee B)\neg C) \tag{4}$$

$$\neg A \vee B \to C \tag{5}$$

$$A \to B \to C \tag{6}$$

⋆ Expression (4) and expressions (5), (6) are not wffs for different reasons. What is the difference?

8 Precedence among connectives

- ⋆ (5) and (6) could be wffs if we are not strict in the use of parentheses.
- Parentheses do not have meaning *per se*, they indicate the right way to read the expression. Usually a *convention* is used to simplify the expressions by reducing the number of parentheses.
  - Outermost parentheses are omitted
  - Connectives bind subformulas in this *order of precedence*:
    1. $\neg$
    2. $\vee$ and $\wedge$
    3. $\rightarrow$ and $\leftrightarrow$

    (1) and (5) are the same wff
- NB: Some add a left-to-right precedence or distinguish further levels to eliminate all ambiguities. I do not: (6) is not well-formed for me.
- ⋆ If in doubt, use parentheses!

9 Propositional Logic - Semantics

- Semantics is defined by a *Valuation function* V whose domain is the set of wffs and whose range is {0,1}.
  0 stands for *false*, 1 for *true*

- For non-atomic wffs, the effect of V is defined by induction:

$V(\neg\phi) = 1$	iff	$V(\phi) = 0$
$V(\phi \wedge \psi) = 1$	iff	$V(\phi) = 1$ and $V(\psi) = 1$
$V(\phi \vee \psi) = 1$	iff	$V(\phi) = 1$ or $V(\psi) = 1$
$V(\phi \rightarrow \psi) = 1$	iff	$V(\phi) = 0$ or $V(\psi) = 1$
$V(\phi \leftrightarrow \psi) =1$	iff	$V(\phi) = V(\psi)$

⋆ This doesn't tell us how to determine the truth of atomic wffs. It is arbitrary; each function V characterizes a *different model* or world.

⋆ *Truth tables* provide an exhaustive list of all possible models for the truth of a set of propositions.

10 Truth tables - 1

- The truth table for a proposition depends from the truth table of the atomic propositions that occur in it.

- Truth table for the connectives

A	B	$\neg A$	$A \wedge B$	$A \vee B$	$A \rightarrow B$
1	1	0	1	1	1
1	0	0	0	1	0
0	1	1	0	1	1
0	0	1	0	0	1

11 Truth tables - 2

A method to compute the semantics of arbitrary complex propositions.
Example:

$$P \wedge Q \rightarrow \neg(P \vee R)$$

P	Q	R	$P \wedge Q$	$P \vee R$	$\neg(P \vee R)$	$P \wedge Q \rightarrow \neg(P \vee R)$
0	0	0	0	0	1	1
0	0	1	0	1	0	1
0	1	0	0	0	1	1
0	1	1	0	1	0	1
1	0	0	0	1	0	1
1	0	1	0	1	0	1
1	1	0	1	1	0	0
1	1	1	1	1	0	0

12 Logical relations

- *Equivalence:* ϕ and ψ are logically equivalent iff for *every* valuation function V (i.e., every model), $\mathsf{V}(\phi) = \mathsf{V}(\psi)$. Notation: $\phi \equiv \psi$
- *Entailment (logical consequence):* ϕ logically entails ψ iff for *every* valuation function V, if $\mathsf{V}(\phi) = 1$ then $\mathsf{V}(\psi) = 1$. Notation: $\phi \models \psi$
- *Tautology:* a wff ϕ is a tautology iff for *every* valuation function V, $\mathsf{V}(\phi) = 1$. Tautologies are also called *valid* formulas.
- *Contradiction:* a wff ϕ is a contradiction iff for every valuation function V, $\mathsf{V}(\phi) = 0$.
- wffs which are neither tautologies nor contradictions are called *contingent* (their truth depends on the choice of V). Most interesting wffs are contingent.

13 Equivalent formulas

- Equivalent wffs are interchangeable, equivalence allows the substitution of complex formulas for simple ones

- Use the truth tables to check if following equivalences hold:
  - $A \wedge A \equiv A$
  - $A \vee B \equiv B \vee A$
  - $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$
  - $\neg(A \wedge B) \equiv \neg A \vee \neg B$
  - $A \rightarrow B \equiv \neg A \vee B$
  - $A \rightarrow B \equiv \neg B \rightarrow \neg A$
  - $A \rightarrow (B \rightarrow C) \equiv A \wedge B \rightarrow C$

14 Logical consequence

- The symbol '$\models$' is used to say that a formula is a *logical consequence* of some other wffs:

$$A, B, C \models D$$

  or of a set Φ of wffs.

$$\Phi \models D$$

- These formulas talk about the semantics of propositional logic. *They are not part of it*. They say that in *any* model where A, B, and C (or, in the second case, all formulas in Φ) are true, then D is true as well.
  - $(P \to Q), (P \to \neg Q) \models \neg P$
  - $(P \to P) \not\models \neg P$
  - ⊡ Check the previous two claims using truth tables.

15 Properties of logical consequence in propositional logic

A is a tautology	if and only if	$\models A$
$A \models B$	if and only if	$\models A \to B$ (Deduction theorem)
$A \equiv B$	if and only if	$\models A \leftrightarrow B$

- Check these claims using truth tables.

First-Order Logic

FOL, aka Predicate Logic

16 From Propositional Logic to Predicate Logic

- Propositional logic gives a clear semantics to connectives, and a compositional semantics to complex formulas
- Why is propositional logic not enough?
  - Unable to account for classical syllogisms such as:
    Every man is mortal,
    Socrates is a man,
    therefore Socrates is mortal.
  - Need to analyze the internal structure of propositions
  - Need to refer to entities, with *terms*, and to their properties and relations, with *predicates*
- Predicate logic, also called First-Order Logic

17 The language of predicate logic (FOL) - 1

- Instead of a vocabulary of letters for atomic propositions, we have:
  - Predicate constants: P_1, $\mathsf{P}_2, \mathsf{P}_3$... (or Loves, Mortal, Human...) often a special binary predicate is distinguished "=" (identity)
  - Terms, to serve as arguments of predicates:
    - Individual constants: a, b, c...
    - Variables: x, y, z...
    - [I omit functions here]

- Atomic propositions are then of the form $\mathsf{P}(t_1, ...t_n)$ where P is an n-ary predicate and t_i a term e.g. Human(s) is an atomic proposition with a unary predicate, Loves(j, m) with a binary predicate

18 The language of predicate logic (FOL) - 2

- We keep the propositional connectives and their rules for wffs
- We add *quantifiers* to bind the variables
  - $\forall$, the *universal* quantifier "for all"
  - $\exists$, the *existential* quantifier "there is at least one"
  - If ϕ is a wff and x a variable, then $\forall x \phi$ and $\exists x \phi$ are wffs

- *Every man is mortal*: $\forall x(\mathsf{Man}(x) \rightarrow \mathsf{Mortal}(x))$

19 Conventions

- The conventional precedence for the connectives we have seen in propositional logic holds here as well and is extended to quantifiers as follows:

  0 $\forall, \exists$

  1 $\neg$

  2 $\vee$ and $\wedge$

  3 $\rightarrow$ and $\leftrightarrow$

  - $\forall x \phi \rightarrow \psi$ is the same as $((\forall x \phi) \rightarrow \psi)$
  - $\exists y \forall z \phi \wedge \neg \exists x \psi$ is the same as $((\exists y (\forall z \phi)) \wedge (\neg (\exists x \psi)))$

- We write $\forall xy$ instead of $\forall x \forall y$, and $\exists xy$ for $\exists x \exists y$.

20 Free and bound variables

- In formulas $(\forall x \phi)$ and $(\exists x \phi)$, an occurrence of the variable x within ϕ is said to be *bound* or *quantified*.
- Occurrences of variables in a formula that are not bound are said to be *free*.

  (1) $\exists x \mathsf{Q}(x, \mathsf{c})$
  (2) $\forall xyz(\mathsf{R}(x, y, \mathsf{c}) \wedge \mathsf{P}(z))$
  (3) $\mathsf{P}(z) \wedge \forall z \mathsf{R}(x, y, z)$
  (4) $\forall x \exists y \mathsf{Q}(x, y)$
  (5) $\mathsf{P}(x) \rightarrow \mathsf{Q}(x)$
  (6) $\forall z(\mathsf{R}(x, y, \mathsf{c}) \wedge \exists z \mathsf{P}(z) \wedge Q(z))$

- A *sentence* or *closed formula* is a formula in which no variable occurs free (e.g. formulas (1), (2) and (4)).

21 Formulas for "some"

Before formally introducing the semantics of the language, let's get the intuition on how to use it.

- "There are white cats" (there is at least one white cat)
  - $\exists x(\mathsf{Cat}(x) \wedge \mathsf{White}(x))$
- "Some chairs are broken" (there is at least one broken chair)
  - $\exists x(\mathsf{Chair}(x) \wedge \mathsf{Broken}(x))$
- The following are not correct for capturing the previous natural language sentences, why?
  - $\exists x(\mathsf{Cat}(x) \rightarrow \mathsf{White}(x))$
  - $\exists x(\mathsf{Chair}(x) \rightarrow \mathsf{Broken}(x))$

22 Formulas for “for all”

- “Any cook knows how to cook pizza” (also with every / each / a)
  “All the cooks know how to cook pizza” (also with bare plural)
  “If somebody is a cook, then s/he knows how to cook pizza”
  - $\forall x(\mathsf{Cook}(x) \rightarrow \mathsf{KnowsCooking}(x, \mathsf{pizza}))$
- “Everybody is a cook and knows how to cook pizza”
  - $\forall x(\mathsf{Cook}(x) \land \mathsf{KnowsCooking}(x, \mathsf{pizza}))$
- “Chianti is the only good wine, if any” that is,
  “if a wine is good, then it is Chianti”
  - $\forall x(\mathsf{Wine}(x) \land \mathsf{Good}(x) \rightarrow \mathsf{Chianti}(x))$

23 Another example... step by step

- "Whoever owns a dog loves animals"
  - First, we list the needed predicates and constants.
    - $\mathsf{Own}(x, y)$, $\mathsf{Dog}(x)$, $\mathsf{LoveAnimals}(x)$
    - no constants
  - Set variables, connectives and some parentheses:
    - for any person, if she owns a dog then she loves animals
    - for any x, if there is a y such that $\mathsf{Dog}(y) \wedge \mathsf{Own}(x, y)$ then $\mathsf{LoveAnimals}(x)$
  - Now fix the quantifiers.
    - for any x, $(\exists y(\mathsf{Dog}(y) \wedge \mathsf{Own}(x, y)) \rightarrow \mathsf{LoveAnimals}(x))$
    - $\forall x(\exists y(\mathsf{Dog}(y) \wedge \mathsf{Own}(x, y)) \rightarrow \mathsf{LoveAnimals}(x))$
  - Finally, check the parentheses and the overall result.

24 Exercises

- Translate into English:
  - $\forall xyz(\mathsf{SpeaksLanguage}(x, z) \wedge \mathsf{SpeaksLanguage}(y, z) \rightarrow \mathsf{Understand}(x, y) \wedge \mathsf{Understand}(y, x))$
  - $\forall x \exists y \mathsf{Loves}(x, y)$
  - $\forall x \exists y \mathsf{Loves}(y, x)$
  - $\exists x \forall y \mathsf{Loves}(x, y)$
- Translate into FOL:
  - John has a son which is a student.
  - John saw a squirrel.
  - A man runs. / Some runner is a man. / No one runs.
  - Monkeys are primates, which are animals.
  - A professor is someone who teaches at school.

25 Semantics of FOL: Extending V

- So far we have used the function V only to assign a truth value to whole propositions, atomic or complex.
- But V should also be used to characterize the meaning of predicate constants and individual constants.
  - For the non-logical symbols, we want to make the following *associations*:

proposition	truth value 1 or 0
constant	individual entity
predicate with arity 1	set of entities
predicate with arity $n > 1$	set of n-tuples of entities

- To this end, it is useful to remind our knowledge of *sets*.

26 Set theory pills

- If A, B and C are sets of individuals and x is an individual:
  - $x \in A$ "x is an element of A"
  - $A \subseteq B$ "A is a subset of B" (true even if $A = B$)
  - $A \cup B = C$ "the union of A and B is equal to C"
  - $A \cap B = C$ "The intersection between A and B is C"
  - $\emptyset$ is the empty set
- Recall moreover that:
  - The identity of a set is entirely determined by its members (extensionality): $A = B$ iff (for all x, $x \in A$ iff $x \in B$);
  - The same element doesn't appear more than once in the same set;
  - Elements in a set are not ordered.

27 Specifying Set Content

- With D = domain, the set of all individuals, the contents of a set can be specified directly:
  - A := {a, e, i, o, u}
- Or using an abstraction:
  - A := $\{x \in D$: x is a vowel$\}$
    "The set of all the x's in D such that x is a vowel"
  - A := $\{x \in D$: Paris is a beautiful city$\}$ *(vacuous restriction: $A = D$)*
  - A := $\{x \in D$: $x \neq x)$ *(contradictory restr.: $A = \emptyset$)*
  - A := $\{y \in D : \{x \in D : x \text{ loves } y\} = \emptyset\}$
    What does this last case mean?

28 Relations as sets

- An n-ary relation R on D is a set of n-uples $< x_1, x_2, ...x_n >$, i.e., R $\subseteq$ D^n
  - Example: the binary relation on $\mathbb{N}$ "*is smaller than*" is the set $\{< 0, 1 >, < 0, 2 >, < 1, 2 >, < 0, 3 >, < 1, 3 >, < 2, 3 >, ...\}$
- A unary function F can always be rendered as a set containing ordered pairs <x,y> (*<input,output>*) such that:
  - For every x, if there are y and z such that <x,y>∈F and <x,z>∈F, then y=z.
  - Example: "*next positive integer of*" {<0,1>, <1,2>, <2,3>, ...}
- Without the unicity constraint, we would have a binary relation which is not a function.

29 Semantics of Predicate Logic (FOL) - 1

Two main ingredients:

- A *model* of FOL is a pair $M =< D, I >$, where D is the domain of individuals and I the *interpretation* function.
- g is an *assignment function*.
- Constraints on I and g:
  - if P is a n-ary predicate, $I(\mathsf{P}) \subseteq D^n$
  - if a is a constant, $I(\mathsf{a}) \in D$
  - if x is a variable, $g(x) \in D$

30 Semantics of Predicate Logic (FOL) - 2

- The semantics of a FOL formula in the model M with the assignment g is given by the function $[\![\ \ ]\!]_g^M$ (extension of our previous V)
  - $[\![\mathsf{P}(t_1, ...t_n)]\!]_g^M = 1$ iff $< [\![t_1]\!]_g^M, ...[\![t_n]\!]_g^M > \in I(\mathsf{P})$
  - $[\![t]\!]_g^M = I(t)$ if t is a constant
  - $[\![t]\!]_g^M = g(t)$ if t is a variable
  - semantics of connectives as for propositional logic
  - $[\![\forall x \phi]\!]_g^M = 1$ iff $[\![\phi]\!]_{g'}^M = 1$ for *all* g' identical to g except maybe in x
  - $[\![\exists x \phi]\!]_g^M = 1$ iff $[\![\phi]\!]_{g'}^M = 1$ for *some* g' identical to g except maybe in x

31 Truth, Satisfiability and Validity

- $[\![\phi]\!]_g^M = 1$ is noted $M, g \models \phi$
  "ϕ is true in model M under assignment g"
- For *closed* formulas, truth does not depend on the choice of the assignment g, so we write simply $M \models \phi$
- Let ϕ be a closed formula (or a set of closed formulas), we write $Mod(\phi)$ to denote the set of all models of ϕ.
- A formula ϕ is *satisfiable* if there is a model M and an assignment g such that ϕ is true in M under g: $M, g \models \phi$
- A set Γ of wffs is *satisfiable* if there is a model M and an assignment g such that $M, g \models \phi$ for all formulas $\phi \in \Gamma$.
- A formula is *valid* (or a tautology) if ϕ is true in every model M and assignment g, this is noted simply $\models \phi$

32 Example - 1

- We want to give meaning to wffs in the language containing constants Ba, De, Du, unary predicates U and G, binary predicate L.
- A possible model:
  - $D = \{Bassi, Dellai, Durnwalder, Guarino\}$

    $I(\mathsf{Ba}) = Bassi$ (University of Trento Rector)

    $I(\mathsf{De}) = Dellai$ (Trento Province Leader)

    $I(\mathsf{Du}) = Durnwalder$ (Bolzano Province Leader)

    $I(\mathsf{U}) = \{Bassi\}$ (employes of University of Trento)

    $I(\mathsf{G}) = \{Durnwalder\}$ (German native speakers)

    $I(\mathsf{L}) = \{\langle Bassi, Dellai\rangle, \langle Bassi, Guarino\rangle, \langle Dellai, Guarino\rangle\}$ (to live in the same province)

33 Example - 2

- Let see if the the described model satisfies the following wffs (if the following wffs are true in the model):
  1. $\mathsf{U}(\mathsf{Ba})$
  2. $\mathsf{L}(\mathsf{De}, \mathsf{Du})$
  3. $\neg\mathsf{U}(\mathsf{Du})$
  4. $\forall x(\mathsf{U}(x) \rightarrow \mathsf{L}(x, \mathsf{De}))$
  5. $\exists xy(\mathsf{L}(x, y) \wedge \mathsf{G}(y))$

34 Exercice

- Give a (finite) model for the following formulas:
  1. $\forall xyz(\mathsf{SpeaksLanguage}(x,z) \wedge \mathsf{SpeaksLanguage}(y,z) \rightarrow \mathsf{Understand}(x,y) \wedge \mathsf{Understand}(y,x))$
  2. $\forall x \exists y\ \mathsf{Loves}(x,y)$
  3. $\exists x \forall y\ \mathsf{Loves}(x,y)$

35 Logical consequence and logical equivalence

- Let Γ be a set of closed formulas and φ a closed formula:
  φ is a *logical consequence* of Γ (or Γ *entails* φ), written $\Gamma \models \varphi$,
  if for any model M such that $M \models \psi$ for all $\psi \in \Gamma$, we also have $M \models \varphi$.
- Semantic deduction theorem:
  $\phi_1, \phi_2 ... \phi_n \models \psi$ iff $\models \phi_1 \wedge \phi_2 ... \wedge \phi_n \rightarrow \psi$
- Two closed formulas φ and ψ are said to be *logically* (or *semantically*) *equivalent*, written $\varphi \equiv \psi$,
  if for all models M we have $M \models \varphi$ iff $M \models \psi$.

36 Equivalent formulas - 1

It easy to prove that the following logical equivalences:

- *renaming of bound variables*

  $\forall x\ P(x) \equiv \forall y\ P(y)$

- *order of variables bound by the same quantifier*

  $\forall xy\ P(x, y) \equiv \forall yx\ P(x, y)$

- *occurrence of quantifiers that bound no variable*

  $\forall xy\ P(y) \equiv \forall y\ P(y)$

37 Equivalent formulas - 2

Important cases of equivalent formulas: *quantifiers and negation*.

- $\forall x\ P(x) \equiv \neg\exists x\ \neg P(x)$
- $\neg\forall x\ P(x) \equiv \exists x\ \neg P(x)$
- $\exists x\ P(x) \equiv \neg\forall x\ \neg P(x)$
- $\neg\exists x\ P(x) \equiv \forall x\ \neg P(x)$

38 Equivalent formulas - 3

Important cases of equivalent formulas: *quantifiers,* $\wedge$ *and* $\vee$

- $\forall x(P(x) \wedge Q(x)) \equiv \forall x\ P(x) \wedge \forall x\ Q(x)$
- $\exists x(P(x) \vee Q(x)) \equiv \exists x\ P(x) \vee \exists x\ Q(x)$
- $\forall x(P(x) \vee Q) \equiv \forall x\ P(x) \vee Q$ where $x \notin \texttt{var}(Q)$
- $\exists x(P(x) \wedge Q) \equiv \exists x\ P(x) \wedge Q$ where $x \notin \texttt{var}(Q)$

39 Equivalent formulas - 4

Important cases of equivalent formulas: *quantifiers and* $\rightarrow$

- $\forall x\ P(x) \rightarrow Q \equiv \exists x(P(x) \rightarrow Q)$ where $x \notin \texttt{var}(Q)$
- $\exists x\ P(x) \rightarrow Q \equiv \forall x(P(x) \rightarrow Q)$ where $x \notin \texttt{var}(Q)$
- $P \rightarrow \forall x\ Q(x) \equiv \forall x(P \rightarrow Q(x))$ where $x \notin \texttt{var}(P)$
- $P \rightarrow \exists x\ Q(x) \equiv \exists x(P \rightarrow Q(x))$ where $x \notin \texttt{var}(P)$

- Using these equivalences, we can always rewrite a formula so that all quantifiers are on the left. E.g.:
  $\forall x\ R(x) \wedge \exists y(\forall x\ P(x) \rightarrow Q(y)) \equiv \forall x \exists y z(R(x) \wedge (P(z) \rightarrow Q(y)))$

Other logics

40 Orders of quantification

- First-order: quantification only on variables denoting individuals
- (Monadic) Second-order: quantification also on variables denoting properties, i.e., sets of individuals
  two kinds of variables
- Full Second-order: quantification on any sort of variables ranging on n-ary relations
- Higher-order: predicates of predicates, and quantification on these new sorts

41 Why not using the most expressive logic?

- Because we want to do reasoning too!
- FOL comes with *deductive systems* (axioms + rules of inference, sequent calculus, tableaux...) to reason *syntactically*.
  Mechanically *proving theorems* instead of relying on handling models and logical consequence
- Possible because such deductive systems are *sound and complete*, i.e, deduction ($\vdash$) tightly matches logical consequence ($\models$)
- Second-order and higher-order logics do **not** have complete deductive systems

42 Modal logics

- Not only one truth in one world: possibility and certainty
- $\Box\phi$: "it is necessary that ϕ"
- $\Diamond\phi$: "it is possible that ϕ"
- *Possible worlds* semantics: one model is a set of worlds related by an accessibility function
- Example: Temporal logic
  - One world = a time, accessibility = precedence between worlds
  - Two sets of modalities, one for the future ("it will always be true" / "it will be true at some point in the future"), one for the past.
- Others: Epistemic logics (beliefs), Deontic logics (obligations)...

43 Decidability

- Completeness is not all: we need *effective* proof methods
- Effective: the calculus should de done in finite time, i.e., it should terminate
- A problem (i.e., determining if a formula is valid (tautology), determining if a formula is satisfiable (consistent)) is decidable iff there is an effective method to solve it
- Deduction in FOL is only **semi-decidable** (as soon as there is a n-ary predicate with $n > 1$): one can prove effectively whether a formula is a theorem, but the proof that a formula is **not** a theorem may not terminate

  - Important to extract decidable fragments of FOL

44 Introduction to Description Logics

- Decidable fragments of FOL for knowledge representation and reasoning, especially for (lightweight) ontology representation and taxonomic reasoning
- Only unary and binary predicates
  - Unary predicates: “concepts” in DLs in general, “class” in OWL
  - Binary predicates: “role” in DLs in general, “property” in OWL (beware, in formal ontology and in philosophy, a property is a unary predicate)
- Very restricted quantification, in fact, variables are even hidden

45 Introduction to Description Logics ($\mathcal{SHIQ}$)

- Vocabulary
  - individual constants a, b, c...
  - concept constants $C, D, ...$, including $\top$ and $\bot$
  - role constants R, Q...
  - concept *constructors*: $\sqcap$ (conjunction), $\sqcup$ (disjunction), $\neg$ (negation), $\forall$ (universal restriction), $exists$ (existential restriction)
  - if C and D are concepts and R a role, then $C \sqcap D$, $C \sqcup D$, $\neg C$, $\forall R \cdot C$, $\exists R \cdot C$ are *concepts*
- wffs
  if C and D are concepts, a and b individual constants, and R a role
  $C(a)$ [or $a : C$] , $R(a, b)$ [or $(a, b) : R$] and $C \sqsubseteq D$ are wffs

46 Semantics

- Models: $\mathsf{M} = \langle \mathsf{D}, \mathsf{I} \rangle$
- $\mathsf{I}(C \sqcap D) = \mathsf{I}(C) \cap \mathsf{I}(D)$, $\mathsf{I}(C \sqcup D) = \mathsf{I}(C) \cup \mathsf{I}(D)$, $\mathsf{I}(\neg C) = \mathsf{D} \setminus \mathsf{I}(C)$
- $\mathsf{I}(\exists R{\cdot}C) = \{x \in \mathsf{D} :$ there is y in D s.t. $\langle x, y \rangle \in \mathsf{I}(R)$ and $y \in \mathsf{I}(C)\}$
- $\mathsf{I}(\forall R \cdot C) = \{x \in \mathsf{D} :$ for all y in D if $\langle x, y \rangle \in \mathsf{I}(R)$ then $y \in \mathsf{I}(C)\}$
- $\mathsf{M} \models C(a)$ iff $\mathsf{I}(a) \in \mathsf{I}(C)$
- $\mathsf{M} \models R(a, b)$ iff $\langle \mathsf{I}(a), \mathsf{I}(b) \rangle \in \mathsf{I}(R)$
- $\mathsf{M} \models C \sqsubseteq D$ iff $\mathsf{I}(C) \subseteq \mathsf{I}(D)$