22년 삼성 AI 전문가과정
6월 8일 수요일 1교시
장병탁

인공지능

8차시 : Knowledge Representation

서울대학교 컴퓨터공학부
담당 교수: 장병탁

Seoul National University
Byoung-Tak Zhang

Lecture Overview

인공지능

8차시 : Knowledge Representation

서울대학교 컴퓨터공학부
담당 교수: 장병탁

Seoul National University
Byoung-Tak Zhang

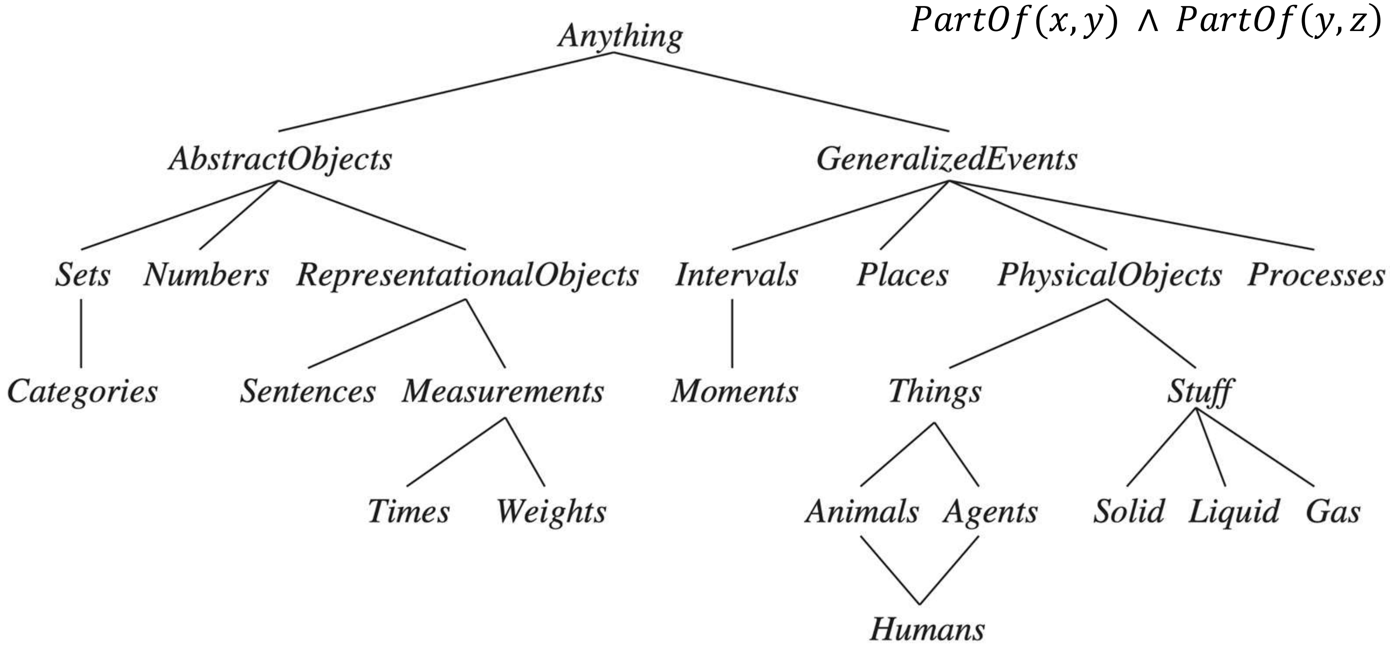
Introduction

- **Knowledge-Based Agents** (Previous lectures)
  - How an agent with a knowledge base can make inferences that enable it to act appropriately
- **Knowledge Representation** (This lecture)
  - What *content* to put into such an agent's knowledge base—how to represent facts about the world in a form that can be used to reason and solve problems
    - First-order logic
    - Semantic networks
    - Scripts, frames
- **Later Lectures …..**
  - Different representation formalisms such as hierarchical task networks for reasoning about plans (Ch 11), Bayesian networks for reasoning with uncertainty (Ch 13), Markov models for reasoning over time (Ch 16), and deep neural networks for reasoning about images, sounds, and other data (Ch 22)

Ontological Engineering

Representing abstract concepts such as *Events, Time, Physical Objects,* and *Beliefs* that occur in many different domains

$b \in Butter \wedge PartOf(p,b) \Rightarrow p \in Butter$
$b \in Butter \Rightarrow MeltingPoint(b, Centigrade(30))$
$PartOf(x,y) \wedge PartOf(y,z) \Rightarrow PartOf(x,z), PartOf(x,x)$



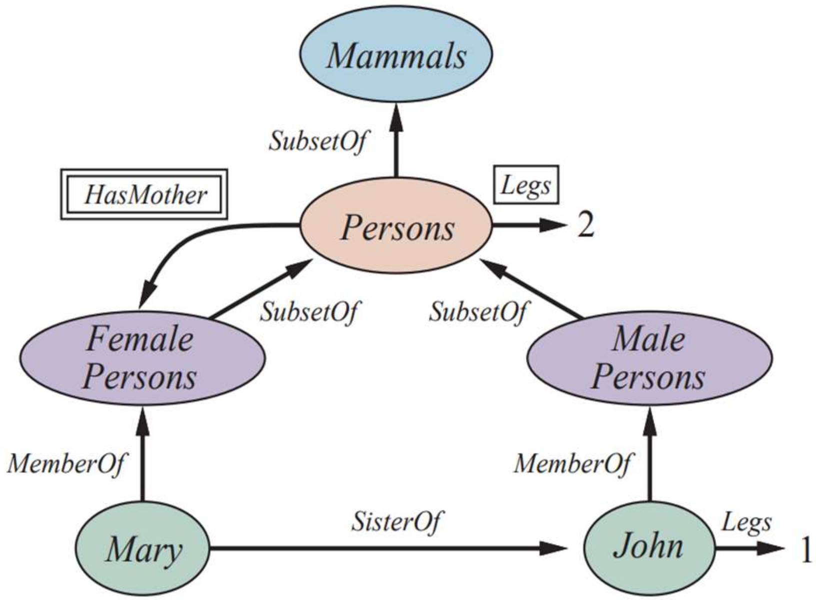
General concepts

Specific concepts

Semantic Networks

Semantic Networks

- Graphical notation with underlying logical representation
- A form of logic, but not FOL
- Capable of representing objects, categories, and relations among objects, etc.
- Convenient representation of inheritance, multiple inheritance (sometimes)



출처: Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4rd Edition). Pearson

Script

EAT-AT-RESTAURANT Script

Props: (Restaurant, Money, Food, Menu, Tables, Chairs)
Roles: (Hungry-Persons, Wait-Persons, Chef-Persons)
Point-of-View: Hungry-Persons
Time-of-Occurrence: (Times-of-Operation of Restaurant)
Place-of-Occurrence: (Location of Restaurant)

Event-Sequence:
- first: Enter-Restaurant Script
- then: if (Wait-To-Be-Seated-Sign or Reservations)
  then Get-Maitre-d's-Attention Script
- then: Please-Be-Seated Script
- then: Order-Food-Script
- then: Eat-Food-Script unless (Long-Wait) when Exit-Restaurant-Angry Script
- then: if (Food-Quality was better than Palatable)
  then Compliments-To-The-Chef Script
- then: Pay-For-It-Script
- finally: Leave-Restaurant Script

Order-Food-Script

Pay-For-It-Script

Outline (Lecture 8)

Stuart Russell & Peter Norvig (2021), Artificial Intelligence: A Modern Approach (4th Edition)

8.1 Ontological Engineering

8.1 Ontological Engineering (1/3)

Knowledge Representation (KR)

- How to represent diverse facts about the real world in a form that can be used to reason and solve problems
  - First-order logic, Bayesian networks, Markov models, deep neural networks, etc.
- No matter what representation you use - the facts about the world need to be handled

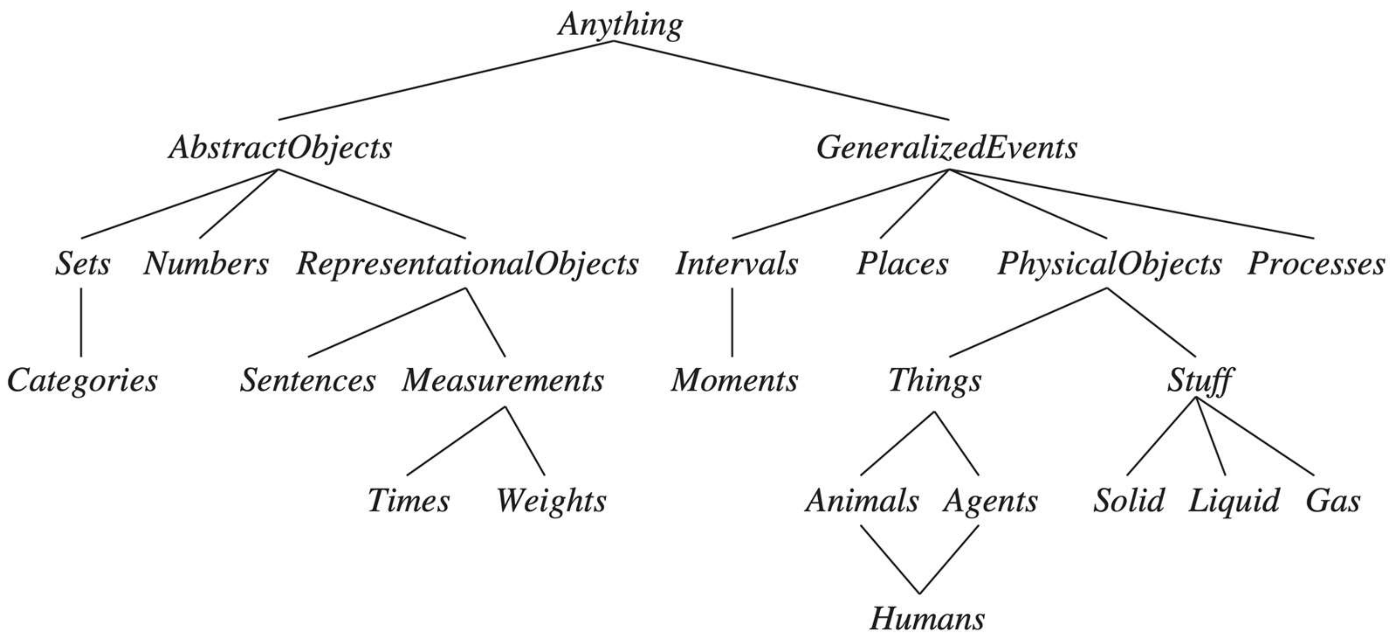
Ontological Engineering

- Representing abstract concepts such as Events, Time, Physical Objects, and Beliefs that occur in many different domains
- Leave placeholders where new knowledge for any domain can fit in
  - e.g. define what it means to be a physical object, and the details of different types of objects—robots, televisions, books, or whatever—can be filled in later

8.1 Ontological Engineering (2/3)

Upper ontology

- General framework of concepts



General concepts

Specific concepts

출처: Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4rd Edition). Pearson

8.1 Ontological Engineering (3/3)

Special-purpose ontology

- Designed to represent a specific domain of knowledge
  - Open Biomedical Ontology

General-purpose ontology

- Should be applicable in any special-purpose domain
- Different areas of knowledge must be unified
- Example:
  - CYC: created by a team of trained ontologists
  - DBPEDIA: created by importing categories, attributes, and values from an existing database
  - Google Knowledge Graph: uses semistructured content from Wikipedia, combining it with other content gathered from across the web under human curation. It contains over 70 billion facts

8.2 Categories and Objects

8.2 Categories and Objects (1/5)

Objects and categories

- Much reasoning takes place at the level of categories rather than individual objects
- Representing categories in first-order logic (FOL)
  - Predicate

    e.g. $Basketball(b)$
  - Reified object

    e.g. $Basketballs, Member(b, Basketballs)$
- Categories organize knowledge through inheritance
- The Class/Subclass relationships among $Food, Fruit$ and $Apples$ is a taxonomy

Example facts using first-order logic (FOL)

- An object is a member of a category

  $BB_9 \in Basketballs$
- A category is a subclass of another category

  $Basketballs \subset Balls$
- All members of a category have some properties

  $(x \in Basketballs) \Longrightarrow Spherical(x)$
- Members of a category can be recognized by some properties

  $Orange(x) \wedge Round(x) \wedge Diameter(x) = 9.5" \wedge x \in Balls \Longrightarrow x \in Basketballs$
- A category as a whole has some properties

  $Dogs \in DomesticatedSpecies$

8.2 Categories and Objects (2/5)

Relations between categories

- **Disjoint:** The categories have no members in common
- **Exhaustive Decomposition:** Any individual must be in one of the subcategories
- **Partition:** Exhaustive decomposition of disjoint sets

  $Disjoint(\{Animals, Vegetables\})$

  $ExhaustiveDecomposition(\{Americans, Canadians, Mexicans\}, NorthAmericans)$

  $Partition(\{Animals, Plants, Fungi, Protista, Monera\}, LivingThings)$

- (Note that the $ExhaustiveDecomposition$ of $NorthAmericans$ is not a $Partition$, because some people have dual citizenship.) The three predicates are defined as follows:

  $Disjoint(s) \Leftrightarrow (\forall c_1, c_2\ c_1 \in s \land c_2 \in s \land c_1 \neq\land\ c_2 \Rightarrow Intersection(c_1, c_2) = \{\})$

  $ExhaustiveDecomposition(s, c) \Leftrightarrow (\forall i\ i \in c \Leftrightarrow \exists c_2\ c_2 \in s \land i \in c_2)$

  $Partition(s, c) \Leftrightarrow Disjoint(s) \land ExhaustiveDecomposition(s, c)$

- Categories can also be *defined* by providing necessary and sufficient conditions for membership. For example, a bachelor is an unmarried adult male: $x \in Bachelors \Leftrightarrow Unmarried(x) \land x \in Adults \land x \in Males$

8.2 Categories and Objects (3/5)

Physical composition

- $PartOf$ relation: one thing is part of another
  - Can group objects into $PartOf$ hierarchies
  - Is transitive and reflexive:

    e.g. $PartOf(x,y) \land PartOf(y,z) \Rightarrow PartOf(x,z), PartOf(x,x)$
- Categories of composite objects
  - characterized by structural relations among parts
  - e.g. a biped is an object with exactly two legs attached to a body:

$$
\begin{aligned}
Biped(a) \Rightarrow\ & \exists l_1, l_2, b\ Leg(l_1) \land Leg(l_2) \land Body(b) \land \\
& PartOf(l_1, a) \land PartOf(l_2, a) \land PartOf(b, a) \land \\
& Attached(l_1, b) \land Attached(l_2, b) \land \\
& l_1 \neq l_2 \land [\forall l_3\ Leg(l_3) \land PartOf(l_3, a) \Rightarrow (l_3 = l_1 \lor l_3 = l_2)].
\end{aligned}
$$

8.2 Categories and Objects (4/5)

Measurements

- Quantitative measures: values assigned to the properties such as height, mass, cost, and so on
  - Easy to represent by combining a units function with a number

$$Diameter(Basketball_{12}) = Inches(9.5)$$
$$ListPrice(Basketball_{12}) = \$(19)$$
$$Weight\big(BunchOf(\{Apple_1, Apple_2, Apple_3\})\big) = Pounds(2)$$
$$d \in Days \Rightarrow Duration(d) = Hours(24).$$

- Natural kinds with no agreed scale
  - Measures are not numbers, but can be ordered

$$e_1 \in Exercises \land e_2 \in Exercises \land Wrote(Norvig, e_1) \land Wrote(Russell, e_2) \Rightarrow$$
$$Difficulty(e_1) > Difficulty(e_2)$$
$$e_1 \in Exercises \land e_2 \in Exercises \land Difficulty(e_1) \land Difficulty(e_2) \Rightarrow$$
$$ExpectedScore(e_1) < ExpectedScore(e_2)$$

8.2 Categories and Objects (5/5)

Objects: Things and stuff

- *Stuff*: a generic name for mass nouns
- *Things*: a generic name for count nouns

$$b \in Butter \land PartOf(p, b) \Rightarrow p \in Butter$$

$$b \in Butter \Rightarrow MeltingPoint(b, Centigrade(30))$$

- Intrinsic and extrinsic properties
  - Intrinsic properties belong to the very substance of the object for stuff, mass nouns

    e.g. density, boiling point, flavor, color, ownership, and so on.
  - Extrinsic properties are not retained under subdivision for things, count nouns

    e.g. weight, length, shape, and so on.

8.3 Events

8.3 Events (1/3)

Event calculus

- Deals with a much richer range of actions than situation calculus
  - In situation calculus, actions are discrete, instantaneous, and happen one at a time.
- Objects of event calculus: events, fluents, and time points
- The complete set of predicates for one version of the event calculus:

$T(f, t_1, t_2)$	Fluent f is true for all times between t_1 and t_2
$Happens(e, t_1, t_2)$	Event e starts at time t_1 and ends at t_2
$Initiates(e, f, t)$	Event e causes fluent f to become true at time t
$Terminates(e, f, t)$	Event e causes fluent f to cease to be true at time t
$Initiated(f, t_1, t_2)$	Fluent f become true at some point between t_1 and t_2
$Terminated(f, t_1, t_2)$	Fluent f cease to be true at some point between t_1 and t_2
$t_1 < t_2$	Time point t_1 occurs before time t_2

8.3 Events (2/3)

Time

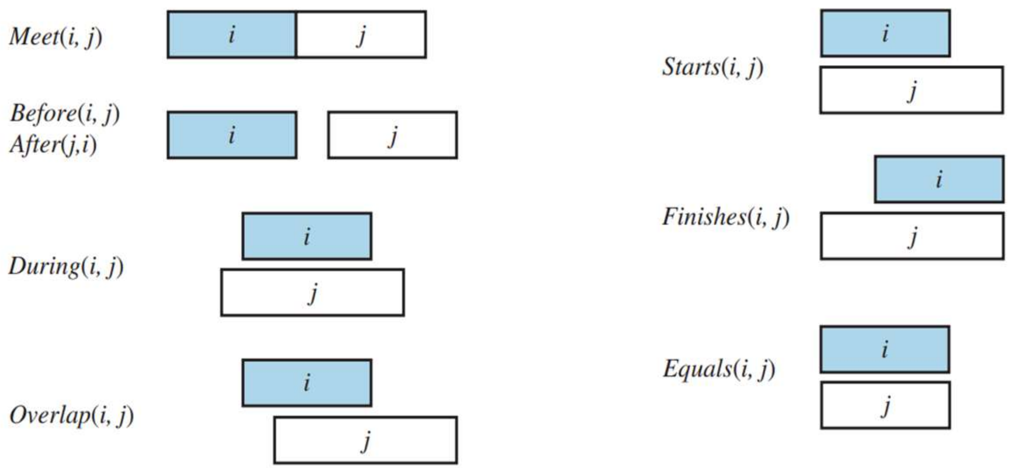
- Time intervals: Moment and extended interval
- Moment: has temporal duration of zero

$Partition(\{Moments, ExtendedInterval\}, Intervals)$

$i \in Moments \Leftrightarrow Duration(i) = Seconds(0).$

$$
\begin{array}{lcl}
Meet(i,j) & \Leftrightarrow & End(i) = Begin(j) \\
Before(i,j) & \Leftrightarrow & End(i) < Begin(j) \\
After(j,i) & \Leftrightarrow & Before(i,j) \\
During(i,j) & \Leftrightarrow & Begin(j) < Begin(i) < End(i) < End(j) \\
Overlap(i,j) & \Leftrightarrow & Begin(i) < Begin(j) < End(i) < End(j) \\
Starts(i,j) & \Leftrightarrow & Begin(i) = Begin(j) \\
Finishes(i,j) & \Leftrightarrow & End(i) = End(j) \\
Equals(i,j) & \Leftrightarrow & Begin(i) = Begin(j) \wedge End(i) = End(j)
\end{array}
$$

Predicates on time intervals



출처: Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4rd Edition). Pearson

8.3 Events (3/3)

Fluents and objects

출처: Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4rd Edition). Pearson

- Physical objects can be viewed as generalized events (a chunk of space-time)
- We can describe the changing properties of objects using state fluents
  - e.g. object $USA \Rightarrow President(USA)$ denotes a single object that consists of different people at different times, To say that George Washington was president throughout 1790, we can write
    $T(Equals(President(USA), GeorgeWashington), Begin(AD1790), End(AD1790))$

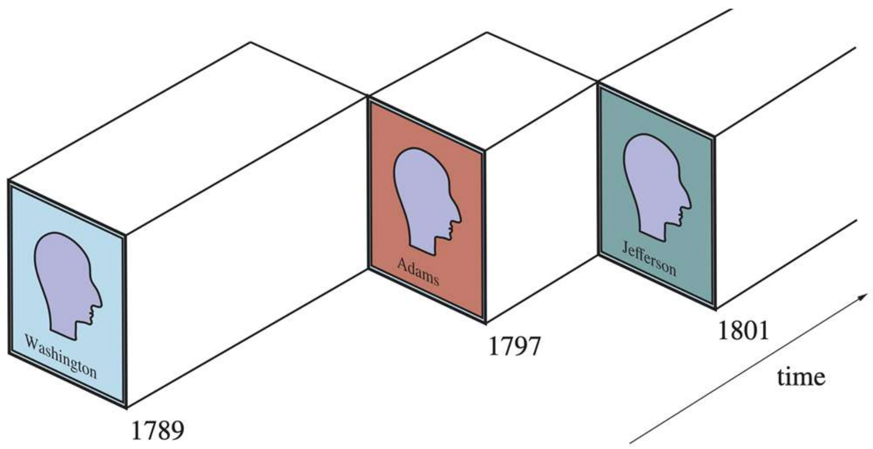


Figure 10.3 A schematic view of the object *President*(*USA*) for the early years.

Script

EAT-AT-RESTAURANT Script

Props:	(Restaurant, Money, Food, Menu, Tables, Chairs)
Roles:	(Hungry-Persons, Wait-Persons, Chef-Persons)
Point-of-View:	Hungry-Persons
Time-of-Occurrence:	(Times-of-Operation of Restaurant)
Place-of-Occurrence:	(Location of Restaurant)

Event-Sequence:

first:	Enter-Restaurant Script
then:	if (Wait-To-Be-Seated-Sign or Reservations) then Get-Maitre-d's-Attention Script
then:	Please-Be-Seated Script
then:	Order-Food-Script
then:	Eat-Food-Script unless (Long-Wait) when Exit-Restaurant-Angry Script
then:	if (Food-Quality was better than Palatable) then Compliments-To-The-Chef Script
then:	Pay-For-It-Script
finally:	Leave-Restaurant Script

Order-Food-Script

Pay-For-It-Script

8.4. Mental Objects and Modal Logic

8.4 Mental Objects and Modal Logic (1/3)

A model of mental objects that are someone's head and of the mental processes

- **Propositional attitude**
  - Mental objects such as $Believes$, $Knows$, $Wants$, and $Informs$

    $Knows(Lois, CanFly(Clark))$
  - Reification: turning a proposition into an object

    $(Superman = Clark) \wedge Knows\big(Lois, CanFly(Superman)\big)$

    $\vDash Knows(Lois, CanFly(Clark))$
  - Referential transparency
    - It doesn't matter what term a logic uses to refer to an object, what matters is the object that the term names.

    $Clark$ and $Superman$ are two names for the same person

    If an agent knows that $2 + 2 = 4$ and $4 < 5$, then the agent knows that $2 + 2 < 5$

8.4 Mental Objects and Modal Logic (2/3)

A model of mental objects that are someone's head and of the mental processes

- **Modal logic:** including modal operators that take sentences as arguments
  - Regular logic: a single modality, i.e. the modality of truth
  - "*A* knows *P*" is represented with the notation $\mathbf{K}_A P$ (**K** is the modal operator for knowledge)
  - Modal operator **K** takes an agent (written as the subscript) and a sentence.
  - Modal logic can be used to reason about nested knowledge sentences: what one agent knows about another agent's knowledge

$$\mathbf{K}_{Lois}[\mathbf{K}_{Clark} Identity(Superman, Clark) \vee \mathbf{K}_{Clark} \neg Identity(Superman, Clark)]$$

  - Modal logic solves some tricky issues with the interplay of quantifiers and knowledge

"Bond knows that someone is spy": First, $\exists x\ \mathbf{K}_{Bond} Spy(x)$ Second, $\mathbf{K}_{Bond} \exists x\ Spy(x)$

In each accessible world there is an x that is a spy, but it need not be the same x in each world

8.4 Mental Objects and Modal Logic (3/3)

Other modal logics

- Modal operators in **linear temporal logic**
  - **X** P: "P will be true in the next time step"
  - **F** P: "P will eventually (**F**inally) be true in some future time step"
  - **G** P: "P is always (**G**lobally) true"
  - P **U** Q: "P remains true until Q occurs"

8.5 Reasoning Systems for Categories

8.5 Reasoning Systems for Categories (1/6)

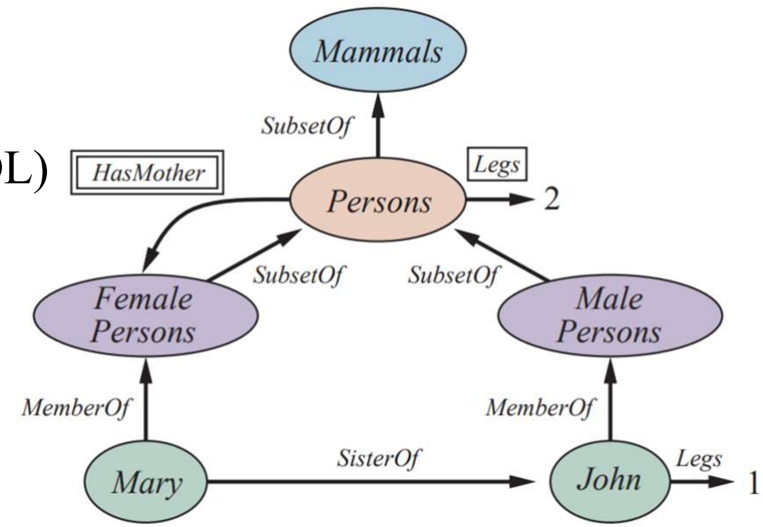
Two systems for reasoning with categories

- Semantic Networks
  - Graphical aids for visualizing knowledge
  - Mechanisms for inferring properties of objects based on category membership
- Description Logics
  - Formal language for constructing and combining category definitions
  - Algorithms for classifying objects and determining subsumption relationships

8.5 Reasoning Systems for Categories (2/6)

Semantic Networks

- Graphical notation with underlying logical representation
- A form of logic, but not first-order logic (FOL)
- Capable of representing objects, categories, and relations among objects, etc.
- Convenient representation of inheritance, multiple inheritance (sometimes)
- Example of semantic network

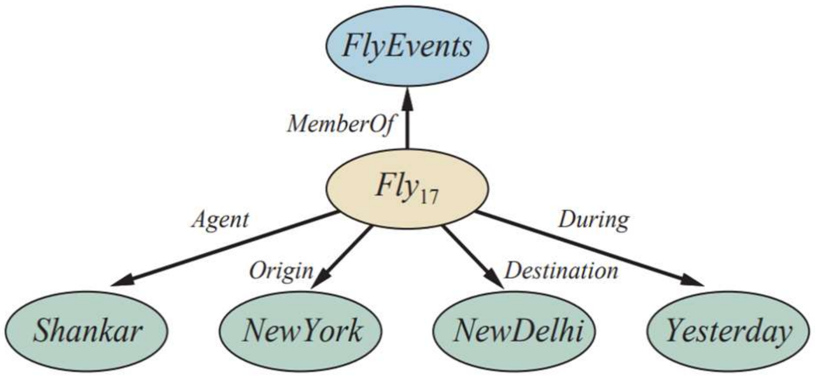


출처: Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4rd Edition). Pearson

8.5 Reasoning Systems for Categories (3/6)

Semantic Networks

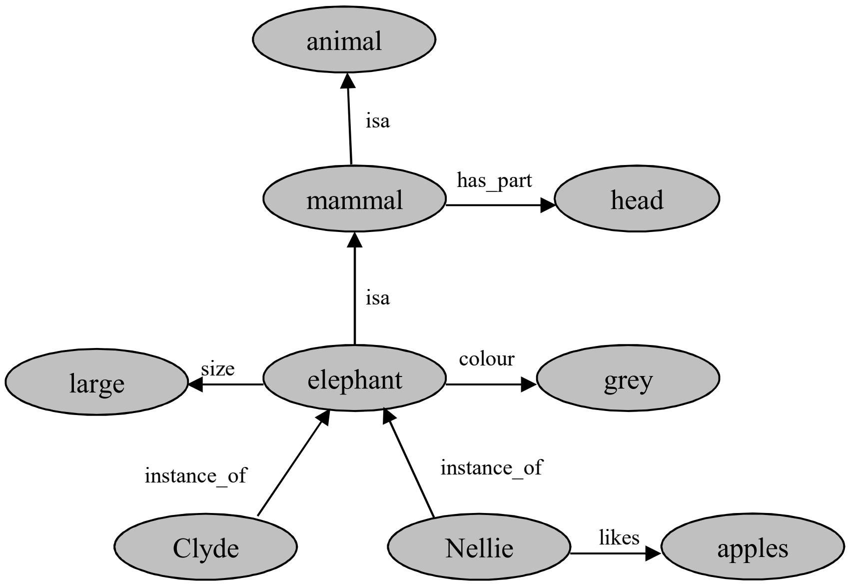
- **Drawbacks** of semantic networks compared to FOL
  - Can only express binary relationships
    - e.g. $Fly(Shankar, NewYork, NewDelhi, Monday)$ cannot be directly represented. ➔ reification
  - Negation, disjunction, nested function symbols, and existential quantification are missing
- Extendable using procedural attachments
- Represents default values – assertions may be overridden by more specific values
- Semantic network representing $Fly(Shankar, NewYork, NewDelhi, Yesterday)$

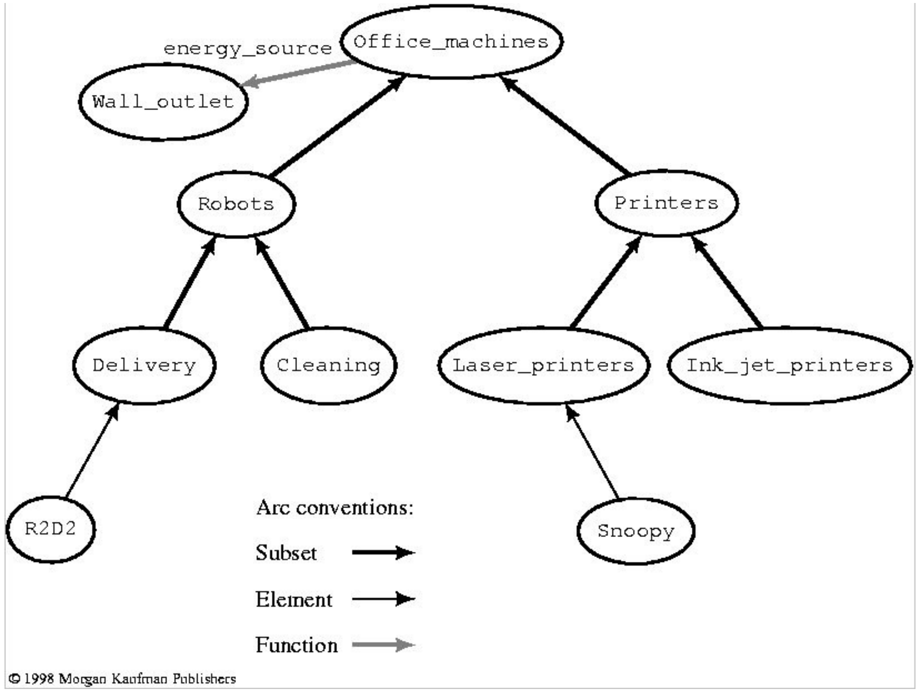


출처: Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4rd Edition). Pearson

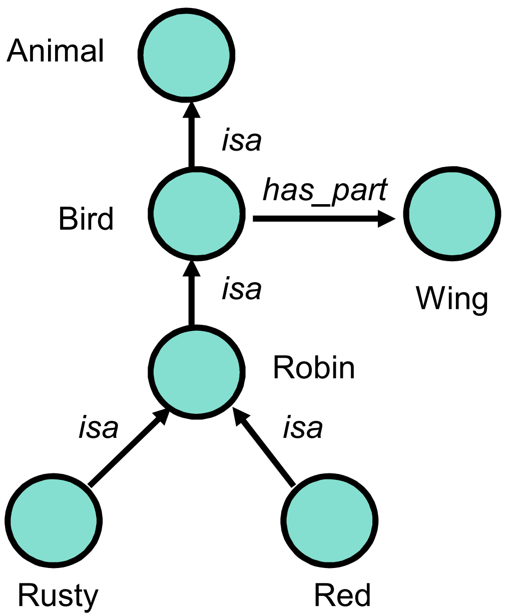
Semantic Networks

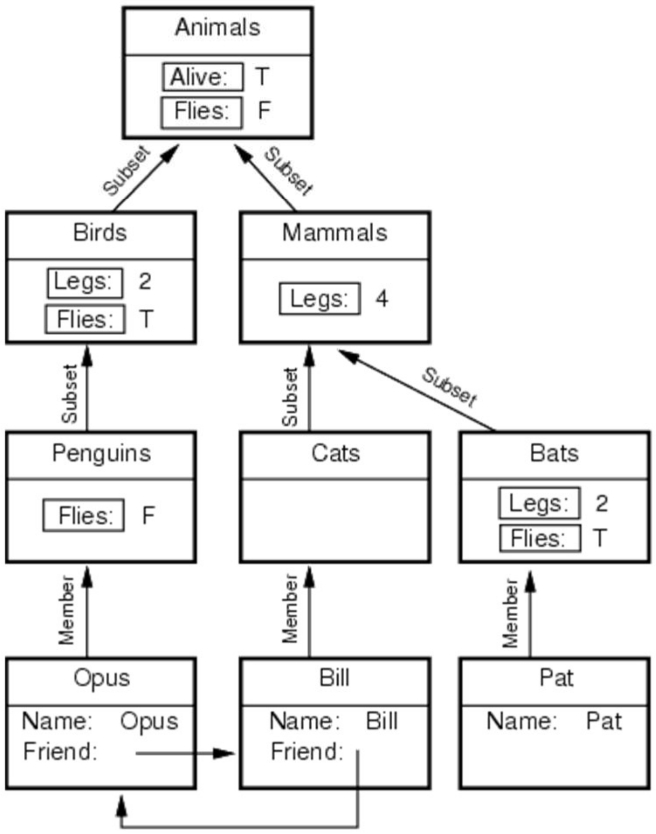
- 의미망 (Semantic Networks)





Frames & FOL





(a) A frame–based knowledge base

Rel(Alive,Animals,T)
Rel(Flies,Animals,F)

Birds $\subset$ Animals
Mammals $\subset$ Animals

Rel(Flies,Birds,T)
Rel(Legs,Birds,2)
Rel(Legs,Mammals,4)

Penguins $\subset$ Birds
Cats $\subset$ Mammals
Bats $\subset$ Mammals

Rel(Flies,Penguins,F)
Rel(Legs,Bats,2)
Rel(Flies,Bats,T)

Opus $\in$ Penguins
Bill $\in$ Cats
Pat $\in$ Bats

Name(Opus,"Opus")
Name(Bill,"Bill")
Friend(Opus,Bill)
Friend(Bill,Opus)
Name(Pat,"Pat")

(b) Translation into first–order logic

8.5 Reasoning Systems for Categories (4/6)

Description Logics

- Notations to make it easier to describe definitions and properties of categories
  - FOL describes things about objects.
  - Evolved from semantic networks to formalize what the networks mean.
- **Principal inference tasks** for description logics
  - Subsumption: Determine if one category is a subset of another
  - Classification: Determine the category in which an object belongs
  - Consistency: Determine if membership criteria are logically satisfiable

8.5 Reasoning Systems for Categories (5/6)

Description Logics

출처: Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4rd Edition). Pearson

- CLASSIC language: typical description logic
- Syntax

$$
\begin{array}{rcl}
\textit{Concept} & \rightarrow & \textbf{Thing} \mid \textit{ConceptName} \\
& \mid & \textbf{And}(\textit{Concept},\ldots) \\
& \mid & \textbf{All}(\textit{RoleName},\textit{Concept}) \\
& \mid & \textbf{AtLeast}(\textit{Integer},\textit{RoleName}) \\
& \mid & \textbf{AtMost}(\textit{Integer},\textit{RoleName}) \\
& \mid & \textbf{Fills}(\textit{RoleName},\textit{IndividualName},\ldots) \\
& \mid & \textbf{SameAs}(\textit{Path},\textit{Path}) \\
& \mid & \textbf{OneOf}(\textit{IndividualName},\ldots) \\
\textit{Path} & \rightarrow & [\textit{RoleName},\ldots] \\
\textit{ConceptName} & \rightarrow & \textit{Adult} \mid \textit{Female} \mid \textit{Male} \mid \ldots \\
\textit{RoleName} & \rightarrow & \textit{Spouse} \mid \textit{Daughter} \mid \textit{Son} \mid \ldots
\end{array}
$$

Figure 10.6 The syntax of descriptions in a subset of the CLASSIC language.

- e.g. bachelors are unmarried adult males: $Bachelor = And(Unmarried, Adult, Male)$
  - In first-order logic (FOL), $Bachelor(x) \Leftrightarrow Unmarried(x) \land Adult(x) \land Male(x)$

8.5 Reasoning Systems for Categories (6/6)

Description Logics (DL)

- Emphasis on tractability of inference
- Inference happens by
  - Describing the problem instance
  - Asserting the instance into the KB to be handled by the subsumption apparatus
- FOL cannot predict solution time
- DL solve in time polynomial in size of KB
- DLs usually lack *disjunction* and *negation* (for time/speed considerations)

8.6 Reasoning with Default Information

8.6 Reasoning with Default Information (1/4)

Circumscription

- **Two nonmonotonic logics**: circumscription and default logic
- Circumscription – more precise version of closed-world assumption
  - Closed world: Information provided is assumed complete, therefore ground sentences not asserted to be true are assumed false
  - Specifying particular predicates that are assumed to be "as false as possible"
  - Default rule stating that birds fly:

$$Bird(x) \land \neg Abnormal_1(x) \Rightarrow Flies(x)$$

  - $Abnormal_1()$ is circumscribed
  - Circumspection is an example of **model preference** logic; a sentence is entailed with default status if it is true in all *preferred* models in KB

8.6 Reasoning with Default Information (2/4)

Default logic

- Default rules express contingencies:
  - $Bird(x)\colon Flies(x)/Flies(x)$
  - If $Bird(x)$ is true, and $Flies(x)$ consistent with KB, then conclude $Flies(x)$ by default
- Default rule form
  - $P : J_1, \dots, J_n/C$
  - P is prerequisite; C is conclusion; and J_i are the justifications
  - If any J_i is false, then C is not true. Any variable that appears in J_i or C must also appear in P.

Nixon-diamond example: represented in default logic with one fact and two default rules

$$Republican(Nixon) \land Quaker(Nixon)$$

$$Republican(x)\colon \neg Pacifist(x)/\neg Pacifist(x)$$

$$Quaker(x)\colon Pacifist(x)/Pacifist(x)$$

8.6 Reasoning with Default Information (3/4)

Truth maintenance systems

- Designed to handle belief revision:
  - Let's say our KB contains sentence P
  - But P is found to be incorrect/untrue, so we want to say TELL($KB, \neg P$)
  - First, need RETRACT(KB,P)
- Justification-based truth maintenance system (**JTMS**)
  - Each sentence in KB is annotated with a justification
  - Justification consisting of the set of sentences from which it was inferred
  - Justifications make retraction efficient
  - Sentences are in or out, based on truth value of supporting sentences

8.6 Reasoning with Default Information (4/4)

Truth maintenance systems

- **Assumption-based** truth maintenance system (**ATMS**)
  - Designed to make belief revision efficient
  - Represents all states at the same time
  - Maintains a set of supporting sentences, representing all states
  - Sentence holds when all assumptions in one of its assumptions sets hold
- Explanation
  - TMS provides a mechanism for generating explanations
  - An explanation of a sentence P is a set of sentences E such that E entails P
  - **Assumptions**: sentences that are not known to be true, but would suffice to prove P if they were true
  - Explanation might include assumptions

Summary

1. A general-purpose ontology needs to cover a wide variety of knowledge and should be capable, in principle, of handling any domain.
2. We presented an upper ontology based on categories and the event calculus.
3. Actions, events, and time can be represented with the event calculus.
4. Special-purpose representation systems, such as semantic networks and description logics, have been devised to help in organizing a hierarchy of categories.
5. Inheritance is an important form of inference, allowing the properties of objects to be deduced from their membership in categories.
6. The closed-world assumption, as implemented in logic programs, provides a simple way to avoid having to specify lots of negative information.
7. Nonmonotonic logics, such as circumscription and default logic, are intended to capture default reasoning in general.
8. Truth maintenance systems handle knowledge updates and revisions efficiently.