Overview

I Artificial Intelligence

II Problem Solving

III Knowledge, Reasoning, Planning

- 7. Logical Agents
- 8. First-Order Logic
- 9. Inference in First-Order Logic
- **10. Knowledge Representation**
- 11. Automated Planning

IV Uncertain Knowledge and Reasoning

V Machine Learning

VI Communicating, Perceiving, and Acting

VII Conclusions

Knowledge Representation

- Introduction
- Ontological Reasoning
- Categories and Objects
- Events
- Mental Objects and Modal Logic
- Reasoning Systems for Categories
- Reasoning with Default Information

Introduction to Knowledge Representation (1)

- Every AI program needs a knowledge representation (KR)
- Ontologies encode knowledge about a domain and thus implies a KR
  - In informatics: Set of terms and relations in a domain, e.g. object-oriented programming
  - Usually, KRs are tailored for the purpose of the program
    - Nevertheless, the choices are difficult and important for the success
    - For important domains like medicine there exist special ontologies
      - and much debate and standardization efforts …

Introduction to Knowledge Representation (2)

- Is it possible to design a general ontology?
  - Maybe: very complex
    - Must deal with Categories, Objects, Events, Time, Physical Objects, Beliefs, …
  - Diverse efforts with limited success
  - „Practical“ Ontologies like DBpedia or Linked Open Data have much knowledge and a simple knowledge representation
    - Popular practical knowledge representations:
      - Semantic networks,
      - Description logic,
      - With concrete KR-languages like RDF, OWL, Classic, …

Some Terminology

Different systems for knowledge representation can be ordered to their representation power:

- **Catalogue, glossar, taxonomy:** simple controlled vocabularies
- **Classification, thesaurus, (taxonomy):** includes hierarchical relations, synonyms
- **Semantic net, ontology, knowledge graph:** includes other types of relations in addition to hierarchical relations
  - includes often inheritance of properties
- **Axiom system, predicate logic:** Semantic net with formal semantics

However, the terms are often used quite freely in the literature and it is necessary to have a closer look on the underlying concepts.

For many tasks like e.g. Natural Language Understanding such (large) repositories are central!

Medical Ontologies

- Important for communication, documentation, billing, research, …
- Relevant entities and popular hierarchical ontologies:
  - Diagnoses (ICD)
    - Diagnoses Billing Groups (DRG)
  - Operations (including surgeries) and procedures (OPS)
  - Drugs (ATC)
  - Laboratory data (LOINC)
  - Radiology (RADLEX) (and other image domains)
- Generalization: SNOMED: Compositional ontology
  - more complex and more expressive; not in widespread use
- Usage for diagnoses enforced by law; for other entities only partially in practical use
  - Necessary for research with „big data“: Standardization and coding effort very high

Example for Medical Ontology: ICD-10-Code (hierarchical)

„Third degree burn of left forefinger caused by hot water."

Übersicht über die Kapitel

Kapitel	Gliederung	Titel
I	A00-B99	Bestimmte infektiöse und parasitä
II	C00-D48	Neubildungen
III	D50-D90	Krankheiten des Blutes und der bl des Immunsystems
IV	E00-E90	Endokrine, Ernährungs- und Stoffw
V	F00-F99	Psychische und Verhaltensstörung
VI	G00-G99	Krankheiten des Nervensystems
VII	H00-H59	Krankheiten des Auges und der Au
VIII	H60-H95	Krankheiten des Ohres und des W
IX	I00-I99	Krankheiten des Kreislaufsystems
X	J00-J99	Krankheiten des Atmungssystems
XI	K00-K93	Krankheiten des Verdauungssyste
XII	L00-L99	Krankheiten der Haut und der Unt
XIII	M00-M99	Krankheiten des Muskel-Skelett-Sy
XIV	N00-N99	Krankheiten des Urogenitalsystem
XV	O00-O99	Schwangerschaft, Geburt und Wo
XVI	P00-P96	Bestimmte Zustände, die ihren Urs
XVII	Q00-Q99	Angeborene Fehlbildungen, Defor
XVIII	R00-R99	Symptome und abnorme klinische
XIX	S00-T98	Verletzungen, Vergiftungen und be
XX	V01-Y84	Äußere Ursachen von Morbidität u
XXI	Z00-Z99	Faktoren, die den Gesundheitszus Gesundheitswesens führen
XXII	U00-U99	Schlüsselnummern für besondere

Kapitel XIX
Verletzungen, Vergiftungen und
(S00-T98)

Dieses Kapitel gliedert sich in folgende

S00-S09	Verletzungen des Kopfes		
S10-S19	Verletzungen des Halses		
S20-S29	Verletzungen des Thorax		
S30-S39	Verletzungen des Abdomens, d		
S40-S49	Verletzungen der Schulter und		
S50-S59	Verletzungen des Ellenbogens		
S60-S69	Verletzungen des Handgelenke		
S70-S79	Verletzungen der Hüfte und de		
S80-S89	Verletzungen des Knies und de		
S90-S99	Verletzungen der Knöchelregion		
T00-T07	Verletzungen mit Beteiligung m		
T08-T14	Verletzungen nicht näher bezei		
T15-T19	Folgen des Eindringens eines F		
T20-T32	Verbrennungen oder Verätzung		
		T20-T25	Verbrennungen oder Verätzungen der äußeren Körperoberfläche, Lokalisation bezeichnet
		T26-T28	Verbrennungen oder Verätzungen, die auf das Auge und auf innere Organe begrenzt sind
		T29-T32	Verbrennungen oder Verätzungen mehrerer und nicht näher bezeichneter Körperregionen
T33-T35	Erfrierungen		
T36-T50	Vergiftungen durch Arzneimittel, Drogen und biologisch aktive Substanzen		
T51-T65	Toxische Wirkungen von vorwiegend nicht medizinisch verwendeten Substanzen		
T66-T78	Sonstige und nicht näher bezeichnete Schäden durch äußere Ursachen		
T79-T79	Bestimmte Frühkomplikationen eines Traumas		
T80-T88	Komplikationen bei chirurgischen Eingriffen und medizinischer Behandlung, anderenorts nicht klassifiziert		
T89-T89	Sonstige Komplikationen eines Traumas, anderenorts nicht klassifiziert		
T90-T98	Folgen von Verletzungen, Vergiftungen und sonstigen Auswirkungen äußerer Ursachen		

T23.-	**Verbrennung oder Verätzung des Handgelenkes und der Hand**
	Inkl.: Daumen (-Nagel) Finger (-Nagel) Handfläche
T23.0	**Verbrennung nicht näher bezeichneten Grades des Handgelenkes und der Hand**
T23.1	**Verbrennung 1. Grades des Handgelenkes und der Hand**
T23.2-	**Verbrennung 2. Grades des Handgelenkes und der Hand**
T23.20	Verbrennung Grad 2a des Handgelenkes und der Hand Verbrennung nicht näher bezeichneten 2. Grades des Handgelenkes und der Hand
T23.21	Verbrennung Grad 2b des Handgelenkes und der Hand
T23.3	**Verbrennung 3. Grades des Handgelenkes und der Hand**
T23.4	**Verätzung nicht näher bezeichneten Grades des Handgelenkes und der Hand**
T23.5	**Verätzung 1. Grades des Handgelenkes und der Hand**
T23.6-	**Verätzung 2. Grades des Handgelenkes und der Hand**
T23.60	Verätzung Grad 2a des Handgelenkes und der Hand Verätzung nicht näher bezeichneten 2. Grades des Handgelenkes und der Hand
T23.61	Verätzung Grad 2b des Handgelenkes und der Hand
T23.7	**Verätzung 3. Grades des Handgelenkes und der Hand**

Example for Medical Ontolgy: SNOMED (Compositional)

„Third degree burn of left forefinger caused by hot water.“

64572001 | disease | :
  246075003 | causative agent | = 47448006 | hot water | ,
  363698007 | finding site | = (
    83738005 | index finger structure | :
    272741003 | laterality | = 7771000 | left |
) ,
  {
   116676008 | associated morphology | = 80247002 | third degree burn injury | ,
   363698007 | finding site | = 39937001 | skin structure |
  }

Components:
Causative Agent
Finding Site
Morphology

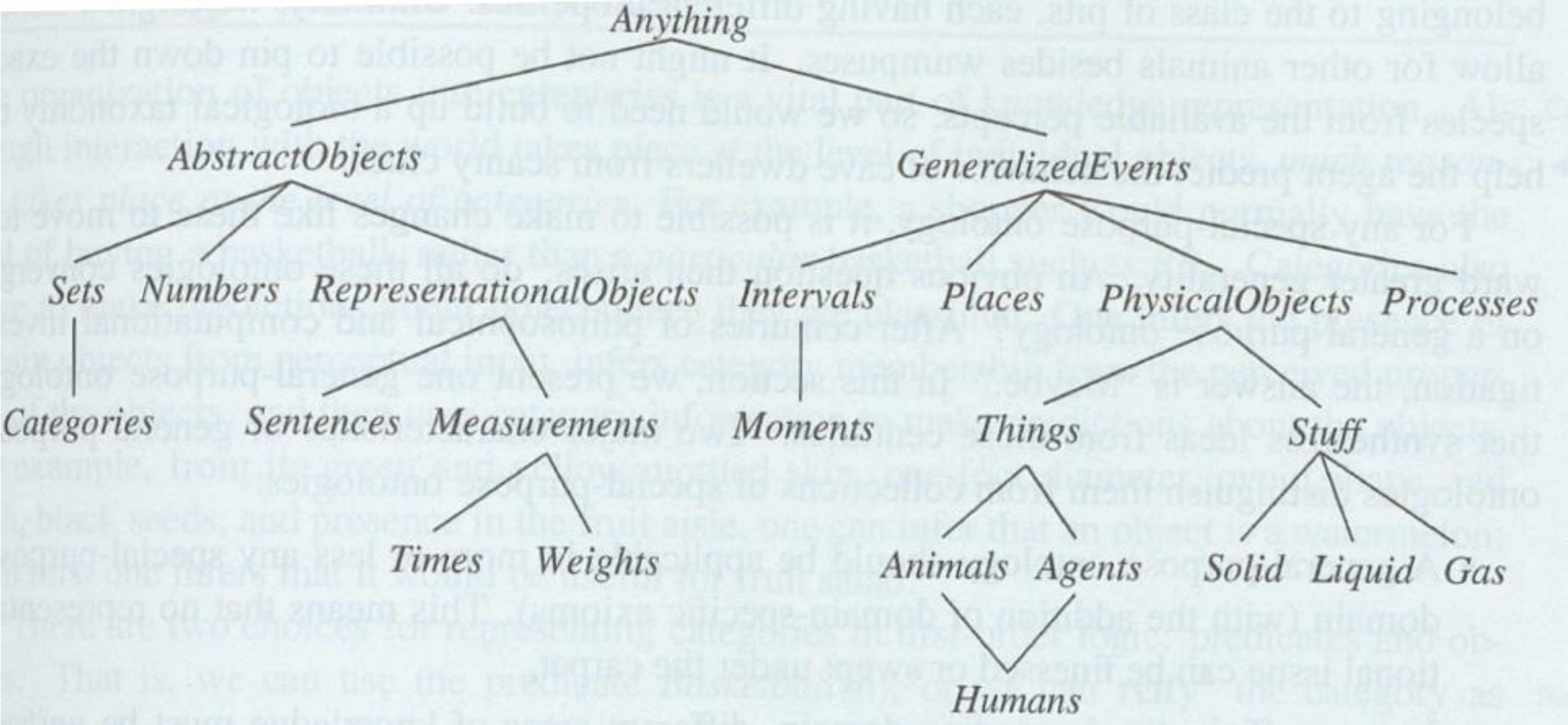
Legend: concept, attribute, value

Universal Ontology

- Human language can describe everything
- Most programs have a specialized goal and therefore need only a domain specific ontology
- For general intelligence, an universal ontology seems necessary
- Basic assumptions:
  - Must be able to express „everything“
    - Specifies a scheme (upper ontology), in which details can be filled in as necessary
    - Must deal with typical things and relations, but also with infinite exceptions
      - Simplification: Use (subset of) first-order logic
      - For exceptions, many approaches: Probability und utility theory, default logic, distinction between typical and other instances etc.
    - Inferences should be efficient:
      - „Practical“ general ontologies (much simpler)

Upper Ontology of the World

- Main concept ist „Categories“ organizing objects (distinct from „PhysicalObjects“)



Categories and Objects (1)

The core of every ontology is to categorize objects and to describe properties for the categories

- Often categories form a taxonomic hierarchy (taxonomies) with inheritance
  - Specialized categories inherit properties from more general categories
  - Popular taxonomies
    - for living things (biology, > 10 million organisms)
    - for books and articles in libraries: Dewey classification
    - for commercial products (e.g. used by Amazon)
    - for medicine (see above)
    - …

Dewey Decimal Classification: top level
000 – Computer science, information & general works
100 – Philosophy & psychology
200 – Religion
300 – Social sciences
400 – Language
500 – Pure Science
600 – Technology
700 – Arts & recreation
800 – Literature
900 – History & geography

Categories and Objects (2)

- First-order logic makes it easy to state facts about categories
  - Two choices (equivalent):
    - Use of a predicate, e.g. Basketball (b)
    - Reification of the category as an object as $b \in Basketballs$
- Examples of statements about facts about categories:
  - An object is a member of a category: $BB9 \in Basketballs$
  - A category is a subclass of another category: $Basketballs \subset Balls$
  - All members of a category have some properties: $(x \in Basketballs) \Rightarrow Spherical\ (x)$
  - Members of a category can be recognized by properties:

    $$Orange\ (x) \land Round\ (x) \land Diameter(x) = 24\ cm \land x \in Balls \Rightarrow x \in Basketballs$$
  - A category as a whole has some properties: $Dogs \in DomesticatedSpecies$
- Categories can be defined by conditions of membership
  - e.g. $x \in Bachelors \Leftrightarrow Unmarried\ (x) \land x \in Adults \land x \in Males$

Decomposition

- Other relations between categories (beyond subclass and membership)
  - Disjointness (sets having no members in common)
    - e.g. Disjoint {Animals, Plants}
  - Exhaustive decomposition (each member of a class must be in at least one subclass)
    - e.g. ExhaustiveDecomposition ({Americans, Canadians, Mexicans}, NorthAmericans)
  - Partition (Exhaustive decompostion of disjoint sets)
    - e.g. ({Animals, Plants, Fungi, Protista, Monera}, LivingThings)

Physical Decomposition

- Objects can be part of other objects, e.g.
  - PartOf (Bucharest, Romania)
  - PartOf (Romania, EasternEurope)
  - PartOf (EasternEurope, Europe)
  - PartOf (Europe, Earth)
- **PartOf relation** is transitive and reflexive (similar to subclass relation), i.e.:
  - $PartOf(x,y) \wedge PartOf(y,z) \Rightarrow PartOf(x,z)$
  - $PartOf(x,x)$
- Categories of **composite objects** are often defined by structural relations among parts, e.g.
  - $Biped(a) \Rightarrow \exists\, l_1, l_2, b\ Leg(l_1) \wedge leg(l_2) \wedge body(b) \wedge PartOf(l_1,a) \wedge PartOf(l_2,a) \wedge PartOf(b,a)$ $Attached(l_1,b) \wedge Attached(l_2,b) \wedge l_1 \neq l_2 \wedge [\forall\, l_3\ Leg(l_3) \wedge PartOf(l_3,a) \Rightarrow (l_3=l_1) \vee (l_3=l_2)]$
  - The notation „exactly two“ is awkward; in description logic, cardinalities are used.

Measurements

- Introduction of „measure objects“ like length, price, weight etc.
- Representation of measure objects with a unit function having a number as argument
  - e.g. Length (L_1) = Inches (1,5) = Centimeter (3,81)
  - Conversion: Centimeters (2,54 x d) = Inches (d)
  - Diameter ($Basketball_{12}$) = Centimeter (24)
  - Price ($Basketball_{12}$) = Euro (19)
- **Qualitative Measurements**
  - Often, only qualitative statements are possible, e.g. about difficulty of tasks, beauty etc.
    - Ordering of qualitative statements possible and important:
      - e.g.: Difficulty ($task_1$) > Difficulty ($task_2$)

Objects: Things and Stuff

- Distinction between Things and Stuff
  - **Things:** Individual objects, like an apple („count nouns“)
    - If cut in half, it is a not longer an apple, but different object (half of the apple)
  - **Stuff:** No indidual object, like butter („mass nouns“)
    - If cut in half, it is still butter
      - However a „pound of butter“ would be an object
- Intrinsic and extrinsic properties:
  - Intrinsic properties remain the same under subdivision (e.g. if an object is cut in half)
    - e.g. edibable (x), melting point (x), flavor (x), etc.
  - Extrinsic properties change under subdivision
    - e.g. weight (x), length (x), shape (x), etc.

➢ Stuff has intrinsic properties only, things have both intrinsic and extrinsic properties

Events

- It is much more easy to represent static categories and objects than dynamic events
  - Many simplifications of events (actions)
    - e.g. in Wumpus world: Actions in discrete time steps, instantaneous and not simultaneous, without variations (how an action is performed) etc.
  - In real world, much more complicated
    - e.g. continuous actions (like filling a bath tube): How is the state during the action? What happens, if other action occur simulateneously?
    - New approach necessary: **Event calculus** (instead of situation calculus in Wumpus world)

Event Calculus (1)

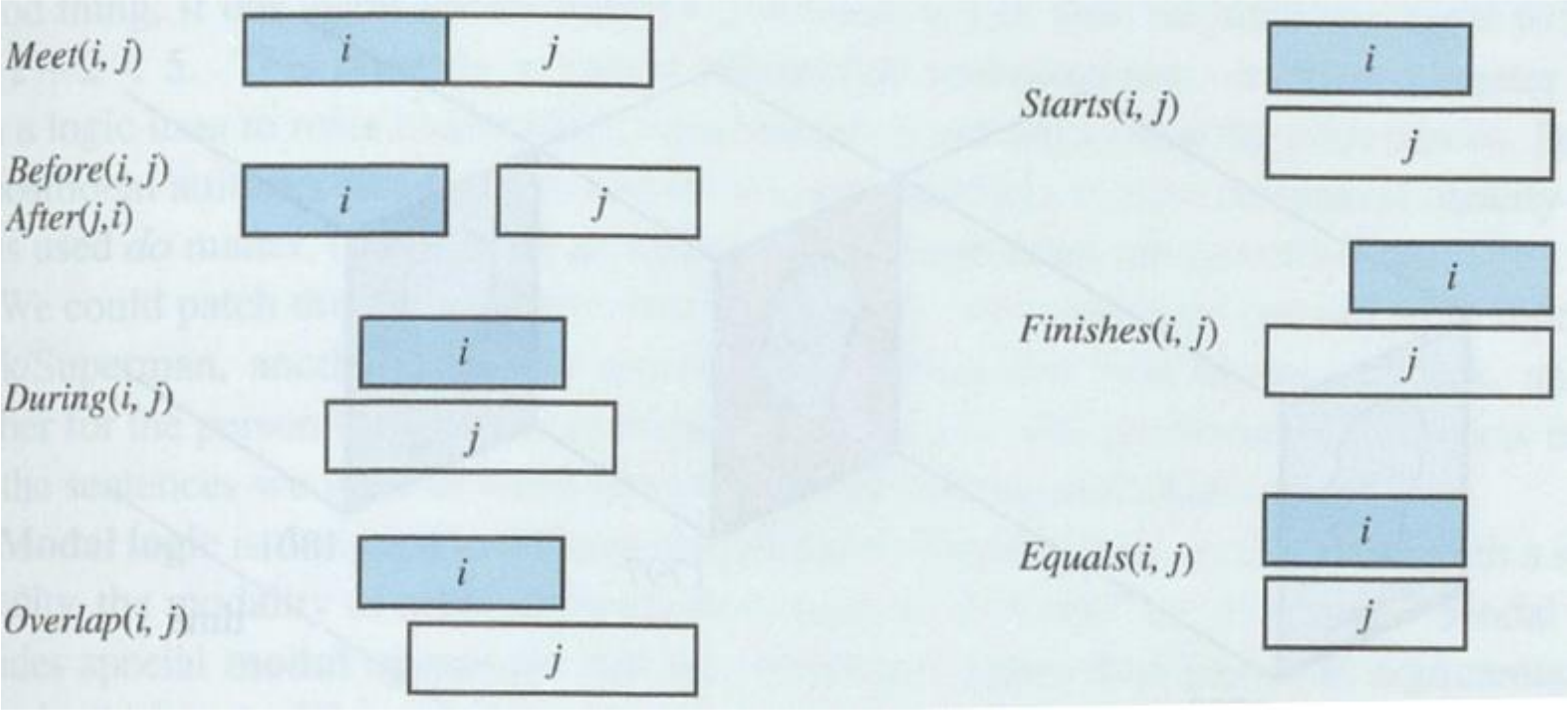
- Deals with events, fluents and time points.
  - By reify events (e.g. flyings) arbitrary statements about the event are possible
  - e.g. $E_1 \in$ Flying $\wedge$ Flyer (E_1,Shankar) $\wedge$ Origin (E_1,SF) $\wedge$ Destination (E_1,DC)
    - Where Flying is a category of all flying events
  - Events change something, e.g. the location of Shankar, which therefore must be a fluent
    - e.g. At (Shankar, Berkely) is a fluent

Event Calculus (2)

- We need predicates, what is true for fluents and how events change fluents:
  - $T(f, t_1, t_2)$ Fluent f ist true between t_1 and t_2
  - Happens (e, t_1, t_2) Event e start 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
- Example description of a flying event:
  - $E = \text{Flyings}(a, here, there) \land \text{Happens}(E, t_1, t_2) \Rightarrow$
    $\text{Terminates}(E, \text{At}(a\ here), t_1) \land \text{Initiates}(E, \text{AT}(a, there), t_2)$
  - Further axioms necessary

Time

- We can deal with time quantitatively and qualitatively
  - Quantitative time: Time points (without duration) and time intervals (with duration)
  - Qualitative time: Time intervals with relations



Fluents and Objekts

- Physical objects can ve viewed as generalized events in time and space
  - USA: Event, that began 1776 with changing properties (fluents) for number of states, population etc.
  - How should the president of the USA be formalized, since it changes every 4 or 8 years?
    - As a single object, that consists of different people at different times!
    - T (Equals (President (USA), George Washington), Begin (AD1789), End (AD1797)
      - New function symbol „Equals“, because the predicate „=“ is not allowed as argument for another predicate („T“) in first-order logic

Mental Objects (1)

- How can we represent, what agents know about their own knowledge und about the knowledge of other?
  - If Alice asks Bob „What is the square root of 1764?“, Bob might answer „I don‘t know“.
    - If Alices insists „Please tell me!“, Bob should realize, that with some more thought, he can indeed answer the question.
  - If Alice asks „Is the president in the Oval Office right now?“, Bob should know, that he cannot answer the question, independently of his effort.

Mental Objects (2)

- Reasoning about what other agents believes, knows, wants, informs is even more important.
  - Difficult problem, because logical omnipotence cannot be assumed and even contradictions are possible.
  - Knows (Lois, CanFly (Superman)) – Lois knows, that Superman can fly
    - Technical problem: Sentence is in second-order logic, but that can be fixed by reification
    - Serious problem: Since Clark ist Superman, Lois should know, that Clark can fly:

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

      - but probably Lois does not know, that Clark is Superman.
  - Logic has **referential transparency**, i.e infers what can be infered, but that might lead to wrong conclusions; we need something like **referential opacity**.
  - Possible Solution: **Modal Logic**

Modal Logic

- Idea: There are several possible worlds. Each should be internally consistent.
  - Includes special modal operators with sentences as arguments
    - e.g. „A knows P“ is represented: $K_A P$ (K = modal operator for Knowledge with 2 arguments: the agent A and the sentence P)
      - Sentences can also formed with modal operator (eg. A knows that B knows P)
    - From the modal operators, several possible worlds are constructed (e.g. that superman is Clark or isn‘t), which are connected by accessibility relations.
    - Contrast: Regular logic is concerned with a single modality (of truth)
- Example for modal logic: Lois does not knows, whether Clark is Superman or not:
  - $K_{Lois}[K_{Clark}\mathrm{Identy(Superman, Clark)} \vee K_{Clark}\neg\mathrm{Identy(Superman, Clark)}]$

Problems with Modal Logic

- Modal logic can maintain several internally consistent models simulataneously
  - It assumes logical omniscience, which is psychologically questionable
    - If an agent knows some axioms, it knows all consequences
  - Every uncertainty implies a new model branch – may increase exponentially
  - It says nothing about preferences for different models

Reasoning Systems for Categories

- Categories are the core of large-scale knowledge systems
  - Two specialized approaches for reasoning with categories
    - Semantic networks
    - Description logic

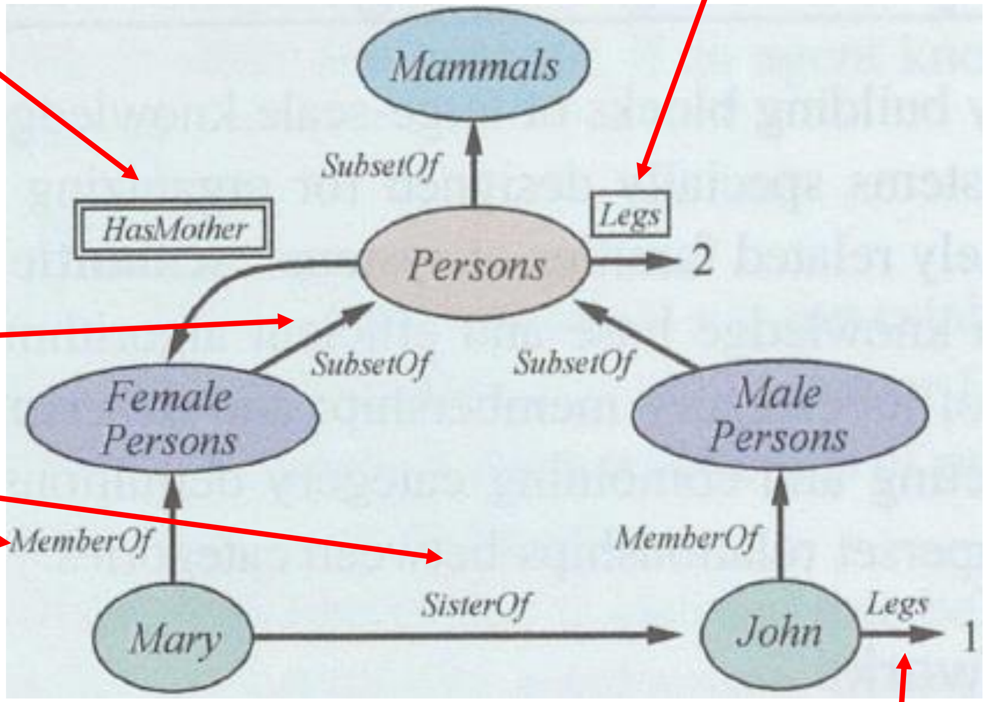
Semantic Networks (1)

- Core (many variants):
  - Individual objects
  - Category of objects
  - Relations among objects and categories
    - Inheritance

- Graphical presentation of a restricted part of logic
  - *SubsetOf* link: FemalePersons $\subset$ Persons
  - *SisterOf* link: SisterOf (Mary, John)
  - *MemberOf* link: Mary $\in$ FemalePersons

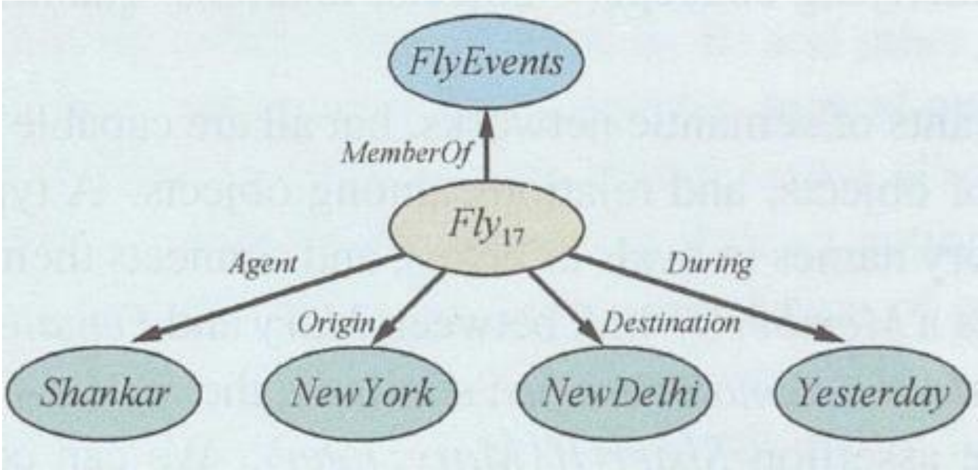
- Simple HasMother-link from Person to FemalePerson would be illegal because hasMother is defined on instances, not on categories (see above right)

- *Legs* boxed Link: $\forall$ x $\in$ Persons $\Rightarrow$ Legs (x,2)
- *HasMother* double-boxed link:
  $\forall$ x $\in$ Persons $\Rightarrow$ [$\forall$ y HasMother(x,y)
  $\Rightarrow$ y $\in$ FemalePersons]



Inheritance: Legs (Mary, 2), but Legs (John, 1)

Semantic Networks (2)



- Semantic networks allow binary relations only
  - For **n-ary relations** reification is necessary
- **Negation, disjunction, nested function symbols and existential quantifier are missing**
  - Extensions possible, but would negate the simplicity and transparency of the inference process
- Extensions of expressive power are usually compensated by **procedural attachment**
  - Assertions or queries are interpreted by special code
- Inheritance (see last slide) allows to represent **default value**, which in pure logic would be a contradiction

Description Logics

- Notations evolved from semantic networks
  - Clearer definitions (in axioms instead of graphs)
  - Main tasks:
    - Supsumption (checking if one category is a subset of another)
    - Classification (checking, in an object belongs to a category)
    - Consistency (are membership criteriy logically satisfiable)
  - Retaining the underlying simplicity
    - Lack of disjunction and negation
      - Or very limited use of disjunction in Fills and OneOf constructs (in CLASSIC), which allows disjunction over explicitely enumerated objects, but not over descriptions

Examples for Description Logic

- Syntax of descriptions in a subset of the Classic language (left)

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

- Example definitions:
  - Unmarried adult males:

  Bachelor = And (Unmarried, Adult, Male)

  - Set of all man with at least three sons, who are athletic and married to doctors and at most two daughters, who are teachers in math or informatics:

And (Man (AtLeast (3, Son), AtMost (2 Daughter),
All (Son, And (Athletic, Married, All(Spouse, Doctor))),
All (Daughter, And (Teacher, Fills (Subject, Math, Informatics))))

Reasoning with Default Information

- Key concept in the real world: „No rule without exception!“
- Semantic Networks used default information with a procedural interpretation
- Closed Word Assumption can also override information (i.e. exhibit **nonmonotonicity**):
  - If a proposition α is not mentioned in a KB, then $KB \models \neg\alpha$, but $KB \wedge \alpha \models \alpha$
- Theoretical Foundation of reasoning with default information
  - Circumscription
  - Default Logic

Circumcription

- Extension of closed-world assumption
- A statement is true, if one cannot infer the opposite
- Example: All bird can fly, except we can prove, that a bird is abnormal
  - $\text{Bird}(x) \land \neg\text{Abnormal}_1(x) \Rightarrow \text{Flies}(x)$
  - Abnormal_1 is circumscribed: a reasoner may assume $\neg\text{Abnormal}_1(x)$ unless $\text{Abnormal}_1(x)$ is known to be true
- Dealing with multiple inheritance and conflicting default rules:
  - Example: Nixon diamond:
    - $\text{Republican}(\text{Nixon}) \land \text{Quaker}(\text{Nixon})$
    - $\text{Republican}(x) \land \neg\text{Abnormal}_2(x) \Rightarrow \neg\text{Pacifist}(x)$
    - $\text{Quaker}(x) \land \neg\text{Abnormal}_3(x) \Rightarrow \text{Pacifist}(x)$

  ➢ Circumscriptive reasoner remains agnostic, whether Nixon is a pacifist.

Default Logic

- Similar to circumscription, but without predicate „abnormal“
- Example of a default rule:
  - Bird (x): Flies (x) / Flies (x)
    - meaning if x is a bird and Flies (x) is consistent with the KB, Flies (x) can be concluded by default
  - General form of default rules:
    - P: J_1, … , J_n / C
      - P: prerequisite, C: conclusion, J_i justifications (if any one of the them can be proven to be false, the conclusion cannot be drawn)
      - In above example: P = Bird (x), J_1 = Flies (x), C = Flies (x)
  - Nixon diamond:
    - Republican (Nixon) $\wedge$ Quaker (Nixon)
    - Republican (x): $\wedge$ $\neg$Pacifist (x) / $\neg$Pacifist(x)
    - Quaker(x) $\wedge$ Pacifist (x) / Pacifist(x)

Default Decisions in the Real World

- Preference of default rules possible (e.g. for Nixon diamond)
- What are the consequences of default rules for actions
  - e.g. „Brakes are o.k“?
    - in a normal situation
    - in a situation drivinig steep mountain roads
- In difficult situations, probability and utility theory are necessary

Truth Maintenance Systems

- Inference systems with default information need an efficient **belief revision** mechanism to retract facts and also all consequences inferred from these fact.
- Simple Approach: recompute everything anew with the current facts
  - For large knowledge bases too inefficient
  - Slight improvement: Recompute everything, when the fact to be retracted was established – still inefficient
- Two approaches for efficient belief revision systems:
  - Justification-based truth maintenance system
  - Assumption-based truth maintenance system
- Basic idea: Each conclusion stores all justifications, why it is inferred
  - If all justifications vanish, the conclusion is retracted

Justification-based Truth Maintenance System

- Justifications: All sentences (e.g. rules) which were used to infer the concept
  - Useful for explanation component
  - Useful for efficient exploring the consequences of different assumptions

Assumption-based Truth Maintenance System

- Justification: the set of facts which were used to derive a conclusion
  - Each conclusion knows, under what assumptions it is valid
  - Even faster comparison of different assumptions
  - But less useful as explanation component