Lecture 32

CS 6380 Artificial Intelligence

Semantic Networks and Description Logics

Department of Computer Science and Engineering

Indian Institute of Technology Madras (IITM), India



Knowledge: Quantifiers and Axioms

- "Bond knows that someone is a spy" has two readings:

  $\exists x\ K_{Bond}\ Spy(x)$ a particular person
  $K_{Bond}\ \exists x\ Spy(x)$ just that spies exist

- Axioms for knowledge:

  $(K_a\ P \wedge K_a(P \Rightarrow Q)) \Rightarrow K_a\ Q \qquad K_a\ P \Rightarrow P \qquad K_a\ P \Rightarrow K_a(K_a\ P)$

- $K_a(P \vee \neg P)$ always holds, but $(K_a\ P) \vee (K_a\ \neg P)$ does not
- Problem: logical omniscience. Agents would know every consequence of what they know

Exercise 1: Who Knows What?

- Write each sentence in modal logic, using $\mathrm{K}_{\mathrm{Alfred}}$ and $\mathrm{K}_{\mathrm{Lois}}$
  a. Alfred knows that some student failed the quiz
  b. There is a student whom Alfred knows failed the quiz
  c. Alfred knows whether Madras is Chennai
  d. Alfred knows that Lois doesn't know whether Superman is Clark
- Does (a) imply (b)? Does (b) imply (a)?

Work in pairs

Exercise 1: Solutions

a. $K_{Alfred}\ \exists s\ Failed(s)$

b. $\exists s\ K_{Alfred}\ Failed(s)$

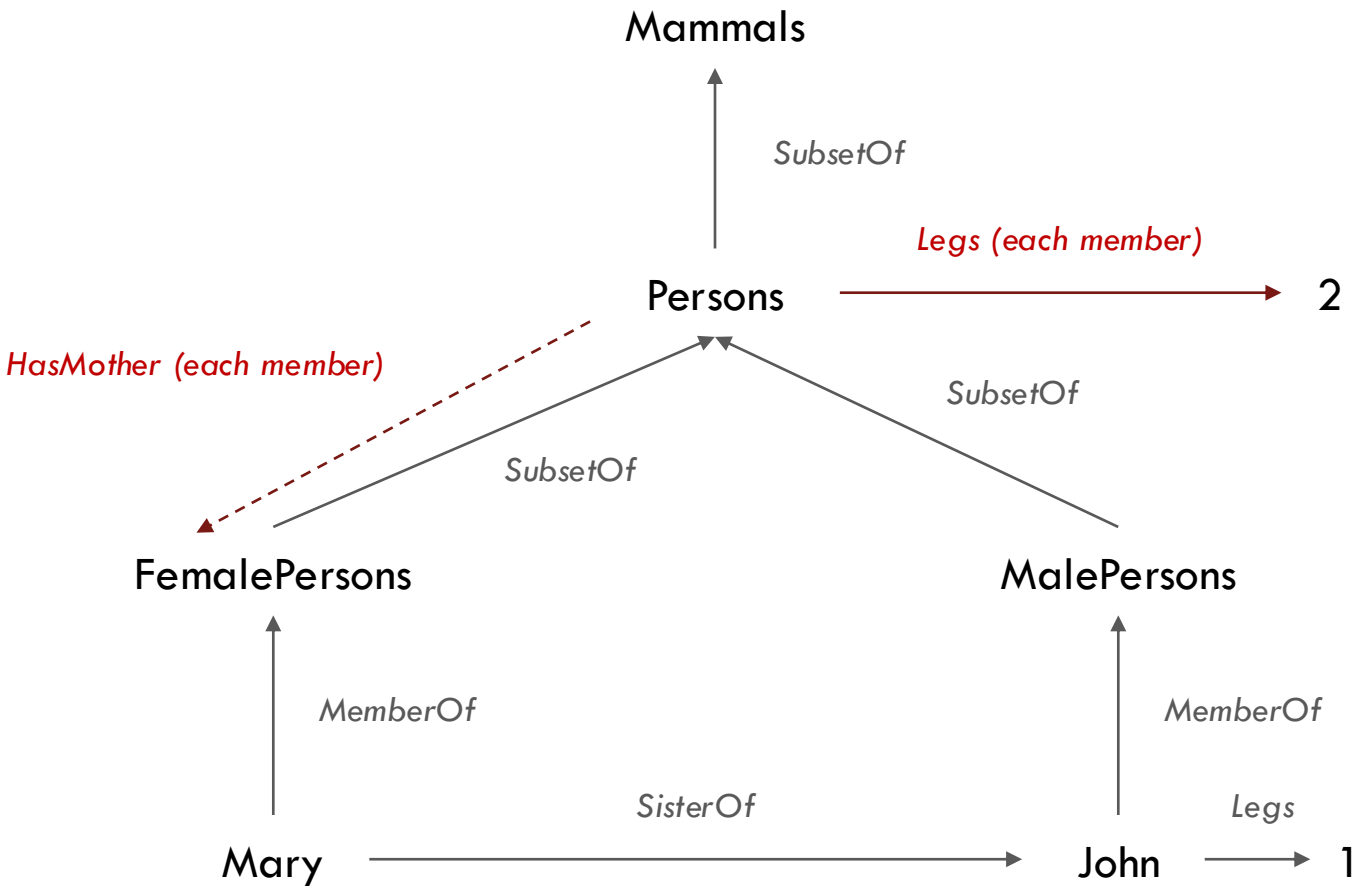
c. $K_{Alfred}\ (Madras = Chennai) \vee K_{Alfred}\ \neg(Madras = Chennai)$

d. $K_{Alfred}\ \neg(K_{Lois}\ I \vee K_{Lois}\ \neg I)$

- (b) implies (a): if one student failed in every world Alfred considers possible, then *some* student failed in each
- (a) does not imply (b): the failing student can differ from world to world
- "Knows whether" is a disjunction outside the K's, not $K(P \vee \neg P)$, which is always true

Semantic Networks

- Peirce (1909): existential graphs
- Nodes are objects and categories; labelled links are relations
- Same content as logic, friendlier notation
- "Mary is a female person and John's sister"



What the Links Mean

- Between objects and categories, links are simple facts:

  $\text{Mary} \in \text{FemalePersons}$ $\quad$ $\text{FemalePersons} \subset \text{Persons}$ $\quad$ $\text{SisterOf}(\text{Mary}, \text{John})$

- A link out of a category is about its members:

  $\forall x \; x \in \text{Persons} \Rightarrow \text{Legs}(x, 2)$
  $\forall x \; x \in \text{Persons} \Rightarrow [\forall y \; \text{HasMother}(x, y) \Rightarrow y \in \text{FemalePersons}]$

- Categories don't have legs or mothers!
  - Mixing these up broke early systems (McDermott, "AI Meets Natural Stupidity", 1976)
  - So did one `IsA` link for both "a cat is a mammal" and "Fifi is a cat"

Inheritance and Defaults

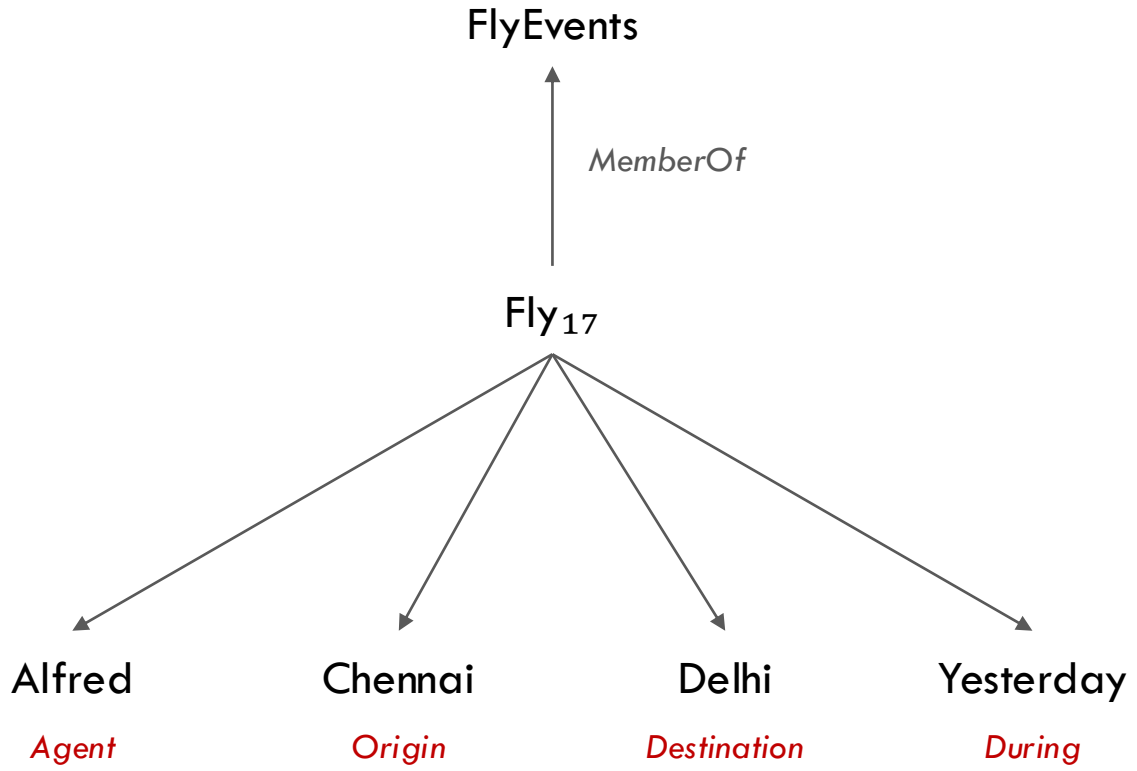
- How many legs does Mary have? Follow $\mathrm{MemberOf}$, then $\mathrm{SubsetOf}$ links up to a Legs link: 2
- John has his own Legs link: 1. The more specific value overrides the default
- A strictly logical version needs an explicit exception:

$$\forall x \; x \in \mathrm{Persons} \wedge x \neq \mathrm{John} \Rightarrow \mathrm{Legs}(x, 2)$$

  - Fine for a fixed network, awkward when new exceptions keep arriving
- Multiple inheritance can give conflicting answers; Java forbids it for classes
- Inference is fast and predictable: a big attraction over theorem proving

Limits of Semantic Networks

- Links are binary. To say Fly(Alfred, Chennai, Delhi, Yesterday), reify the event
- No negation, disjunction, nested functions or ∃
- Adding them gives full FOL, and loses the simple, predictable inference
- Procedural attachment: call special code for some relations



Description Logics

- Designed to define categories, with tractable inference
- Subsumption: is one category a subset of another?
- Classification: does an object belong to a category?
- In CLASSIC:

  $\text{Bachelor} = \text{And}(\text{Unmarried}, \text{Adult}, \text{Male})$

- In FOL:

  $\text{Bachelor}(x) \Leftrightarrow \text{Unmarried}(x) \wedge \text{Adult}(x) \wedge \text{Male}(x)$

```
Concept → Thing | ConceptName
   | And(Concept, …)
   | All(RoleName, Concept)
   | AtLeast(Integer, RoleName)
   | AtMost(Integer, RoleName)
   | Fills(RoleName, IndividualName, …)
   | SameAs(Path, Path)
   | OneOf(IndividualName, …)
Path → [RoleName, …]
```

Syntax of a subset of CLASSIC

Expressiveness vs. Tractability

```
And(Man, AtLeast(3, Son), AtMost(2, Daughter),
    All(Son, And(Unemployed, Married, All(Spouse, Doctor))),
    All(Daughter, And(Professor, Fills(Department, Physics, Math))))
```

- Men with at least three sons, all unemployed and married to doctors, and at most two daughters, all physics or maths professors
- Goal: subsumption in polynomial time
- Price: little or no negation or disjunction, which force case analysis
- Today: OWL for the Semantic Web; SNOMED CT for medical terms

Default Reasoning: Nonmonotonicity

- You see a parked car and assume four wheels, though you see only three
  - You withdraw that if you see it jacked up with a wheel beside it
- Classical logic is monotonic:

  $$\text{if } \mathrm{KB} \models \alpha \text{ then } \mathrm{KB} \wedge \beta \models \alpha$$

- Defaults are nonmonotonic: new information can remove conclusions
- We have already met two: inheritance with overriding, and the closed-world assumption
- Two formalisms, both from 1980: circumscription and default logic

Circumscription

$\text{Bird}(x) \wedge \neg \text{Abnormal}_1(x) \Rightarrow \text{Flies}(x)$

- Circumscribe Abnormal_1: assume it is false wherever possible
- $\text{Bird}(\text{Tweety})$ gives $\text{Flies}(\text{Tweety})$… until we learn $\text{Abnormal}_1(\text{Tweety})$
  - Penguins, ostriches, birds with broken wings
- Model preference: a conclusion holds if it is true in all *preferred* models
  - Preferred = fewest abnormal objects
- The closed-world assumption is the special case that prefers the fewest true atoms

The Nixon Diamond

$\text{Republican}(\text{Nixon}) \wedge \text{Quaker}(\text{Nixon})$

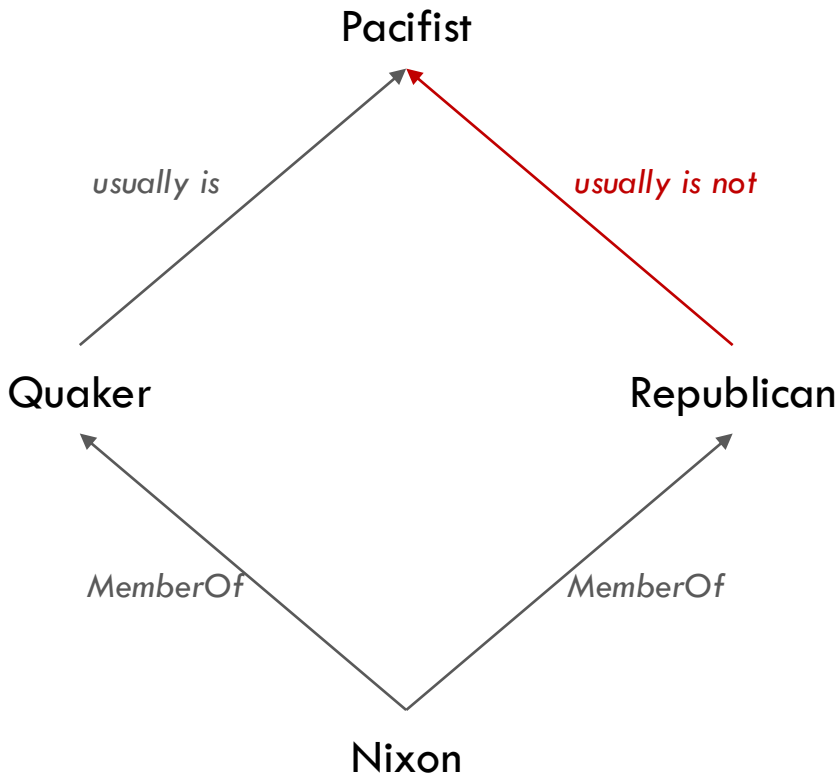
$\text{Republican}(x) \wedge \neg\text{Abnormal}_2(x) \Rightarrow \neg\text{Pacifist}(x)$

$\text{Quaker}(x) \wedge \neg\text{Abnormal}_3(x) \Rightarrow \text{Pacifist}(x)$

Two preferred models: one where Nixon is a pacifist, one where he isn't

Circumscription stays agnostic

Prioritized circumscription: let religion win over politics



Default Logic

$\mathrm{Bird}(x) : \mathrm{Flies}(x) \;/\; \mathrm{Flies}(x)$

- If $\mathrm{Bird}(x)$ holds and $\mathrm{Flies}(x)$ is consistent with the KB, conclude $\mathrm{Flies}(x)$
- In general: $P : J_1, \ldots, J_n \;/\; C$ (prerequisite : justifications / conclusion)
- An extension is a maximal consistent set of conclusions. The Nixon diamond has two
- Open question: how should defaults drive decisions?
  - "My brakes are always OK": fine for a car in the city, not for a loaded truck going down a ghat road
  - One answer: read defaults as statements about high probability

Exercise 2: Birds, Penguins and Conflicts

- Birds usually fly. Penguins are birds, and penguins usually don't fly

  $\text{Tweety} \in \text{Birds}$ $\text{Opus} \in \text{Penguins}$ $\text{Penguins} \subset \text{Birds}$

  a. Using inheritance in a semantic network, does Tweety fly? Does Opus?

  b. Write both rules in default logic. How many extensions does Opus get?

  c. Now add: Opus ∈ CartoonStars, and cartoon stars usually fly. What should a reasoner conclude about Opus?

Work in pairs

Exercise 2: Solutions

a. Tweety flies by default. Opus doesn't: the Penguins value is more specific and overrides Birds

b. Two defaults:

$$\text{Bird}(x) : \text{Flies}(x) \,/\, \text{Flies}(x) \qquad \text{Penguin}(x) : \neg\text{Flies}(x) \,/\, \neg\text{Flies}(x)$$

- Opus gets two extensions. Plain default logic doesn't prefer the more specific rule
- Fix: add ¬Penguin(x) to the bird rule's justification, or use priorities

c. Penguins and CartoonStars are *unrelated*, so neither is more specific

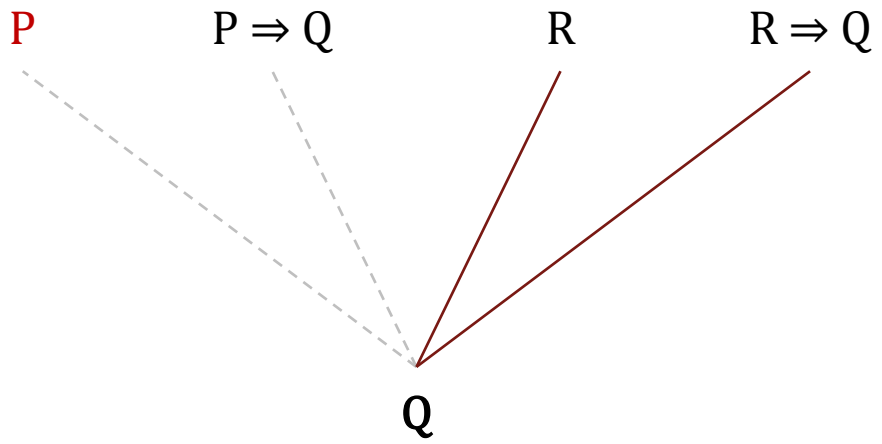
- This is a Nixon diamond: stay agnostic unless we add a priority

Truth Maintenance

- Belief revision: the KB says P, and now we learn ¬P
  - First we must $\mathrm{RETRACT}(P)$
- But Q may have been derived from P using $P \Rightarrow Q$
- Removing everything derived from P is wrong if Q also follows from R and $R \Rightarrow Q$
- Naive fix: roll back to just before P was added, then replay
  - Correct, but far too slow for a large, busy knowledge base

Justification-Based Truth Maintenance

- Each sentence stores its justifications, the sentences it was inferred from
- Retract P: remove sentences with P in *every* justification
- Mark sentences out instead of deleting them, so they are cheap to bring back
- Fast what-if analysis: change one choice, reuse the rest



P is retracted, so its justification for Q is gone (dashed). Q stays in: it still has {R, R ⇒ Q}

Exercise 3: What Goes Out?

- A JTMS holds premises A, B, E, and these justified sentences:

$$C \leftarrow \{A, B, A \wedge B \Rightarrow C\}$$
$$D \leftarrow \{C, C \Rightarrow D\} \quad \text{and} \quad \{E, E \Rightarrow D\}$$
$$F \leftarrow \{D, D \Rightarrow F\}$$

a. RETRACT(A). Which sentences go out?

b. Then RETRACT(E). Which go out now?

c. Then assert A again. What comes back, and how much work does that take?

Work in pairs

Exercise 3: Solutions

a. Only C goes out. D keeps its justification through E, so F stays too

b. Now D has no valid justification, so it goes out, and F goes with it

c. C, D and F all come back in

- The justifications were kept, so nothing is re-derived: the TMS just relabels

- Compare the naive approach: undo everything since A was added and replay it all

Assumptions and Explanations

- ATMS: keep every hypothetical world at once
  - Label each sentence with the sets of assumptions that make it true
- An explanation of P is a set of sentences E that entails P
  - It may include assumptions, things not known but enough to prove P

  $\text{“The car won't start”} \Leftarrow \{\text{BatteryDead}\} \text{ or } \{\text{NoFuel}\}$
- Prefer minimal explanations
- Truth maintenance is NP-hard in general, but very useful in practice

Case Study: An Internet Shopping Agent

- Input: a product description, such as *"laptops"* or *"digital camera under ₹30,000"*
- Percepts: web pages as HTML text, not pixels
- Output: relevant offers, ranked
- A page is an offer if a link or form on it says "buy", "price" or "add to cart"
- Strategy: start at a store's home page and follow only relevant category links

$$\text{Amazon} \in \text{OnlineStores} \wedge \text{Homepage}(\text{Amazon}, \text{“amazon.com”})$$

Following Relevant Links

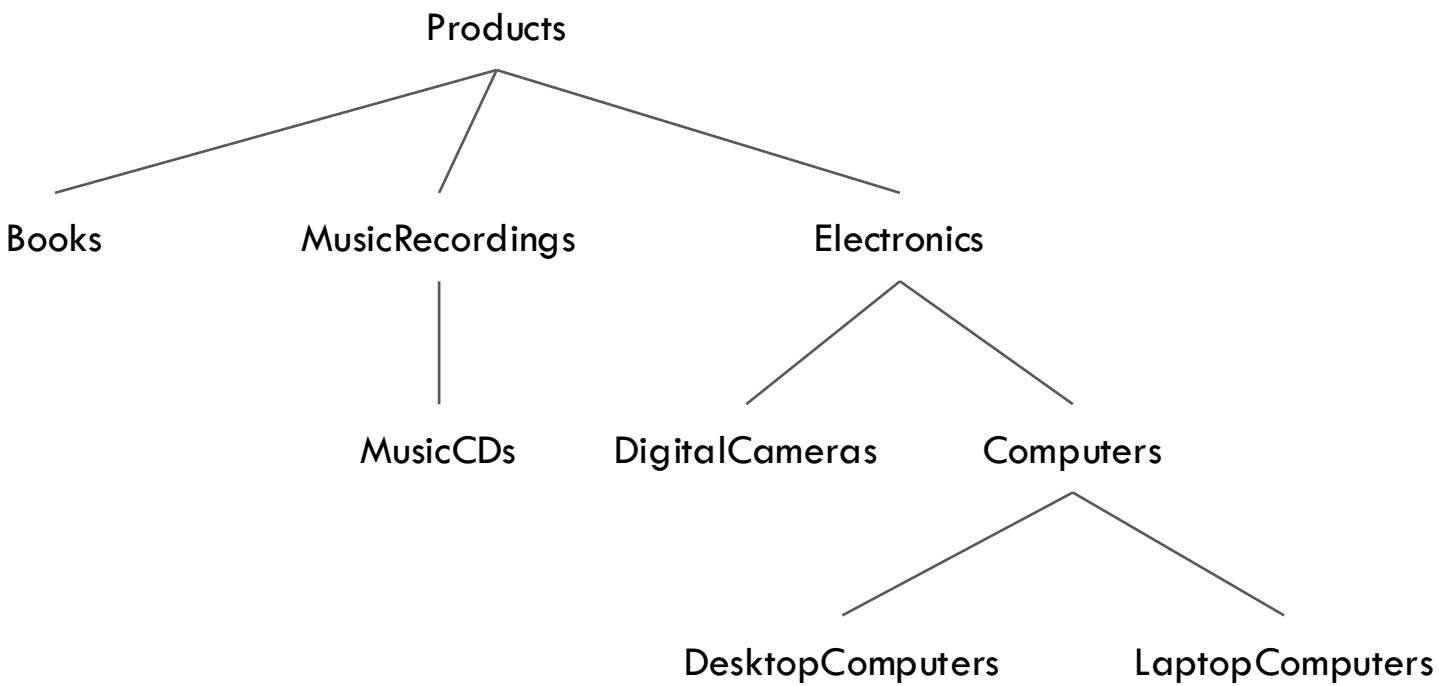
- A chain of links is relevant if every anchor text names a relevant category:

$$\begin{aligned}&\text{RelevantChain}(\text{start}, \text{end}, \text{query}) \Leftrightarrow (\text{start} = \text{end})\\ &\quad \vee (\exists u, \text{text}\ \ \text{LinkText}(\text{start}, u, \text{text}) \wedge \text{RelevantCategoryName}(\text{query}, \text{text})\\ &\qquad \wedge \text{RelevantChain}(u, \text{end}, \text{query}))\end{aligned}$$

- Recursive, with the empty chain as base case: the `path` program from last lecture
- Anchor text is relevant if it names the same, a broader or a narrower category:

$$\begin{aligned}&\text{RelevantCategoryName}(\text{query}, \text{text}) \Leftrightarrow\\ &\quad \exists c_1, c_2\ \ \text{Name}(\text{query}, c_1) \wedge \text{Name}(\text{text}, c_2) \wedge (c_1 \subseteq c_2 \vee c_2 \subseteq c_1)\end{aligned}$$

Categories and Their Names



Name("books", Books)

Name("CDs", MusicCDs)

Name("laptops", LaptopComputers)

Name("notebooks", LaptopComputers)

Name("CDs", CertificatesOfDeposit)

- Synonymy: "laptops" and "notebooks" name one category
- Ambiguity: "CDs" names two. Both multiply the paths the agent must follow

Getting and Comparing Offers

- Contents(url) by procedural attachment: an HTTP request made during inference
- Wrappers extract facts: Price(offer, $(399)), MemorySize(c, Gigabytes(2))
- The price belongs to the offer, not the product

Offer	CPU	RAM	Disk	Price	
A	1.4 GHz	2 GB	250 GB	$299	undominated
B	1.2 GHz	4 GB	350 GB	$500	undominated
C	1.2 GHz	2 GB	250 GB	$399	dominated by A

- X dominates Y if X is better on at least one attribute and worse on none. Return the undominated offers