

For Monday after Spring Break

- Check website for reading assignment
- Homework:
 - Chapter 10, exercise 22
 - Note that this is a VERY challenging exercise

Program 2

- Any questions?

Question 4

- How do we talk about belief?

Reification

- Turning propositions into objects.
- Why would we want (need?) to do this?

Consider the following:

- Jack thinks that the President is still George Bush.
- When I was in Washington, D.C. last month, I got to meet the President.

- This is the issue of **referential transparency** vs. **referential opaqueness**.

- Special rules for handling belief:
 - If I believe something, I believe that I believe it.
 - Need to still provide a way to indicate that two names refer to the same thing.

Knowledge and Belief

- How are they related?
- Knowing **whether** something is true
- Knowing **what**

And Besides Logic?

- Semantic networks
- Frames

Semantic Networks

- Use graphs to represent concepts and the relations between them.
- Simplest networks are ISA hierarchies
- Must be careful to make a **type/token distinction**:

Garfield isa Cat Cat(Garfield)

Cat isa Feline $\forall x (\text{Cat}(x) \Rightarrow \text{Feline}(x))$

- Restricted shorthand for a logical representation.

Semantic Nets/Frames

- Labeled links can represent arbitrary relations between objects and/or concepts.
- Nodes with links can also be viewed as frames with slots that point to other objects and/or concepts.

First Order Representation

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")

Inheritance

- Inheritance is a specific type of inference that allows properties of objects to be inferred from properties of categories to which the object belongs.
 - Is Bill alive?
 - Yes, since Bill is a cat, cats are mammals, mammals are animals, and animals are alive.
- Such inference can be performed by a simple graph traversal algorithm and implemented very efficiently.
- However, it is basically a form of logical inference
$$\forall x (\text{Cat}(x) \Rightarrow \text{Mammal}(x))$$
$$\forall x (\text{Mammal}(x) \Rightarrow \text{Animal}(x))$$
$$\forall x (\text{Animal}(x) \Rightarrow \text{Alive}(x))$$
$$\text{Cat}(\text{Bill})$$
$$\vdash \text{Alive}(\text{Bill})$$

Backward or Forward

- Can work either way
- Either can be inefficient
- Usually depends on branching factors

Semantic of Links

- Must be careful to distinguish different types of links.
- Links between tokens and tokens are different than links between types and types and links between tokens and types.

Link Types

Link Type	Semantics	Example
$A \xrightarrow{\text{subset}} B$	$A \subset B$	Cats \subset Mammals
$A \xrightarrow{\text{member}} B$	$A \in B$	Bill \in Cats
$A \xrightarrow{R} B$	$R(A,B)$	Bill $\xrightarrow{\text{Age}}$ 12
$A \xrightarrow{\boxed{R}} B$	$\forall x, x \in A \Rightarrow$ $R(x,B)$	Birds $\xrightarrow{\boxed{\text{Legs}}}$ 2
$A \xrightarrow{\boxed{\boxed{R}}} B$	$\forall x \exists y, x \in A \Rightarrow y$ $\in B \wedge R(x,y)$	Birds $\xrightarrow{\boxed{\boxed{\text{Parent}}}}$ Birds

Inheritance with Exceptions

- Information specified for a type gives the default value for a relation, but this may be over-ridden by a more specific type.
 - Tweety is a bird. Does Tweety fly?
Birds fly. Yes.
 - Opus is a penguin. Does Opus fly?
Penguin's don't fly. No.

Multiple Inheritance

- If hierarchy is not a tree but a directed acyclic graph (DAG) then different inheritance paths may result in different defaults being inherited.
- Nixon Diamond

Nonmonotonicity

- In normal monotonic logic, adding more sentences to a KB only entails more conclusions.

if $KB \vdash P$ then $KB \cup \{S\} \vdash P$

- Inheritance with exceptions is not monotonic (it is nonmonotonic)
 - Bird(Opus)
 - Fly(Opus)? yes
 - Penguin(Opus)
 - Fly(Opus)? no

- Nonmonotonic logics attempt to formalize default reasoning by allow default rules of the form:
 - If P and concluding Q is consistent, then conclude Q.
 - If Bird(X) then if consistent Fly(x)

Defaults with Negation as Failure

- Prolog negation as failure can be used to implement default inference.

```
fly(X) :- bird(X), not(ab(X)).
```

```
ab(X) :- penguin(X).
```

```
ab(X) :- ostrich(X).
```

```
bird(opus).
```

```
? fly(opus).
```

Yes

```
penguin(opus).
```

```
? fly(opus).
```

No