NON-MONOTONIC LOGIC

Presentation by
Ariel Ginis

The Problem(s) With Classical Logic

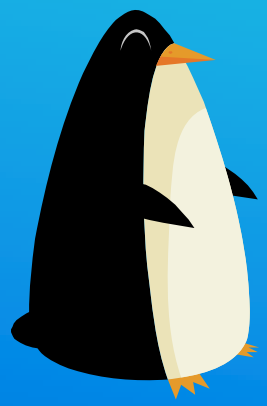
- Size of Theory necessary to describe real situations is overwhelmingly large
- Very weak in the face of incomplete knowledge
- Too rigid to deal with new (conflicting) knowledge

Meet Tweety

BIRD(Tweety)

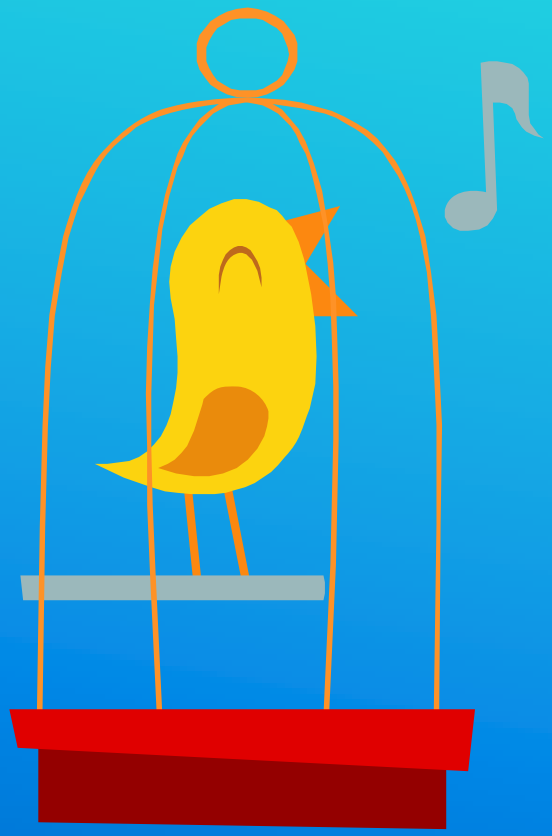
Problem:Size

$\forall x.\mathbf{Bird}(x) \rightarrow \mathbf{Fly}(x)$




Problem:Size

$\forall$x.Bird(x)&-Penguin(x)&-Ostrich(x)&-Peacock(x)&-Chicken(x)$\rightarrow$Fly(x)



Problem:Size

∀x.B(x)&-P1(x)&-O(x)&-P2(x)&-C(x)&-Tied(x) &-Caged(x)&-Cemented(x)→Fly(x)

Problem:Size

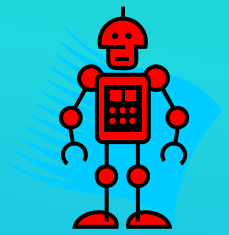
∀x.B(x)&-P1(x)&-O(x)&-P2(x)&-C1(x)&-T1(x) &-C2(x)&-C3(x)&-Injured(x)&-Dead(x)&-Baby(x)→Fly(x)

Problem:Size

- Well you get the picture!
- Always possible to add more detail to the environment
- This relates directly to the Frame Problem

The Frame Problem

Consider a world that has a finite collection of objects with assorted characteristics and properties. An agent can perform some set of actions in this world to move from static state to static state.



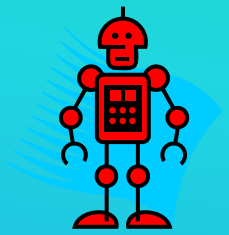
Example: Colored Blocks

In this example a robot can Move or Color a set of blocks.

We would like to capture the notion that "Everything that can stay the same does stay the same."

When the robot moves block A does block B move? Usually not but what if they are connected? Or if A moves to the position B occupied before?

Does moving A change its color? Usually not but what if A is moved into a bucket of paint?



Example: Colored Blocks

In a complex world the qualifications become long which is the issue of size.

On the other hand trying to express the rule that anything that can stays the same is difficult with classical logic due to the abstract nature of “what can stay the same.” This is the issue of weakness.

Problem:Weak

Lets agree to focus on a more practical example. We are given the axiom (in our practical world) that $\forall$**x.Bird(x)&-Penguin(x)**$\rightarrow$**Fly(x)**

We already know that **Bird(Tweety)** so what can we conclude about **Fly(Tweety)?**

Problem:Weak

Unfotunately, NOTHING! Until we know whether or not Tweety is a penguin we are stuck and can draw no conclusions about Tweety's flying abilities.

General rules are weak in classical logic because each exception must be described individually.

Problem:Rigid

For practical purposes let us assume that Tweety is NOT a Penguin. But today we are informed that Tweety actually is a Penguin. Now our theory contains a contradiction and any predicate is trivially true. We are unable to incorporate new information into our theory effectively. In real situations we often find that some of our assumptions are mistaken and need to be retracted.

Before Non-Monotonic Logic

- Hypothetical (Counterfactual) Reasoning

  What would be true in a world where we change the truth value of some predicates.

  (e.g. What would the world be like if Germany won WWII?)

- Paraconsistent Logic

  Logic where it is not necessarily true that $A \wedge -A \Rightarrow B$

$Married(Jack) \wedge -Married(Jack) \Rightarrow Student(Jack)$

Non-Monotonic Logic (NML)

Motivation

To define a strong yet flexible system to deal with situations of incomplete knowledge.

- Declarative: Syntax and Semantic rules
- Should have a practical implementation

Non-Monotonic Logic (NML)

Definition

Monotonic Property – For every theory T and T'
$T \models A \Rightarrow T \cup T' \models A$

Non-Monotonic – a system which violates the monotonic property.

Declarativism

If we made NML inference procedure based (the truth of a predicate is determined by the output of a logic program) we could run into major problems when the output does not match our intuitive expectations.

Example: PROLOG & Nixon

PROLOG is a logic programming language

Facts have the form $P(x_1 \ldots x_n)$.

Rules have the form

$P(x_1 \ldots x_n)$:- $Q_1(x_i \ldots x_k), \ldots, Q_m(x_i \ldots x_k)$

To check if $P(x_1 \ldots x_n)$ is true the program checks if $Q_1(\ldots)$ through $Q_2(\ldots)$ are true

Example: PROLOG & Nixon

Consider the following encoding

Pacifist(x):- quaker(x), not(hawk(x))

Hawk(x):- republican(x), not(pacifist(x))

Quaker(Nixon)

Republican(Nixon)

What happens when we ask the program

Pacifist(Nixon)?

Endless loop!!

Symbolic vs. Numeric

Symbolic – mapping predicates to either true or false	Numeric – using probability to evaluate predicates (values 0-1)
Can be applied in situations where probability can't be used. (e.g. exact statistical data is not available). Hard to distinguish between axioms and default conclusions.	Probability theory is already well developed. Default conclusions don't have the same truth value as axioms. ε-semantics approximate symbolic approach

Challenges of NML

Formalization

- How will we define our syntax and semantics
- Which inference rules can be used to draw conclusions

Revision

- How do we update our conclusions with the introduction of new evidence (axioms)
- Resolve contradictions
- Reject some earlier inferences based on new axioms
- Domino effect

Absence of Information

Every day reasoning is characterized not only by supporting evidence but lack of contrary evidence as well.

The positive evidence provides the cause and effect part of the conclusion while the lack of negative evidence ensures that the conclusion is rational.

Work and Mondays

1. On weekdays I usually go to work.
2. Today is a weekday.

Conclusion: I am going to work today.

This is a cause and effect, however if I add the fact that

3. Today is Christmas.

The conclusion is no longer rational although the cause and effect relationship has not changed.

General Form of a NM Rule

Given A, in the absence of evidence B, infer conclusion C

Or more generally:

Given A, in the absence of evidence to the contrary infer C

Types of NML

Autoepistemic Reasoning

- Incomplete representation of complete knowledge
- Depends on context
- P is valid because otherwise I would have information that leads to –P.

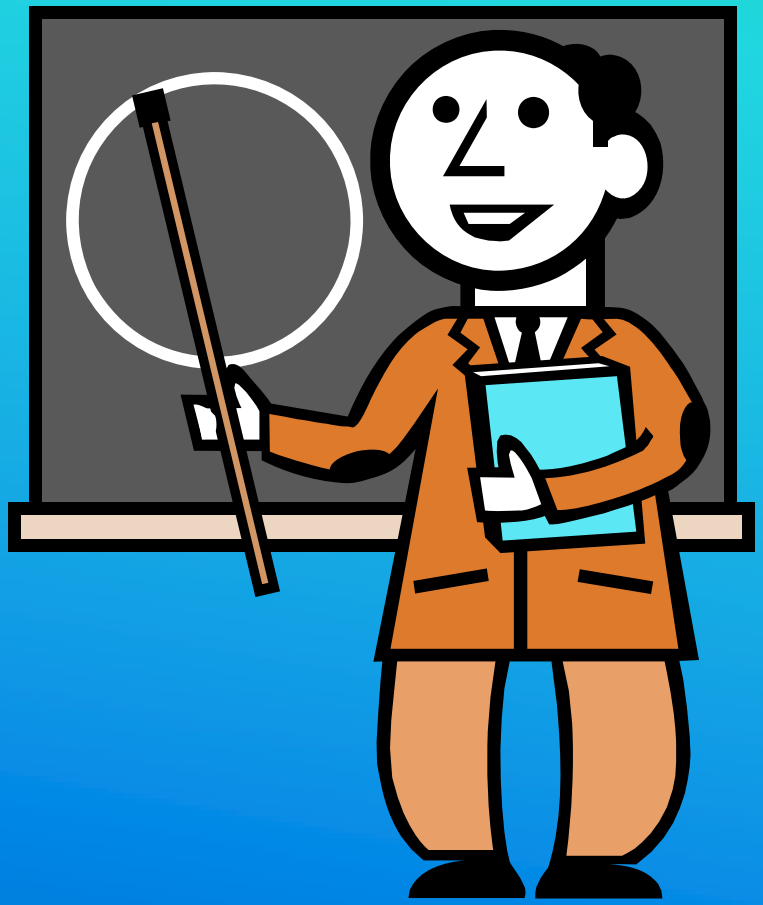
Explicit Autoepistemic Reasoning

Use an explicit convention to draw conclusions:

"The next lecture will be on Thursday"

The convention used is:

"Lectures are every Thursday unless we are told otherwise."



Subjective Autoepistemic Reasoning

No convention is given but you believe that P must be true or you would have evidence to the contrary.

"The university is still on strike"

Belief:

"If the strike were over someone would have told me."

Default Reasoning

- Rational conclusions from partial information
- Defeasible: new evidence can invalidate the conclusion.

<u>4 types</u>

1. Prototypical
2. No-risk
3. Best-guess
4. Probabilistic

Prototypical Reasoning

- Typically P is true so it is rational to assume it is true now.

Bird(Tweety)

Typically birds fly therefore

Fly(Tweety)

No-Risk Reasoning

When the cost of making a mistake that – P outweigh the cost of making a mistake that P then assume P.

"The semester will probably be cancelled but just in case I will do the homework anyway because if I'm wrong I will fail the course"

The cost of assuming the semester will be cancelled and being wrong is much higher than the cost of assuming it won't be cancelled and being wrong.

Best-Guess Reasoning

When no evidence supports P or –P choose whichever is more convenient.

"I need to enter the university but I don't know which gate is open–go to the gate that is closest."

Probabilistic Reasoning

If P has a high probability assume P.

"The probability that I will wait less than 10 minutes for a bus to the university is 0.8 therefore I can assume that I will wait less than 10 minutes."

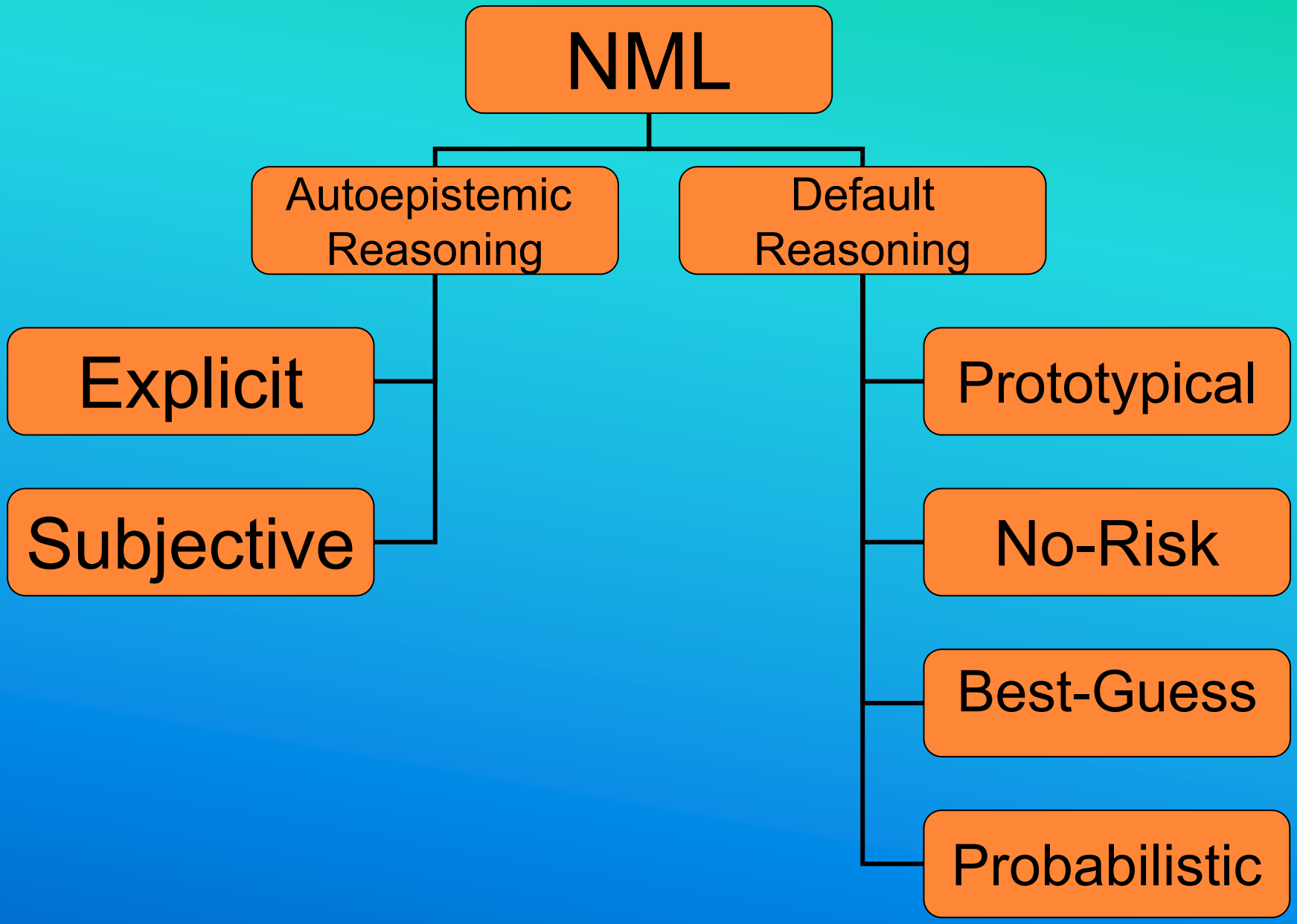
Rules and Types

A rule can be used or explained by different types of NML.

Rule: Boats can usually be used to cross a river.

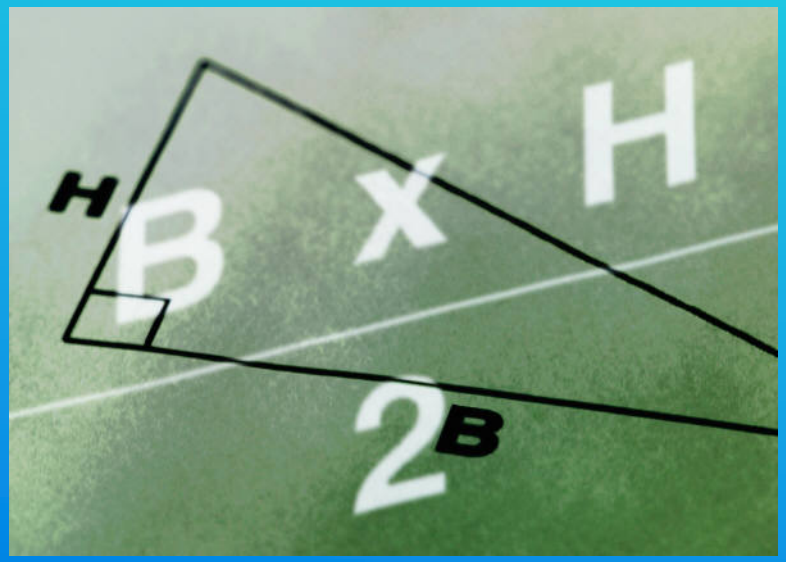
Can a particular boat cross the river?

1. Yes. I would know if there was something wrong with the boat (subjective autoepistemic)
2. Yes. Typically boats can cross the river (prototypical)
3. Yes. I need to cross the river and have no other way to do it. (Best-Guess)


NML
Autoepistemic Reasoning
Default Reasoning
Explicit
Subjective
Prototypical
No-Risk
Best-Guess
Probabilistic

BREAK!!

Formal Approaches to NML



Early Attempts at Formalization

- Modal operators ‘normally’ and ‘consistent’ – McCarthy & Hayes 1969
- Unless operator – Sandewall 1972
  T |~ Unless(A) iff **not** T|- A

  A ^ Unless(B) → C

Difficulties of ‘Unless’ Operator

Given:

1. A
2. A ^ Unless(B) → C

Conclude

1. C
2. Unless(C)

Given:

1. Q(Nixon)^R(Nixon)
2. ∀x.Q(x)^
   Unless(–P(x))→ P(x)
3. ∀x.R(x)^
   Unless(P(x))→ –P(x)

Conclude

1. P(x) ^ –P(x)

Default Logic (Reiter 1980)

A theory is a pair <W,D> where W is a set of regular axioms and D is a set of defaults

A default is a rule that has the following form

$$\frac{\text{Bird}(x):\text{Flies}(x)}{\text{Flies}(x)}$$

And is understood to mean "if x is a bird and it is consistent to believe x flies then assume x flies"

Circumscription

Uses the special predicate Ab(x) which stands for Abnormal(x).

The interpretation of Ab(x) is intuitive (not formally defined) – an object is abnormal if it does not satisfy the rule.

$$\forall x.Bird(x) \wedge -Ab(x) \rightarrow Flies(x)$$

Minimization of abnormal predicates (no object is abnormal unless it must be).

Closed World Assumption (CWA)

Essentially assume that everything you don't know (can't prove) is false.

Example: Travel Agent Database

Database of flights with origin and destination. If a flight is not in the database assume it does not exist.

Preferential Models (Shoham 1988)

Define a pair $<L,\sqsubset>$ where L is a language NML that supports models then $\sqsubset$ is a partial order on the frames.

$M_1 \sqsubset M_2$ reads as M_2 is preferred over M_1.

If M is a model of T (M|–T) M is a preferred model if no other model M' is a model of T and $M \sqsubset M'$.

We say that $T|\!-_{\sqsubset} P$ iff P is true for all preferred models of T

Applications

Inheritance Heirarchies

Similar to circumscription - which objects are abnormal.

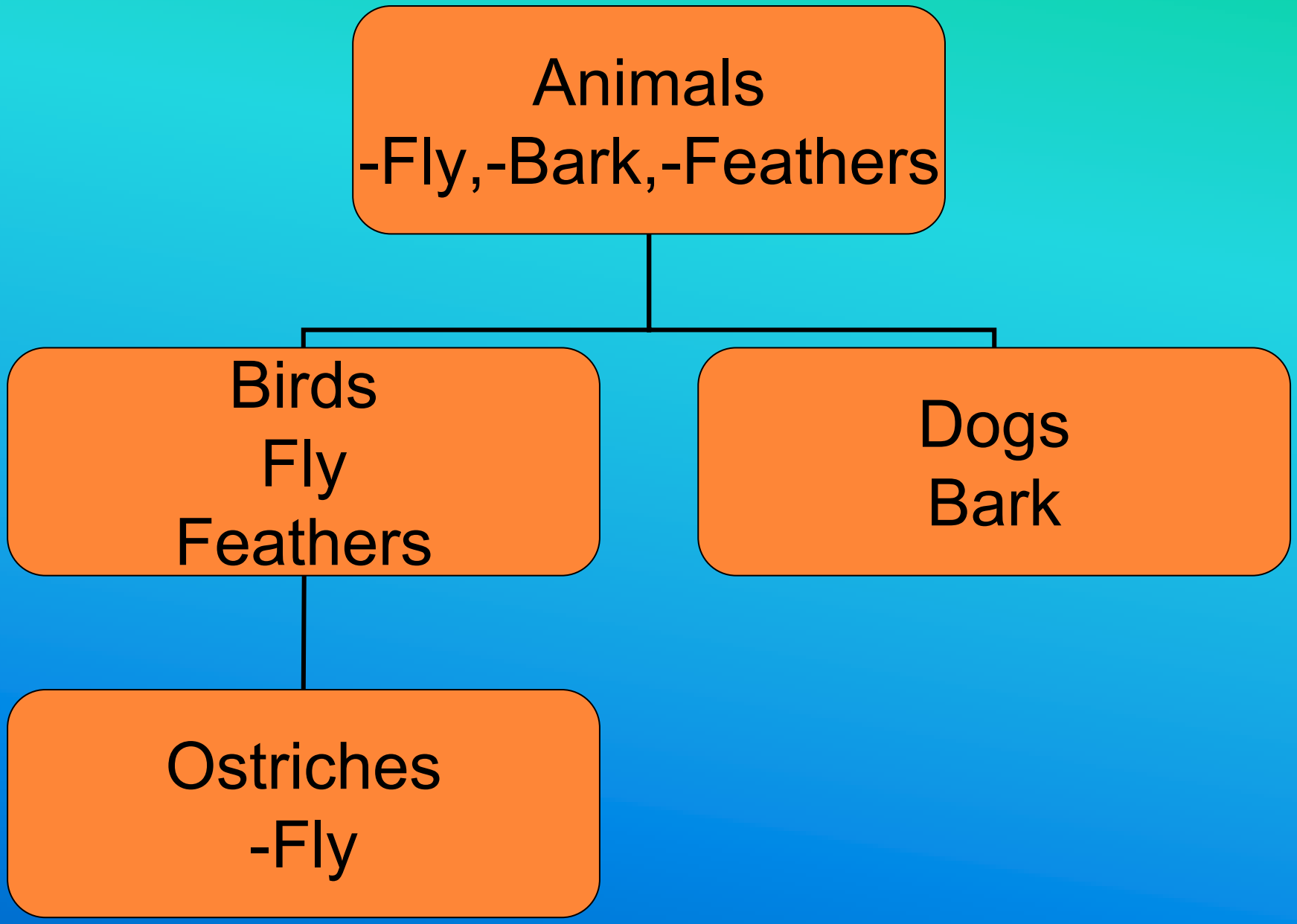
Animals don’t fly

Birds do fly

Ostriches don’t fly

Ostriches in a plane do fly

To decide if an object has a certain property go up the tree until you find an answer.


Animals
-Fly,-Bark,-Feathers
Birds
Fly
Feathers
Dogs
Bark
Ostriches
-Fly

Diagnostics

If a program works properly it should take an input and return a value.

We assume that each subroutine works properly. However, if the program fails then one of our assumptions was incorrect.

Approach the problem not as device/code failure but as design/algorithm failure. How do we change the design to match the output?

Reasoning about Actions

How to make rational decisions based on partial information.

When looking for your car assume it is where you left it. (Even though this may not be true, it could have been towed or stolen)

Database Updates

Any dynamic database, can upon receipt of new entries be in a state of conflict when there are two contradictory entries.

This is the revision problem how do we reconcile the database (and hopefully be able to use any non-affected parts of the database.)

Communication

When you describe a situation you usually describe only the parts of the situation that violate default rules.

For example if you decide to tell someone about a trip to the movies you won’t mention that you were dressed or that you had bought a ticket. The default rule being that you are always dressed in public and that you pay for the movie.

Questions