Expert Systems 2024

Boston University

CS 640, AI

Slides by Margrit Betke, Yiwen Gu

The ZooKeeper, Bagger Example is by P. Winston.

What is an Expert System?

Expert system

= Knowledge-based system

= Knowledge navigator

= Rule-based system

= Deduction/reaction system

= Recommendation system

= Software agent

Early Successful Expert Systems

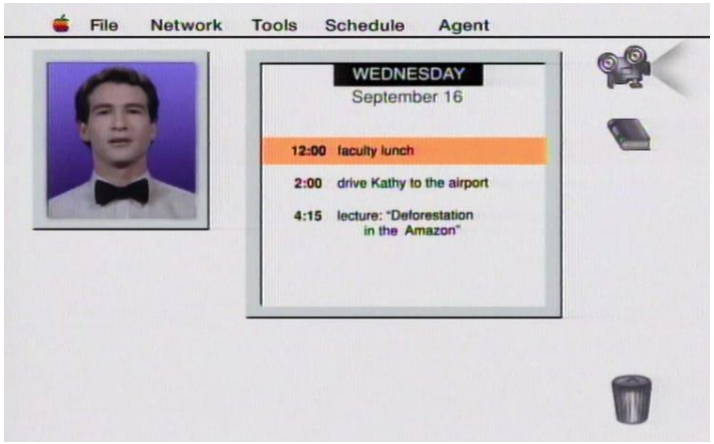
- Most Famous Industry Products:
  - Mavent: check mortgage loans against a variety of laws and regulation, review for compliance.
  - Mycin: diagnose bacterial infections and recommend antibiotic treatments.
  - XCON: configure orders for new computer systems.
- Many More Agricultural Expert Systems:
  - LIMEX: Assessment, irrigation, fertilization, and pest control recommendations to assist lime growers
  - Tomatex (tomatoes)
  - Citex (oranges)
  - Cupex (cucumbers)
  - Neper Wheat

Source: WIKI

Not So Successful Office Assistants

- Microsoft's "Clippit" or "Clippy"
  - Lack of Intelligence
  - Intrusiveness

- Apple's Knowledge Navigator
  - Concept video
  - Not an actual product
  - → Siri, a hybrid, incorporates rule-based components for specific tasks (e.g. "set a timer")



Research Trend since 2020

- Early, rule-based approaches to expert systems need to be understood to develop a new-generation expert system
- Pure data-based (statistical) machine learning may not be the answer
- Recent research trend: Combination of rule-based systems and knowledge bases (domain knowledge) with statistical machine learning systems

Expert Systems Around Us

- When you call your bank…

←**Interactive Voice Response (IVR) systems**. These systems prompt users with a series of pre-recorded messages and options, allowing them to navigate menus by pressing numbers or, in some cases, using voice commands.

- When you browse some website…

←**Website Chatbots.** If you visit a customer service page, the chatbot might ask questions like, "How can I help you today?" and provide options such as "Track my order" or "Return a product." Based on your selection, the chatbot follows a programmed path.

Types of Expert Systems, Rule Syntax

Rule-based Deduction Systems

Reaction Systems

Rule Syntax:

R_n : IF if1

if2

…

THEN then1

then2

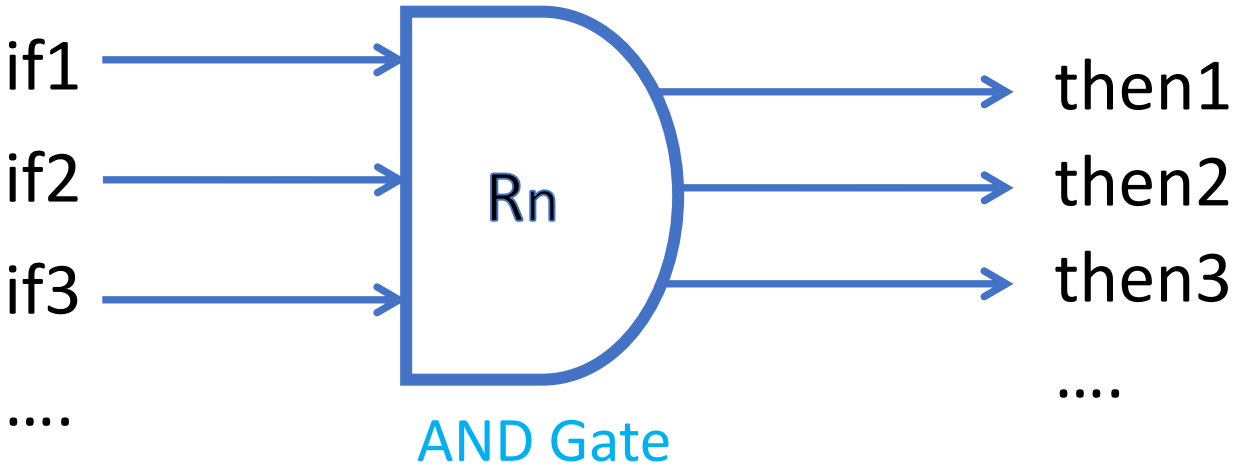
then3

…

antecedents

consequents

Circuit Visualization



IF *all antecedents of Rn are satisfied*
THEN *rule is "triggered"* or
action is "fired"

Terminology

- "Antecedent" = assertion
- "Consequent" = assertion in deduction system
  = reaction in reaction systems
- Fact = something true (never false)
- Assertion = statement that something is true (can be false)

Facts or Assertions?

- It's going to be sunny tomorrow.
  - Assertions
- It's going to be sunny tomorrow, according to the weather forecast.
  - Facts
- An apple everyday keep the doctor away.
  - Assertions
- Milk is good for your health.
  - Assertions
- Milk is a good source of calcium, which is important for bone health.
  - Facts

Terminology

- "Antecedent" = assertion
- "Consequent" = assertion in deduction system
  = reaction in reaction systems
- Fact = something true (never false)
- Assertion = statement that something is true (can be false)

- " Working Memory" = collection of assertions
- " Forward Chaining" = moving from IF to THEN:

Example: ZooKeeper

- Deduction system to identify animals in a zoo
- 7 animals:

  cheetah, tiger, giraffe, zebra, ostrich, penguin, albatross

<u>1st idea: Build a rule for each type of animal:</u>

R_n : IF *all characteristics of an animal*

THEN *This is animal of type n*

=> 7 rules with long list of antecedents

Example: ZooKeeper

- Deduction system to identify animals in a zoo
- 7 animals:

  cheetah, tiger, giraffe, zebra, ostrich, penguin, albatross

2nd idea: Create rules that produce intermediate assertions

- Reduce number of antecedents by checking common characteristics
- Produce intermediate assertions
- Compute chains of conclusions

Z1: IF *?x has hair*

THEN *?x is a mammal*

?x is a variable

More Terminology

- If the assertion “Stretch has hair” is in the working memory, variable x can be bound to Stretch
- We also say “Stretch is the binding of x”
- If all variables in the antecedent are bound, the rule is instantiated

ZooKeeper Rules

Z1: IF ?x has hair
THEN ?x is a mammal

Z2: IF ?x gives milk
THEN ?x is a mammal

Z3: IF ?x has feathers
THEN ?x is a bird

Z4: IF ?x flies
?x lays eggs
THEN ?x is a bird

Z5: IF ?x is a mammal
?x eats meat
THEN ?x is a carnivore

Z6: IF ?x is a mammal
?x has pointed teeth
?x has claws
?x has forward-pointing eyes
THEN ?x is a carnivore

Z7: IF ?x is a mammal
?x has hoofs
THEN ?x is an ungulate

Z8: IF ?x is a mammal
?x chews cud
THEN ?x is an ungulate

Z9: IF ?x is a carnivore
?x has tawny color
?x has dark spots
THEN ?x is a cheetah

ZooKeeper Rules Cont.

Z10: IF ?x is a carnivore
?x has tawny color
?x has black strips
THEN ?x is a tiger

Z11: IF ?x is an ungulate
?x has long legs
?x has long neck
?x has tawny color
?x has dark spots
THEN ?x is a giraffe

Z12: IF ?x is an ungulate
?x has white color
?x has black stripes
THEN ?x is a zebra

Z13: IF ?x is a bird
?x does not fly
?x has long legs
?x has long neck
?x is black and white
THEN ?x is an ostrich

Z14: IF ?x is a bird
?x does not fly
?x swims
?x is black and white
THEN ?x is a penguin

Z15: IF ?x is a bird
?x is a good flyer
THEN ?x is an albatross

Example: What is Stretch?

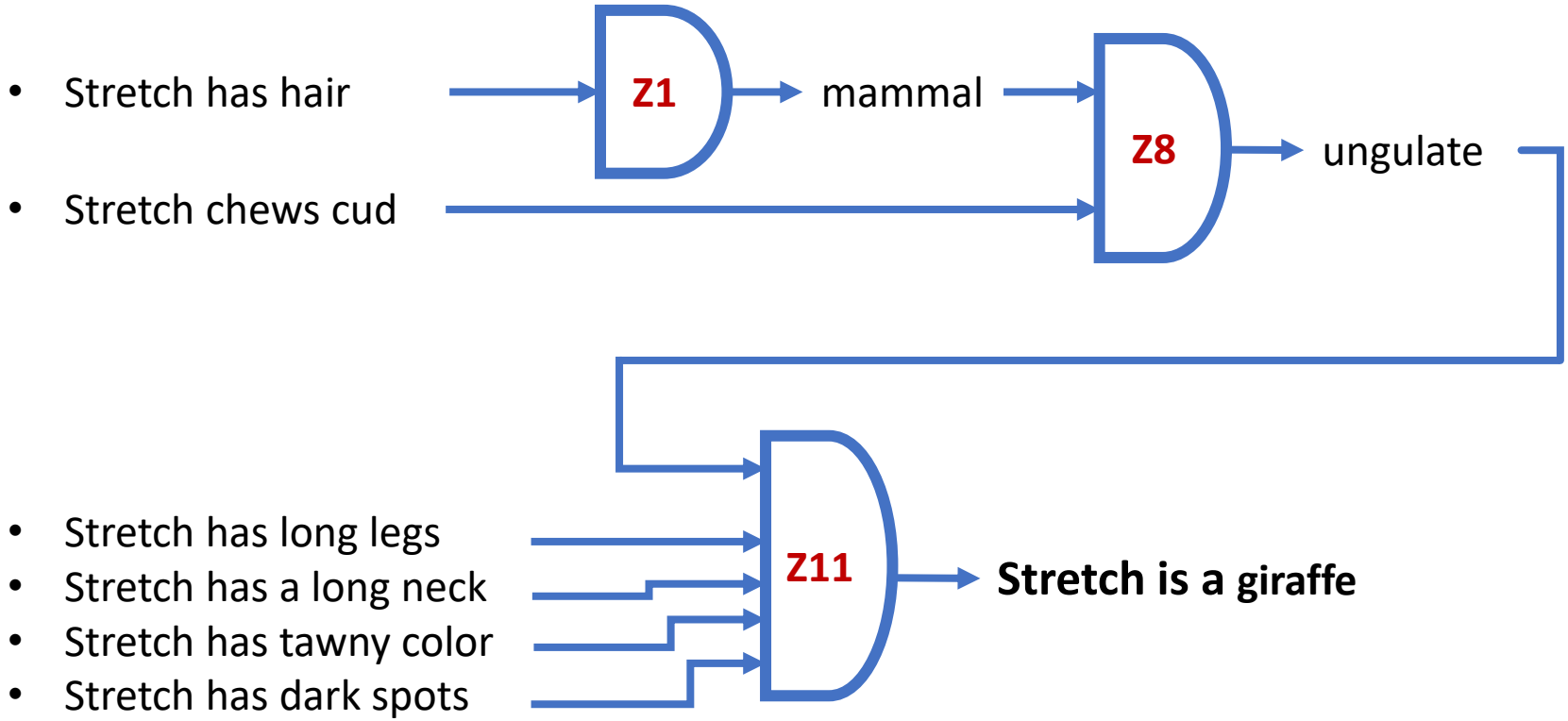
Working Memory:

- Stretch has hair
- Stretch chews cud
- Stretch has long legs
- Stretch has a long neck
- Stretch has tawny color
- Stretch has dark spots

Forward Chaining Procedure

- Until no rule produces a new assertion,
  - For each rule,
    - For each set of possible variable bindings determined by matching the antecedents to working memory,
      - Instantiate the consequent.
      - Determine whether the instantiated consequent is already asserted. If it is not, assert it.

What is Stretch?



Backward Chaining Procedure

- Until all hypotheses have been tried and none have been supported or until success,
  - For each **hypothesis**,
    - Find a **rule** whose consequent matches the hypothesis and create a binding set
    - Try to support each of the rule's **antecedents** by matching it to assertions in the working memory or by backward chaining through another rule, creating new hypotheses. Be sure to check all matching and instantiation alternatives.
    - If all the rule's antecedents are supported, announce success and conclude that the hypothesis is true.

Example: What is Swifty?

Working Memory:

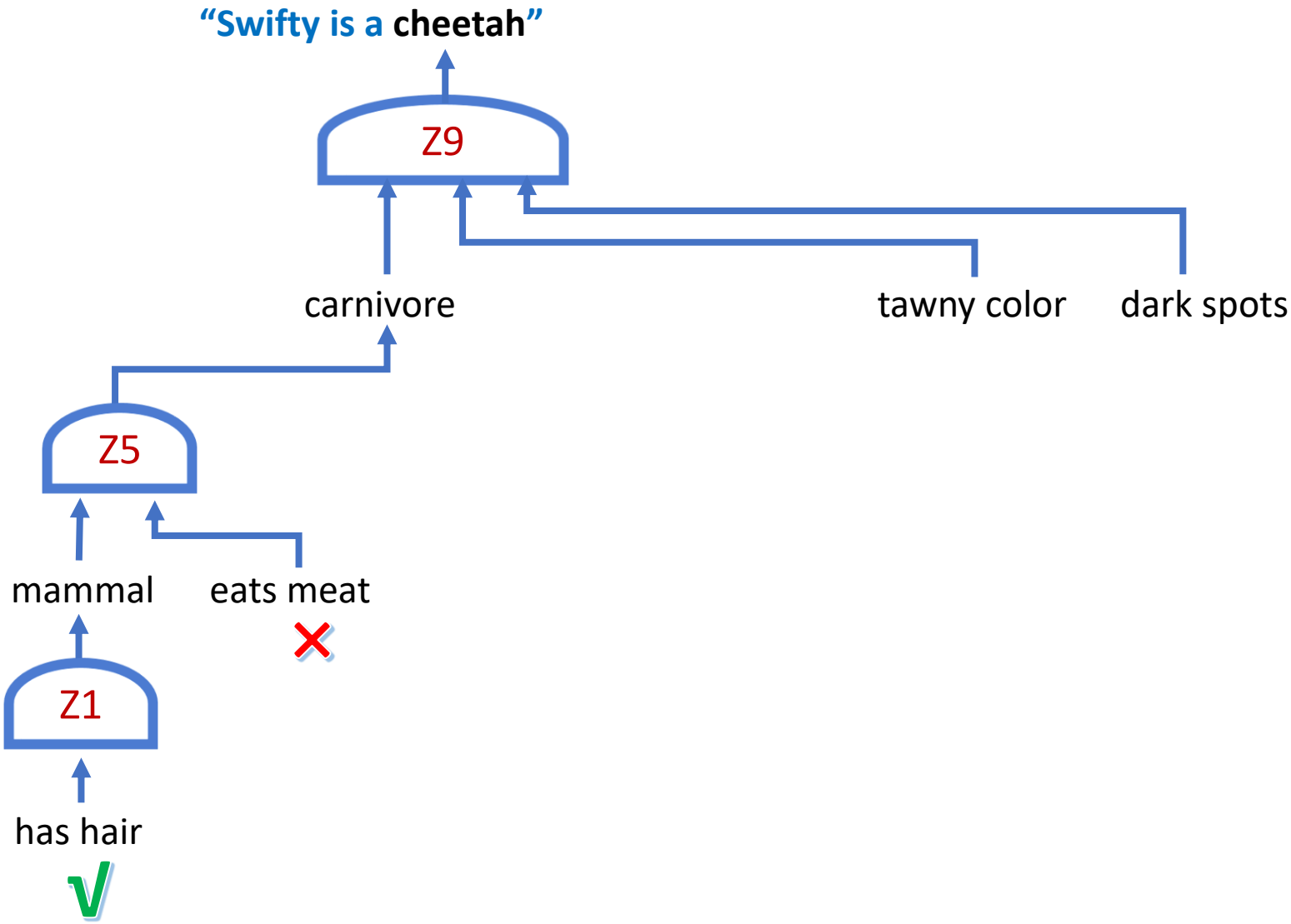
- Swifty has hair
- Swifty has pointed teeth
- Swifty has claws
- Swifty has forward-pointing eyes
- Swifty has tawny color
- Swifty has dark spots

Hypothesis:

"Swifty is a cheetah"

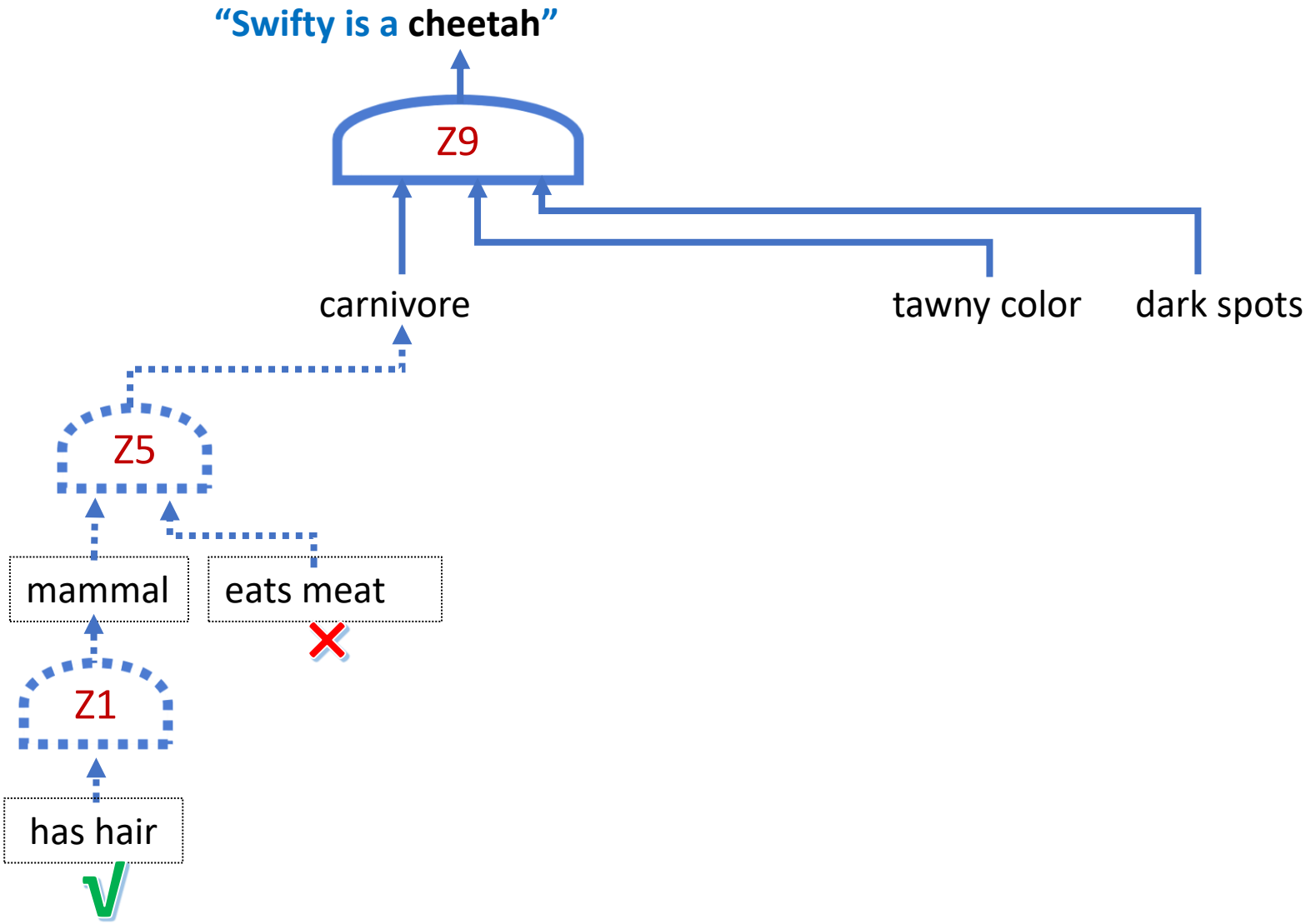
What is Swifty?

Backward Chaining circuit visualization



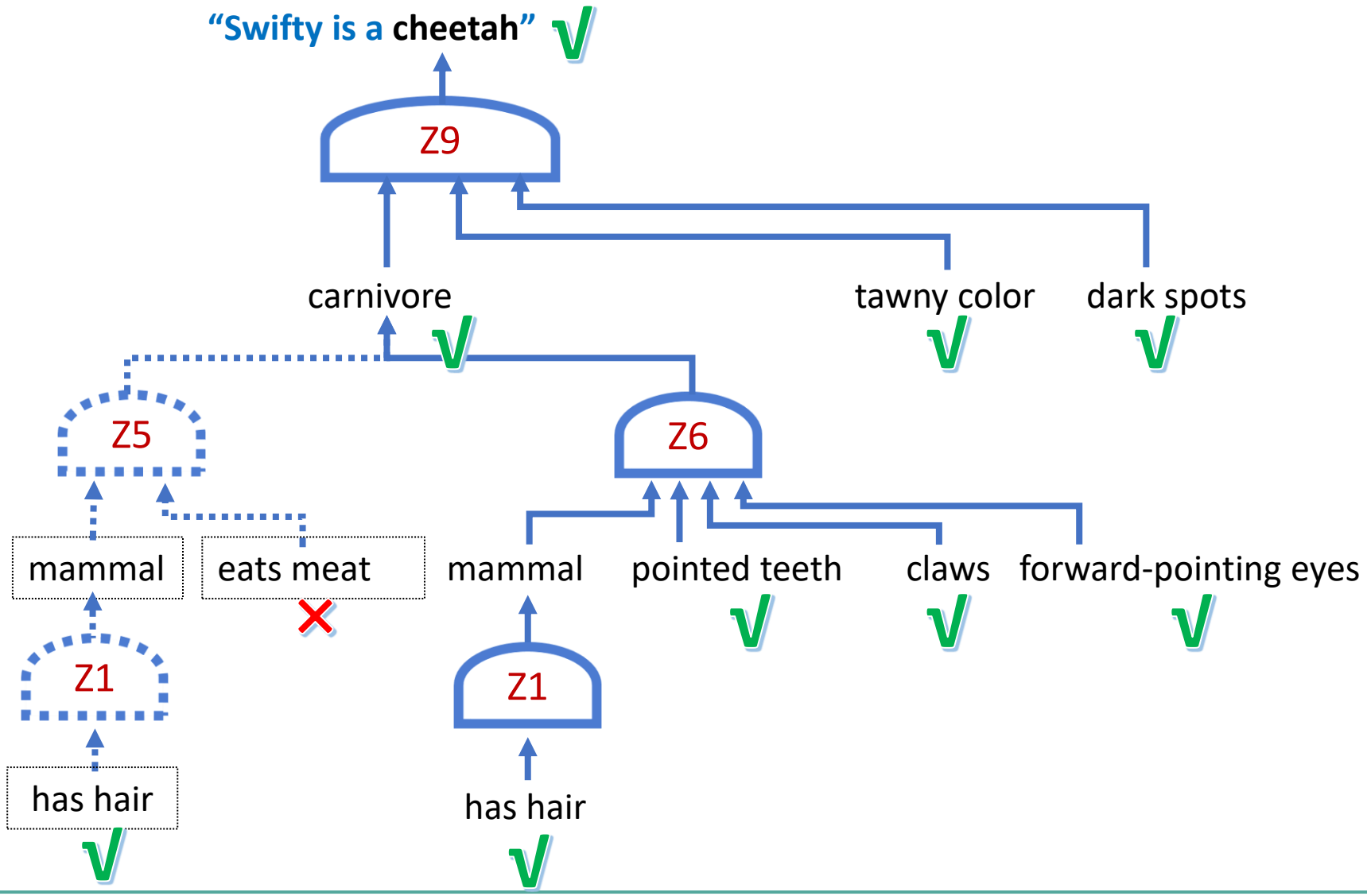
What is Swifty?

Backward Chaining circuit visualization



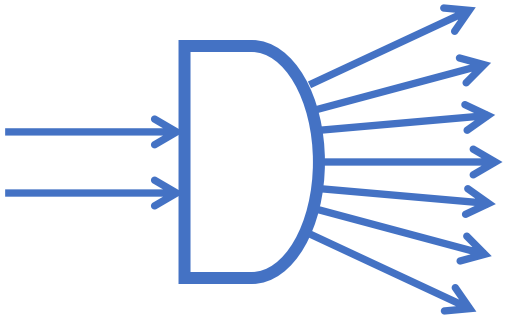
What is Swifty?

Backward Chaining circuit visualization



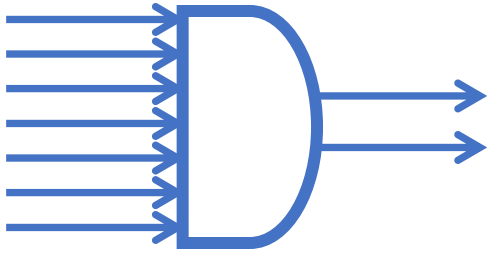
Forward vs. Backward Chaining

- Which direction is better?
  - Fan-in << Fan-out



backward

- Fan-in >> Fan-out



forward

Forward vs. Backward Chaining

- Which direction is better?

- Similar fan-in and fan-out:

  Backward chaining if

  - Interested in only one of many possible conclusions
  - Interested in only a general conclusion (“carnivore”)
  - Facts not gathered yet, only gather useful facts

  Forward chaining if

  - Animal disappeared, cannot gather new facts

Rule-Based Reaction System

- ZooKeeper is a deduction system
- An example of the reaction system: Bagger
  - Bag groceries in the store
  - Four steps:
    - The **check-order step**, make suggestions
    - The **bag-big-items step**, big bottles first
    - The **bag-medium-items step**, frozen items go to freezer bag
    - The **bag-small-items step**
  - See the handout for the complete set of rules

Rule-Based Reaction System: Bagger

IF *all antecedents of Rn are satisfied*
THEN *rule is "triggered"* or
action is "fired"

B2: IF step is check-order
THEN step is no-longer check-order
step is bag-big-items

→

B2: IF step is check-order
DEL step is check-order
ADD step is bag-big-items

Example: Bagger

Item	Container Type	Size	Frozen?
Bread	Plastic bag	Medium	No
Glop	Jar	Small	No
Granola	Cardboard box	Large	No
Ice cream	Cardboard carton	Medium	Yes
Potato chips	Plastic bag	Medium	No
Pepsi	Bottle	Large	No

Initial Working Memory:

- Step is check-order
- Bag1 is the current bag
- Bread is to be bagged
- Glop is to be bagged
- Granola is to be bagged
- Ice cream is to be bagged
- Potato chips are to be bagged

Example: Bagger

Item	Container Type	Size	Frozen?
Bread	Plastic bag	Medium	No
Glop	Jar	Small	No
Granola	Cardboard box	Large	No
Ice cream	Cardboard carton	Medium	Yes
Potato chips	Plastic bag	Medium	No
Pepsi	Bottle	Large	No

Working Memory:

Step is check-order
Bag1 is the current bag
Bread is to be bagged
Glop is to be bagged
Granola is to be bagged
Ice cream is to be bagged
Potato chips are to be bagged

B1
→ (yes) →

Step is check-order
Bag1 is the current bag
Bread is to be bagged
Glop is to be bagged
Granola is to be bagged
Ice cream is to be bagged
Potato chips are to be bagged
Pepsi is to be bagged

→

Example: Bagger

Item	Container Type	Size	Frozen?
Bread	Plastic bag	Medium	No
Glop	Jar	Small	No
Granola	Cardboard box	Large	No
Ice cream	Cardboard carton	Medium	Yes
Potato chips	Plastic bag	Medium	No
Pepsi	Bottle	Large	No

Working Memory:

B2, B3, B4 → → →

Step is bag-large-items
Bag1 contains Pepsi
Bag1 contains granola
Bag1 is the current bag
Bread is to be bagged
Glop is to be bagged
Ice cream is to be bagged
Potato chips are to be bagged

B6 →

Step is bag-medium-items
Bag1 contains Pepsi
Bag1 contains granola
Bag1 is the current bag
Bread is to be bagged
Glop is to be bagged
Ice cream is to be bagged
Potato chips are to be bagged

→

Example: Bagger

Item	Container Type	Size	Frozen?
Bread	Plastic bag	Medium	No
Glop	Jar	Small	No
Granola	Cardboard box	Large	No
Ice cream	Cardboard carton	Medium	Yes
Potato chips	Plastic bag	Medium	No
Pepsi	Bottle	Large	No

Working Memory:

B9, B8, (B7, B8), B8
→ → → →

Step is bag-medium-items
Bag1 contains Pepsi
Bag1 contains granola
Bag2 is the current bag
Bag2 contains bread
Bag2 contains ice cream (in freezer bag)
Bag2 contains potato chips
Glop is to be bagged

B10
→

Step is bag-small-items
Bag1 contains Pepsi
Bag1 contains granola
Bag2 is the current bag
Bag2 contains bread
Bag2 contains ice cream (in freezer bag)
Bag2 contains potato chips
Glop is to be bagged

→

Example: Bagger

Item	Container Type	Size	Frozen?
Bread	Plastic bag	Medium	No
Glop	Jar	Small	No
Granola	Cardboard box	Large	No
Ice cream	Cardboard carton	Medium	Yes
Potato chips	Plastic bag	Medium	No
Pepsi	Bottle	Large	No

Working Memory:

B12, B11, B13
→ → →

Step is done

Bag1 contains Pepsi

Bag1 contains granola

Bag2 contains bread

Bag2 contains ice cream (in freezer bag)

Bag2 contains potato chips

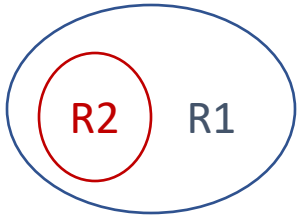
Bag3 is the current bag

Bag3 contains glop

Conflict Resolution Strategies

Rule ordering based on:

- **Priority list:** Rule ordering
- **Context**: Groups of rules, only one group active at any time
- **Specificity**:

  Conditions of R1 & R2: 

  Use R1 (the superset)

- **Data**: Prioritize assertions, use the triggered rule with highest priority assertion in conditions list
- **Size**: Use triggered rule with longest list of conditions
- **Recency**: Use least recently used rule

Learning Outcomes of this Lecture

- Can define what an expert system is (and know various alternative terms, e.g., knowledge base)
- Know terminology of rule-based systems
- Understand how the working memory is used and changed by instantiating rules
- Can describe forward & backward chaining, including drawing a circuit visualization
- Understand pro's and con's of forward & backward chaining
- Know when & how to use conflict resolution strategies