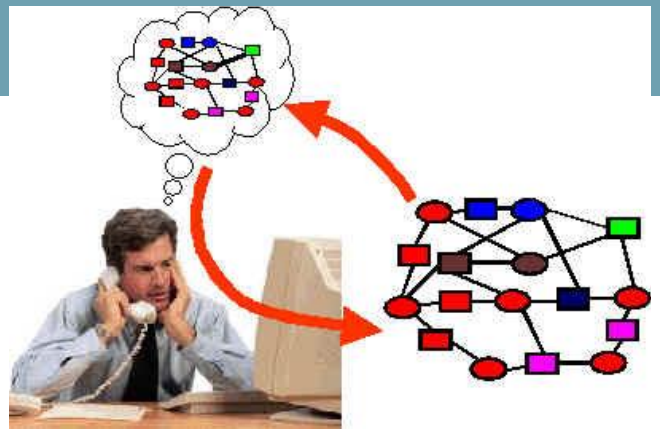
Artificial Intelligent Systems

(BMA-EL-IZB-LJ-RE 1. year 2024/2025)

Knowledge Representation

Simon Dobrišek

Lecture Topics



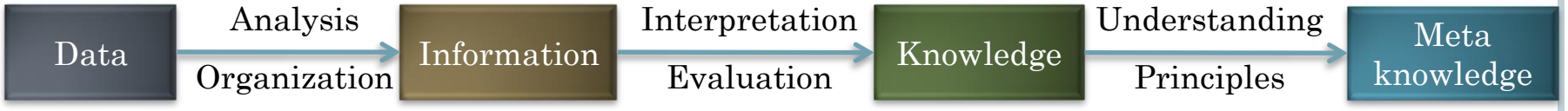
- How to present knowledge
- Knowledge model and knowledge map
- Knowledge representation schemes
- From informal to formal knowledge representation

How to present knowledge?

- Knowledge is a rather general concept that is often purely-defined.
- Knowledge provides the answers to the questions, such as "*How to do something*" and/or "*What is true or not true*".
- For instance, to know "*How to drive a car*" is a procedural knowledge, and to know that "*The sun is shining outside*" is a declarative knowledge.
- Knowledge representation is the way in which knowledge is stored in a system or presented to a person.
  - Knowledge is a description of the world.
  - Representation is the way knowledge is encoded.

From data to Wisdom

- Knowledge development begins with the data that is a collection of disconnected facts and have limited utility.
- Analysis and organization of the data (building relationships) create information.
- Interpretation and evaluation of information leads to knowledge.
- Understanding of basic principles that are embedded in the knowledge leads to the meta knowledge or "wisdom" (knowledge about knowledge).



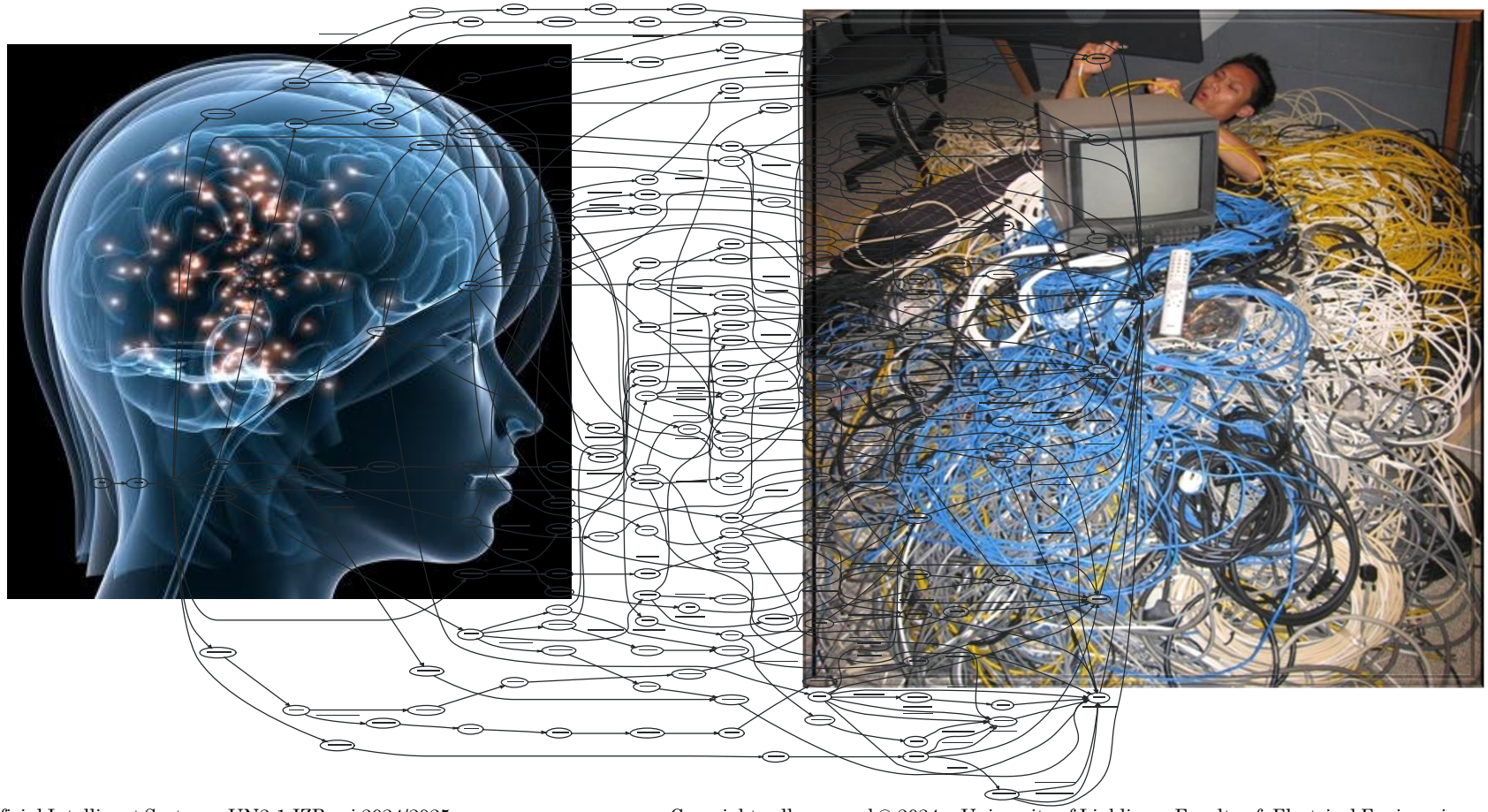
Knowledge Progression

- **Data** is viewed as collection of disconnected facts.
- **Information** emerges when relationships among facts are established and understood; Provides answers to *"who"*, *"what"*, *"where"*, and *"when"*.
- **Knowledge** emerges when relationships among patterns are identified and understood; Provides answers to *"how"* .
- **Wisdom** is the pinnacle of understanding, uncovers the principles of relationships that describe patterns; Provides answers as *"why"*.

- Example: *It is raining.*
- Example: *The temperature dropped 15 degrees and then it started raining.*
- Example: *If the humidity is very high and the temperature drops substantially, then atmospheres is unlikely to hold the moisture, so it rains.*
- Example: *There are interactions between raining, evaporation, air currents, temperature gradients and changes.*

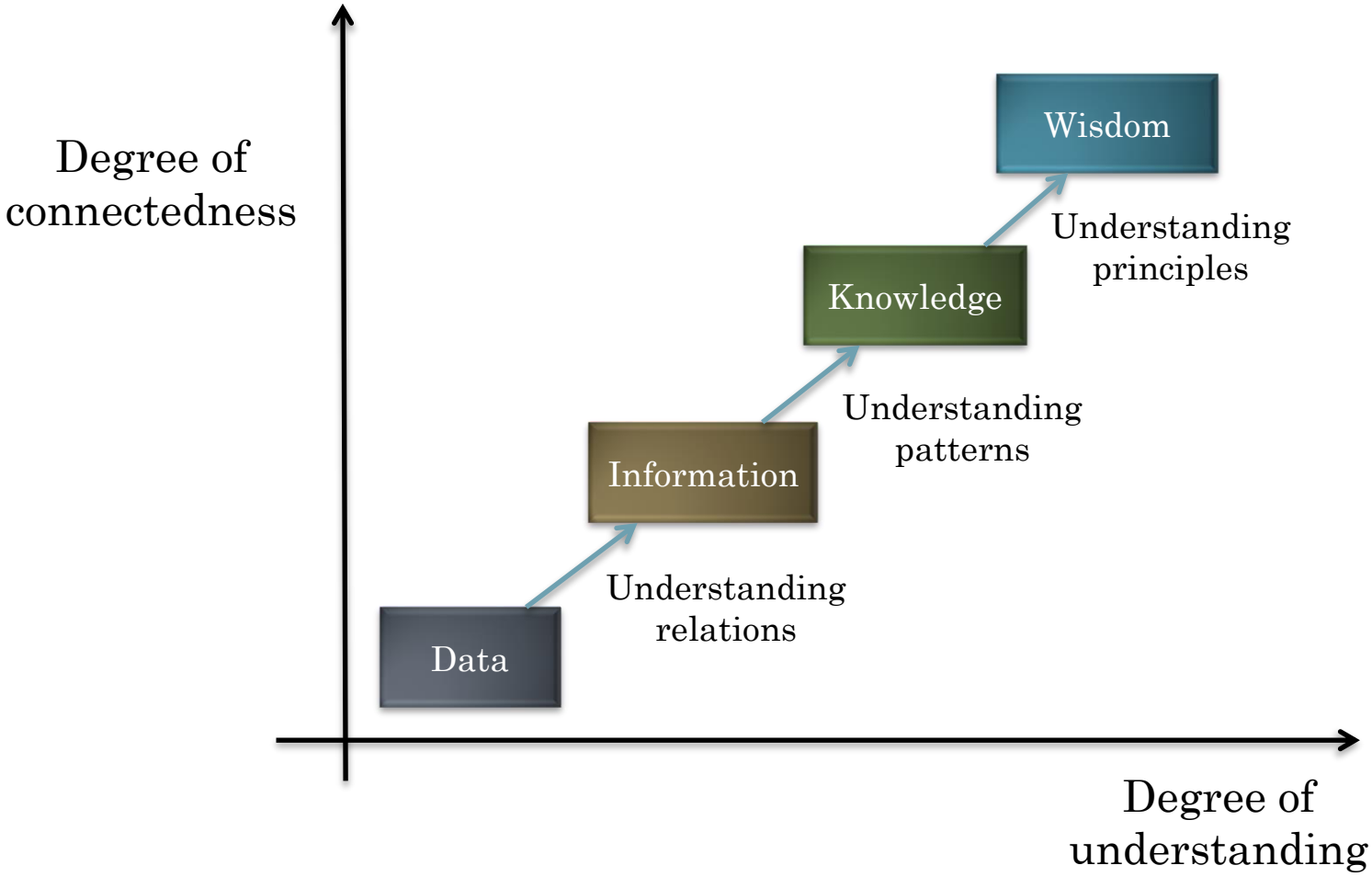
Knowledge Model

- A knowledge model tells the degree of *"connectedness"*, and *"understanding"* increases when progressing from *data* through *information* and *knowledge* to *wisdom*.



Knowledge Model

- The understanding support the transitions from one stage to the next stage.



Knowledge Model

- The distinctions between data, information, knowledge, and wisdom are not very discrete.
  - "Data" and "information" deal with the past; They are based on the gathering facts and adding context.
  - "Knowledge" deals with the present that enable us to perform.
  - "Wisdom" deals with the future, acquire vision for what will be, rather than for what is or was.

Implicit and Explicit knowledge

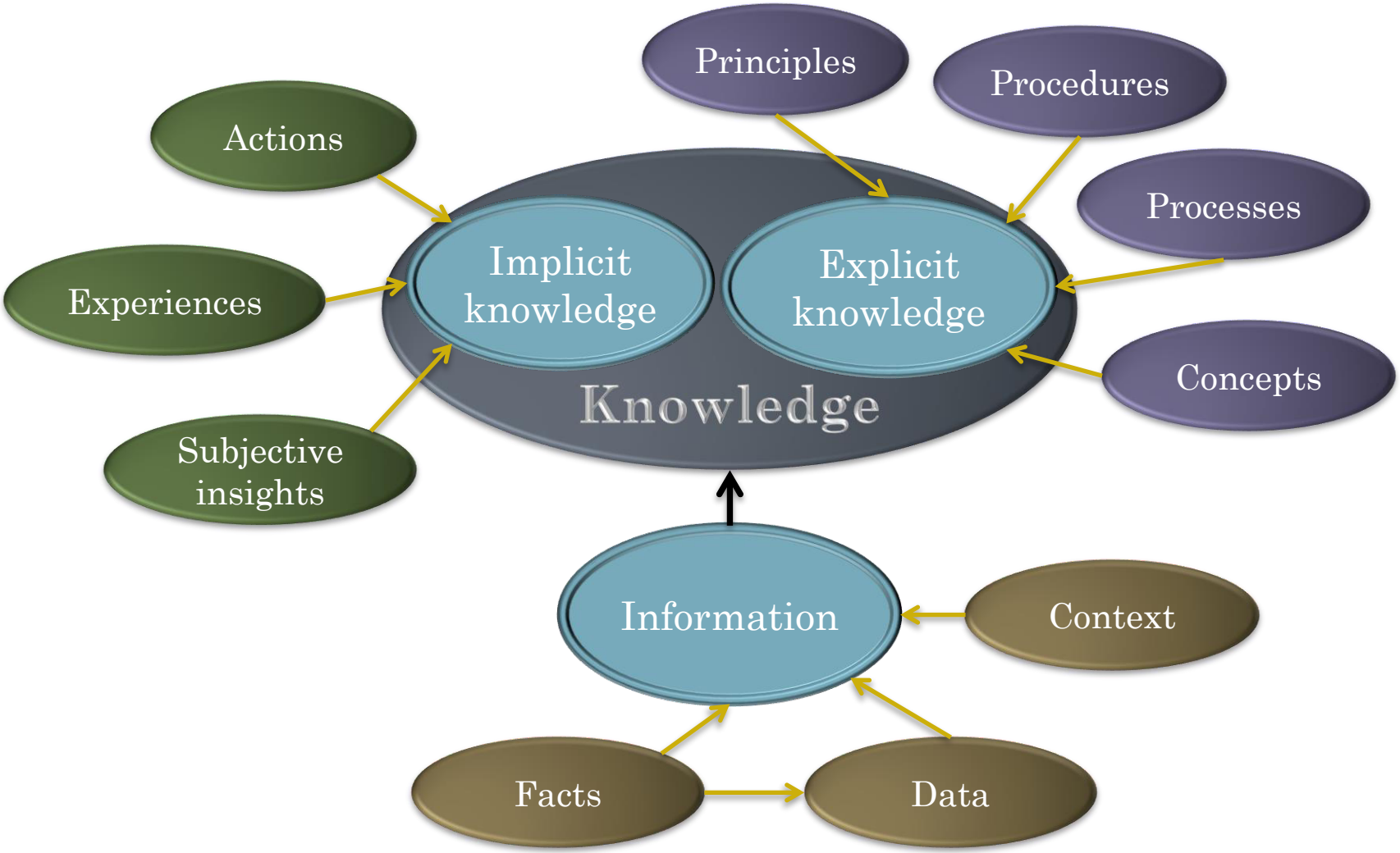
- Knowledge is usually categorized into two major types:
  - "Tacit" corresponds to "informal" or "implicit" type of knowledge
  - "Explicit" corresponds to "formal" type of knowledge.

Tacit knowledge	Explicit knowledge
Embodied in a human being.	Embedded in a human being.
Difficult to formally articulate.	Can be formally articulated.
Difficult to communicate and share.	Can be shared, copied, processed and stored.
Hard to steal or duplicate it.	Easy to steal or duplicate it.
Drawn from experiences, actions, and subjective insight.	Drawn from artifact of some type as principle, procedure, process, concepts.

KNOWLEDGE TYPOLOGY MAP

- Facts: are data or instance that are specific and unique.
- Concepts: are class of items, words, or ideas that are known by a common name and share common features.
- Processes: are flow of events or activities that describe how things work rather than how to do things.
- Procedures: are series of step-by-step actions and decisions that result in the achievement of a task.
- Principles: are guidelines, rules, and parameters that govern; principles allow to make predictions and draw implications;

KNOWLEDGE TYPOLOGY MAP



Procedural and Declarative Knowledge

- Procedural knowledge is a knowledge of how to do something.
- Declarative knowledge is a knowledge about what is true or false.
- Procedural knowledge can be tacit (e.g. cooking by feeling) or explicit (e.g. cooking recipes).
- Procedural knowledge is sometimes seen as declarative (declarative descriptions of methods as kinds of facts).
- Declarative knowledge is usually explicit, and it can be formally articulated and presented.

Procedural and Declarative Knowledge

Procedural knowledge

- Knowledge about "how to do something"; e.g., to determine if John or Thomas is older, first find their ages.
- Focuses on the tasks that have to be performed to reach a particular objective or goal.
- Knowledge representations schemes: procedures, rules, strategies, agendas, policies, models.

Declarative knowledge

- Knowledge about "that something is true or false". e.g., A car has four tires; John is older than Thomas;
- Refers to representations of objects and events; knowledge about facts and relationships;
- Knowledge representation schemes: concepts, objects, facts, propositions, assertions, semantic nets, logic and descriptive models.

Examples of procedural knowledge

- *In order to determine whether John is richer than Thomas, first determine their incomes, and then compare the amount of money earned.*
- *To make the car drivable, its fuel tank should not be empty.*
- *Bread is baked so that the dough made from flour is exposed to high temperatures for at least half an hour.*
- *...*
- *To bake bread, an oven is needed .*
  (Something between procedural and declarative knowledge)

Examples of Declarative knowledge

- *It's raining outside.*
- *Cars have four wheels.*
- *John is richer than Thomas.*
- *Isosceles triangles have two sides of equal length.*
- *Dogs are animals.*
- *Laura is driving to Seattle.*
- …

Computer Representation of Knowledge

- Computers require a well-defined problem to be processed in order to produce a well-defined solution to the problem.
- Representation of explicit knowledge is based on a natural language.
- To be used by computers the knowledge should be converted to a formal computer (programming) language.
- Computer problem solving thus requires the conversion of non-formal knowledge representation to the formal knowledge representation, i.e. the conversion of tacit knowledge to explicit knowledge.
- In computer science, many different formal languages for knowledge representation has been developed.

Computer Representation of Knowledge

- Converting informally articulated knowledge to a formal representation is still largely left to the intuition and experiences of the developers of such systems.
- For formal knowledge presentation, we need to understand how facts are mapped into symbols and symbols back to the facts.
- Natural languages (such as English, Slovenian, ...) are too complex to be used for the symbolic representation of knowledge that could be directly used by today's computers.
- Therefore, various schemes of simplified formal knowledge representations have been developed.

Inference From Knowledge

Facts

- *Barney is a dog.*
- `dog('Barney')`
- $\forall$ `dog(X)` $\rightarrow$ `has(X,'tail')`

Representation

- The fact is represented in a natural language (English).
- A symbolic representation of the fact in the formal language of mathematical logic.
- Logical representation of the fact, „*All dogs has a tail.*“

- The mechanism of deductive logical inference derive a symbolically representation of a new fact.

- `has('Barney','tail')`
- *Barney has a tail.*

- A symbolically representation of a new fact.
- A representation of the inferred fact in the natural language.

Knowledge Representation Schemes

- Different types of knowledge may require different formal knowledge representation scheme.
- The most common knowledge representation schemes are:
  - Decision tables and decision trees,
  - Production rules, rules with exceptions,
  - The language of mathematical logic,
  - Semantic networks and semantic frames,
  - (Fuzzy) Petri nets,
  - Regression trees, …
- Each knowledge presentation scheme may require a different method of inference from the input facts.

Decision Tables

- The simplest scheme for knowledge representation are decision tables.
- Their drawback is difficult selection/matching of the attributes and poor flexibility

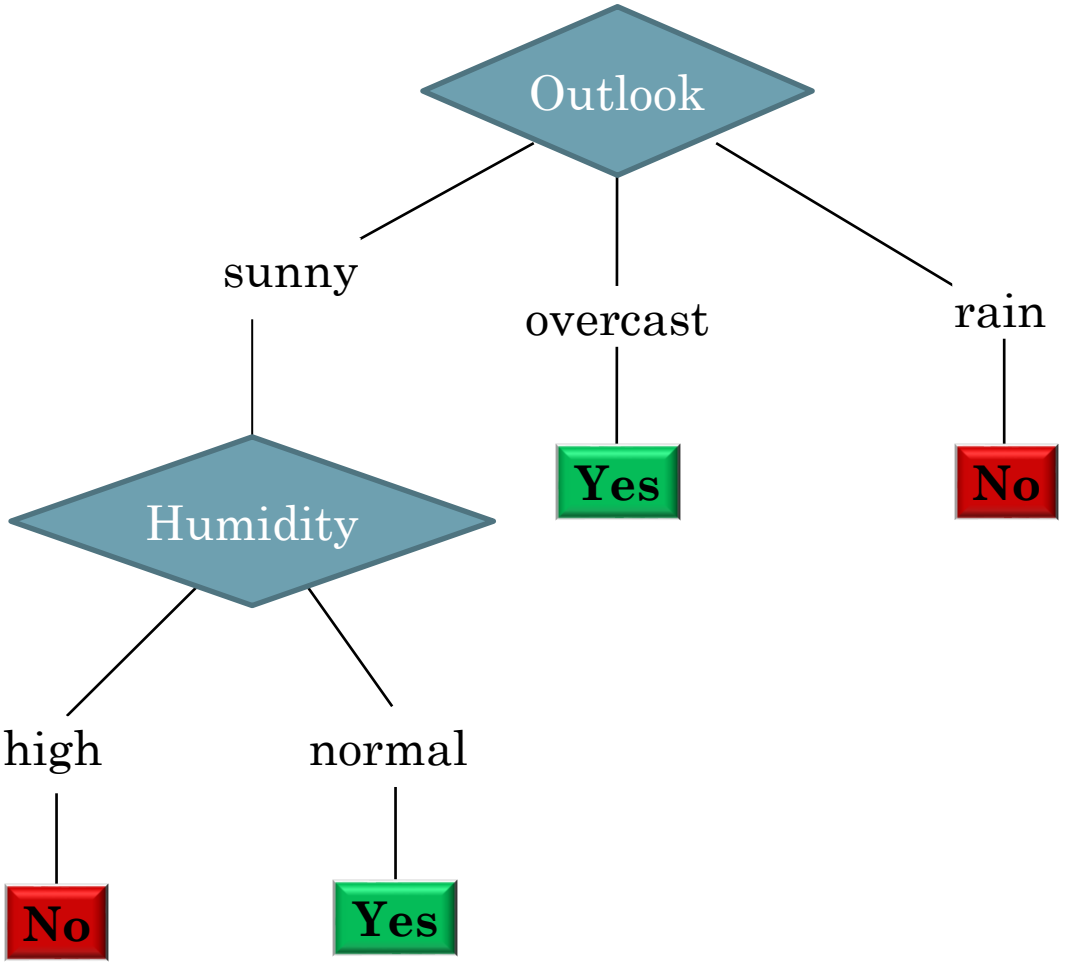
Outlook	Humidity	Play
Sunny	High	No
Sunny	Normal	Yes
Overcast	High	Yes
Overcast	Normal	Yes
Rainy	High	No
Rainy	Normal	No

The decision table for the weather problem

Decision Trees

- "Divide-and-conquer" approach produces a tree.
- Nodes involve testing a particular attribute.
- Attribute value is usually compared to a constant
- Other possibilities:
  - Comparing values of two attributes;
  - Using a function of one or more attributes.
- Leaves assign classification, set of classifications, or probability distribution to instances.
- Unknown input instance is routed down the tree.

DECISION TREE FOR THE WEATHER PROBLEM



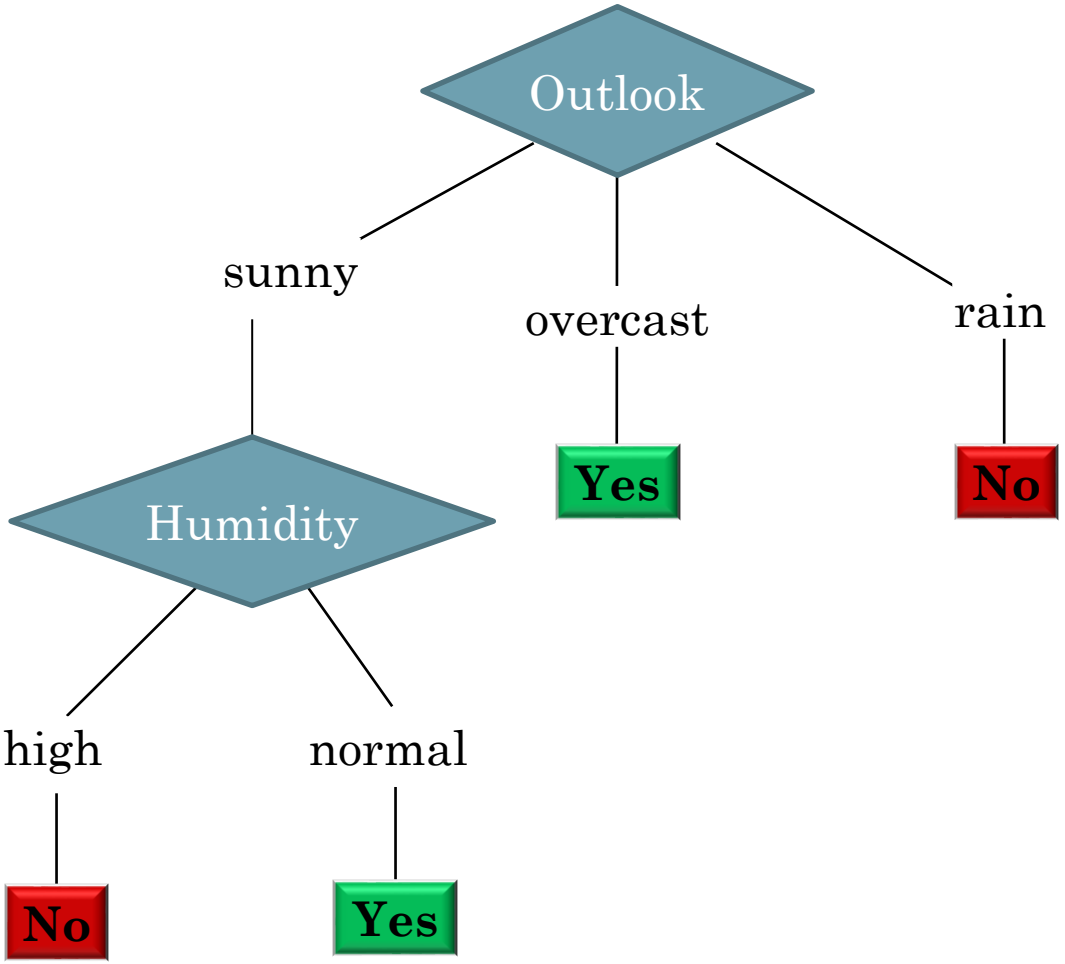
From Decision Trees to Production Rules

- Production rules are popular alternative to decision trees.
- *Antecedents* (pre-conditions) can be seen as the tests at the nodes of a decision tree.
- Tests are usually logically AND-ed together (but may also be general logical expressions).
- *Consequents* (conclusions): decisions, classes, set of classes, or probability distribution assigned by the rule.
- Individual rules are often logically OR-ed together.
  - Conflicts arise if different conclusions apply .

From Decision Trees to Production Rules

- Converting a tree into a set of rules is relatively easy.
  - One rule for each leaf:
    - Antecedent contains a condition for every node on the path from the root to the leaf.
    - Consequent is a decision/class assigned by the leaf.
- Produces rules that are unambiguous
  - Doesn't matter in which order they are executed.
- However, the resulting rules are sometimes unnecessarily complex
  - Pruning to remove redundant tests/rules

Writing Rules from the Tree



Writing Production Rules from the Decision Tree

```
IF outlook=sunny AND humidity=high
THEN play=no

IF outlook=sunny AND humidity=normal
THEN play=yes

IF outlook=overcast THEN play=yes

IF outlook=rain THEN play=no
```

Advantages and disadvantages of Production Rules

- Production rules are simple and easy to understand and are simpler than full predicate logic.
- Simple forward and backward chaining inference/search algorithms.
- Easy for programmers to implement and construct the rules and humans tend to understand the rules.

- Simple implementations are very inefficient.
- Some types of knowledge are not easily expressed in production rules.
- Large sets of rules become difficult to understand and maintain.

Decision Tree Learning

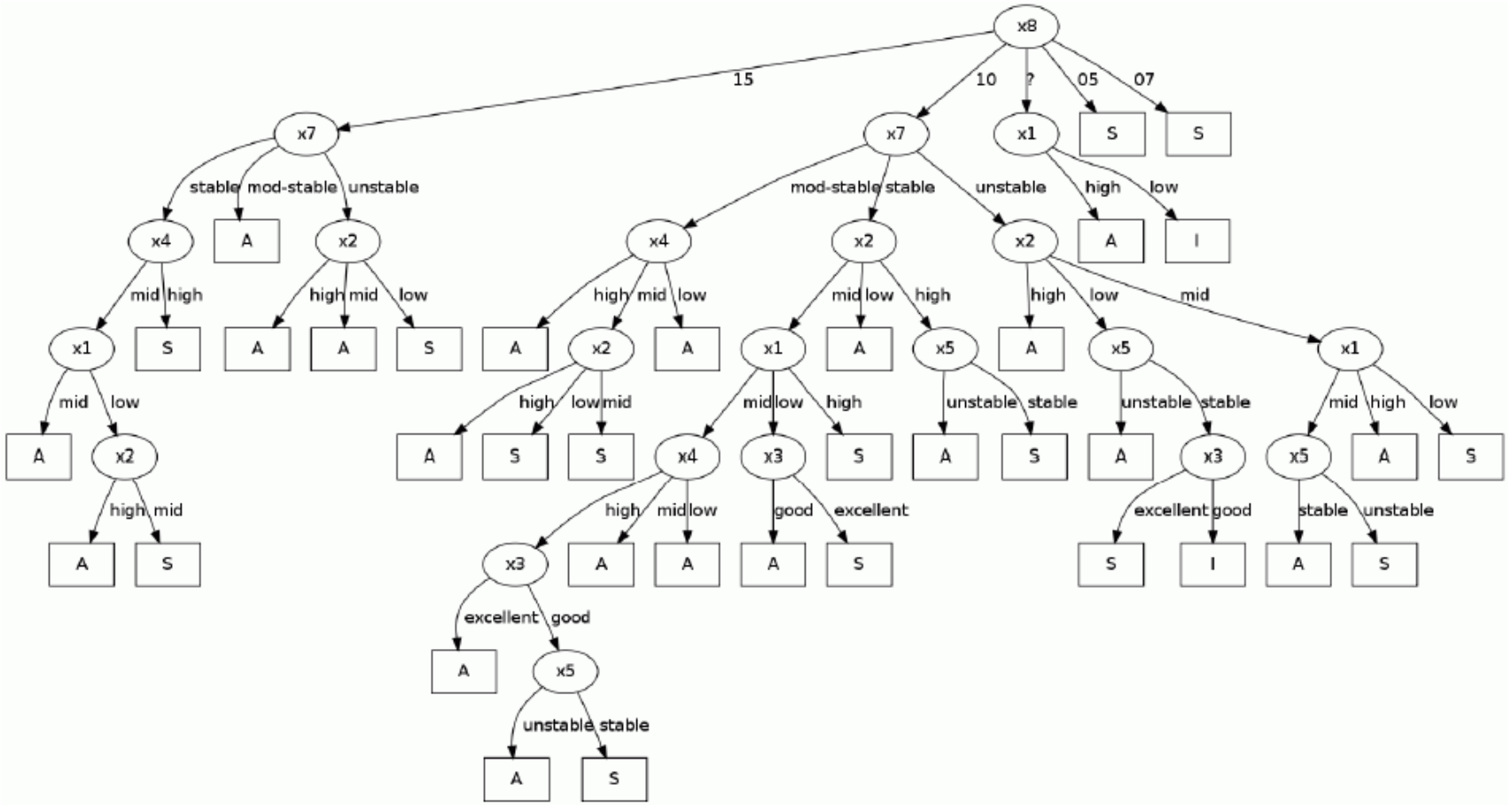
- Decision trees can be generated from production rules as well.
- Decision trees can thus be generated for the knowledge that was acquired from a domain expert, or learned using machine-learning methods.
- Decision tree learning is the construction of a decision tree from class-labelled training tuples (classification or decision tables).
- There are many specific decision-tree algorithms. Notable ones include:
  - ID3 (Iterative Dichotomiser 3)
  - C4.5 (successor of ID3)
  - CART (Classification And Regression Tree)
  - CHAID (CHi-squared Automatic Interaction Detector).
  - MARS: extends decision trees to handle numerical data better.
  - …

An Example Decision Table

T-CORE	T-SURF	O2	BP	T-SURF-S	T-CORE-S	BP-S	COMFORT	DECISION
mid	low	excellent	mid	stable	stable	stable	15	A
mid	high	excellent	high	stable	stable	stable	10	S
high	low	excellent	high	stable	stable	mod-stable	10	A
mid	low	good	high	stable	unstable	mod-stable	15	A
mid	mid	excellent	high	stable	stable	stable	10	A
high	low	good	mid	stable	stable	unstable	15	S
mid	low	excellent	high	stable	stable	mod-stable	5	S
high	mid	excellent	mid	unstable	unstable	stable	10	S
mid	high	good	mid	stable	stable	stable	10	S
mid	low	excellent	mid	unstable	stable	mod-stable	10	S
mid	low	excellent	mid	unstable	stable	mod-stable	10	S
mid	mid	good	mid	stable	stable	stable	15	A
mid	low	good	high	stable	stable	mod-stable	10	A
high	high	excellent	high	unstable	stable	unstable	15	A
…	…	…	…	…	…	…	…	…

The decision table for the postoperative medical decisions

An Example Decision Tree



The decision tree for the postoperative medical decisions

From Informal to Formal knowledge representation

- The fact expressed in the natural language

  „*Laura is driving to Seattle.*“

  can be stored in a computer database as the usual string of symbols.

- In this case, a computer program that enables search in the database can not give an answer to the question

  „*Who is driving to Seattle?*“

- The record of the given fact must thus be enriched with additional information that will allow the answers to such questions.

Semantic Content Enrichment

- The symbolic presentation of the facts is enriched by the introduction of additional semantic tags.

Semantic category	Meaning
Action:	'driving'
Actor:	'Laura'
Origin:	'?'
Goal:	'Seattle'
Time:	'present'
Means:	'?'

Semantic Content Enrichment

- Using an inference mechanism and another existing knowledge enables further enrichment of the semantic content.

Semantic category	Meaning	Additional semantic tag	
Action:	'driving'	(semantic types)	'motion'
Actor:	'Laura'	(semantic types)	'human'
Origin:	'?'	(default meaning)	'Laura's home'
Goal:	'Seattle'	(semantic types)	'city', 'city in U.S.'
Time:	'present'	(semantic types)	'tense'
Means:	'?'	(default meaning)	'car'

Semantic Networks

- Semantic networks are graphical representation for propositional information.
- They were originally developed by M. R. Quillian as a model for human memory.
- The are represented as labelled, directed graphs, where nodes represent objects, concepts, or situations, and links their relationships.
- The relationships contain the structural information of the knowledge to be represented.
- The label indicates the type of the relationship

Semantic Networks History

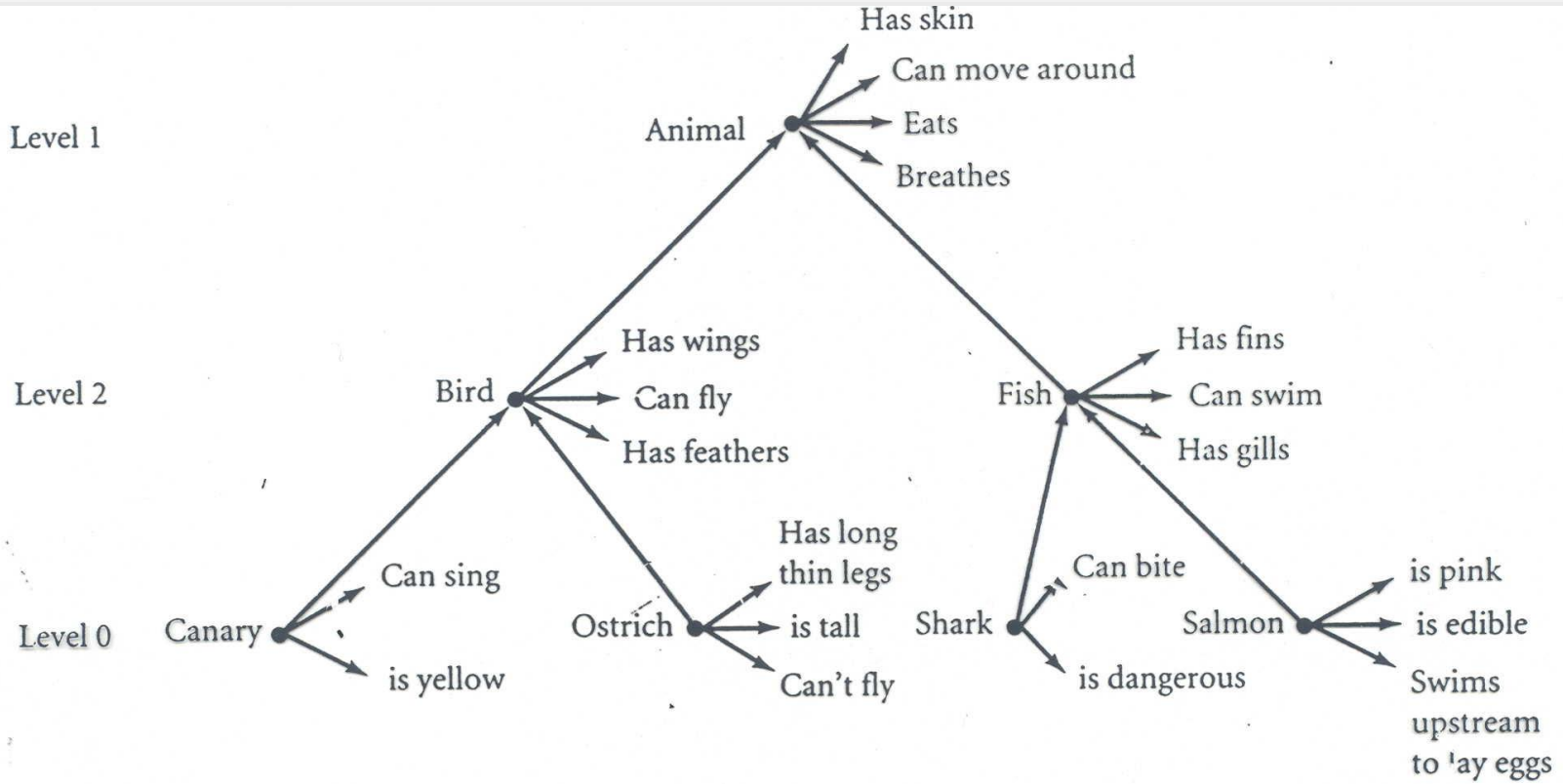
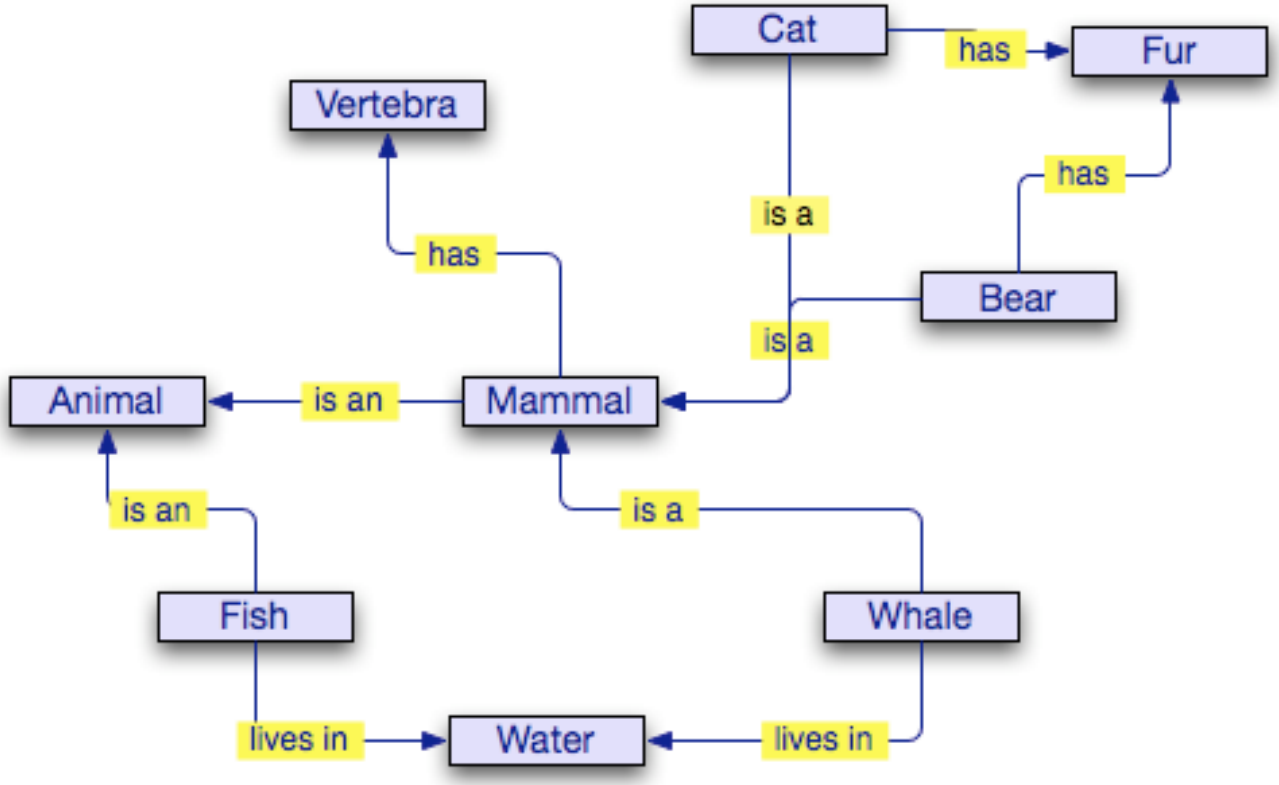


FIGURE 5.8: Hypothetical memory structure for a three-level hierarchy. (Adapted from Collins and Quillian, 1969. Reprinted by permission of Academic Press.)

Semantic Networks Motivation

- Instead of trying to recall that a canary flies, and a robin flies, and a swallow flies etc., humans remember that canaries, robins, swallows etc. are birds and that birds usually have an associated property of flying.
- More general properties, such as eating, breathing, moving etc. are stored at an even higher level associated with the concept animal.
- Thus reaction times to questions such as "Can a canary breathe?" were even longer again as more travel up the hierarchy is necessary to determine the answer.
- The fasted recall was for traits specific to the bird such as "Is a canary yellow?" or "Can a canary sing?"

An example of a Semantic Network



Problems with Semantic Networks

- Lack of naming standards for relationships.
- Often it is hard to distinguish between
  - statements about object relationships with the world, and
  - properties of the object.
- Unable to represent negation, quantification and disjunction.
- May result in large sets of nodes and links.
- Search may lead to combinatorial explosion.

Semantic Frames

- Semantic frames are a variant of semantic networks
- All the information relevant to a concept is stored in a single complex entity called a frame.
- Frames look like record data structures or class, however, at the very least frames also support inheritance.
- Slots can have related procedures that get executed when the value is added, modified or deleted
- Frames can be arranged in a hierarchy or graph

Semantic Frames

- In frame-based systems we refer to
  - objects – *Mammal*, *Elephant*, and *Nellie*;
  - slots – properties such as *colour* and *size*;
  - slot-values – values stored in the slots, e.g. *grey* and *large*.
- Slots and the corresponding slot-values are inherited through the class hierarchy.
- In general children classes inherit the properties of the parent class.
- Default values - properties that are typical for a class.
- Instances or subclasses whose properties differ from these default values are able to override them.

An Example of Semantic Frames

Mammal:

subclass: Animal
has-part: head

Elephant:

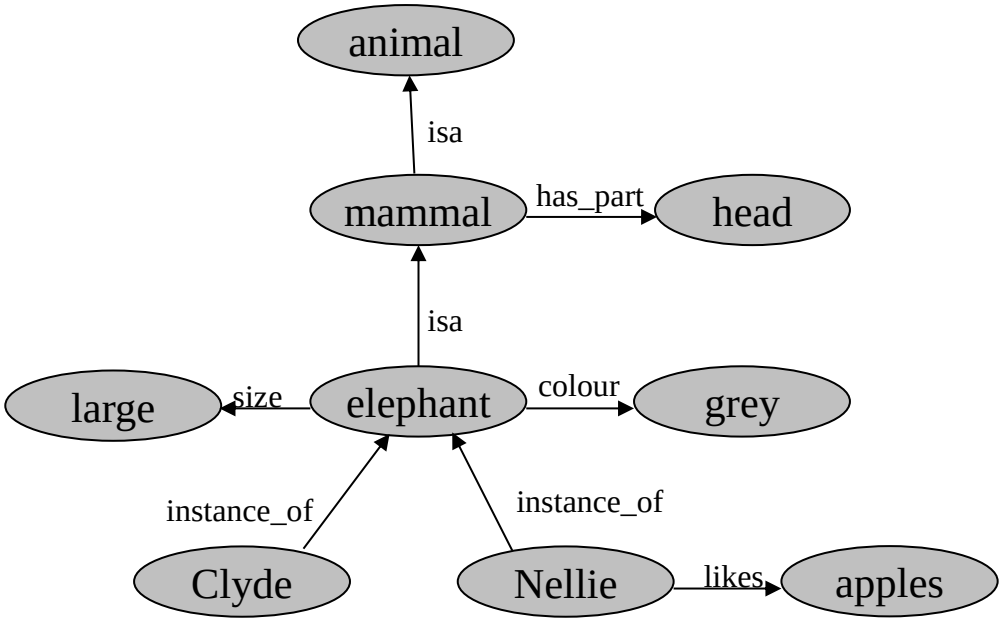
subclass: Mammal
colour: grey
size: large

Nellie:

instance: Elephant
likes: apples

Clyde:

instance: Elephant



Inference in semantic networks and frames

- Semantic networks and frames provide a fairly simple and clear way of representing the properties and categories of objects.
- A basic type of inference is defined whereby objects may inherit properties of parent objects.
- However, inheriting properties from more than one parent, or defining conflicting properties if often problematic.

The Semantic Web

- An effort to create a web that uses the concepts from semantic networks.
- It would allow people (and programs) to better understand and search web content.
- Two main representations at present:
  - RDF (Resource Description Framework) is a family of World Wide Web Consortium (W3C) specifications originally designed as a metadata data model.
  - OWL (Web Ontology Language) is a family of knowledge representation languages for authoring ontologies that add semantics to RDF.
  - SPARQL (Simple Protocol and Rdf Query Language) is a protocol and query language for semantic web data sources.

Semantic Web Languages

- RDF is a datamodel for objects ("resources") and relations between them (low level, triples – (node1, relationship, node2)). These datamodels can be represented in an XML syntax.
- RDFS (RDF Schema) is a vocabulary for describing properties (subclass, subproperty, domain, range, etc) and classes of RDF resources, with a semantics for generalization-hierarchies of such properties and classes.
- OWL adds more vocabulary for describing properties and classes: among others, relations between classes, cardinality (e.g. "exactly one"), equality, richer typing of properties, characteristics of properties (e.g. symmetry), and enumerated classes.
- OWL adds constraints on classes and the types of relationships permitted between them. These provide semantics by allowing systems (reasoners) to infer additional information and provide classification based on the data explicitly provided

RDF Examples

```
<?xml version="1.0"?>

<RDF>
  <Description about="http://www.w3schools.com/rdf">
    <author>Jan Egil Refsnes</author>
    <homepage>http://www.w3schools.com</homepage>
  </Description>
</RDF>
```

```
<?xml version="1.0"?>

<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:cd="http://www.recshop.fake/cd#">

<rdf:Description
rdf:about="http://www.recshop.fake/cd/Empire Burlesque">
  <cd:artist>Bob Dylan</cd:artist>
  <cd:country>USA</cd:country>
  <cd:company>Columbia</cd:company>
  <cd:price>10.90</cd:price>
  <cd:year>1985</cd:year>
</rdf:Description>
```

The **xmlns:cd** namespace, specifies that elements with the cd prefix are from the namespace "http://www.recshop.fake/cd#".

OWL Example

Example 3-4: XML Presentation Syntax for owlx:DifferentIndividuals

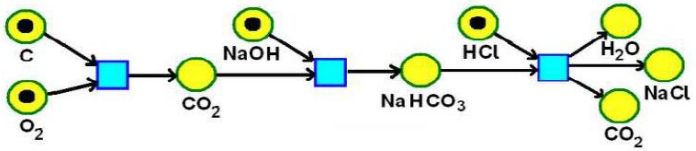
```
<owlx:Individual owlx:name="Dry">
  <owlx:type owlx:name="WineSugar" />
</owlx:Individual>

<owlx:Individual owlx:name="Sweet">
  <owlx:type owlx:name="WineSugar" />
</owlx:Individual>

<owlx:DifferentIndividuals>
  <owlx:Individual owlx:name="#Sweet" />
  <owlx:Individual owlx:name="#Dry" />
</owlx:DifferentIndividuals>
```

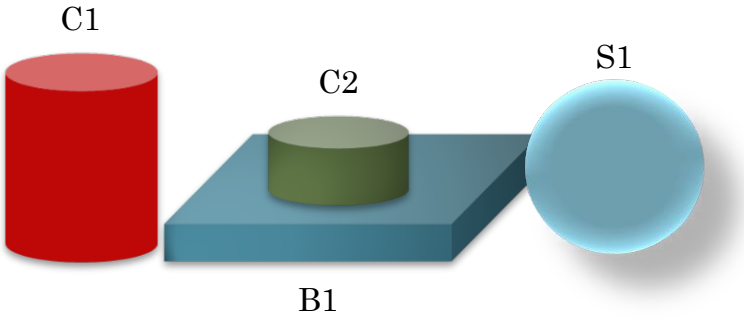
RDF/XML Syntax: owl:differentFrom (see also 4.3 in [OWL Guide])

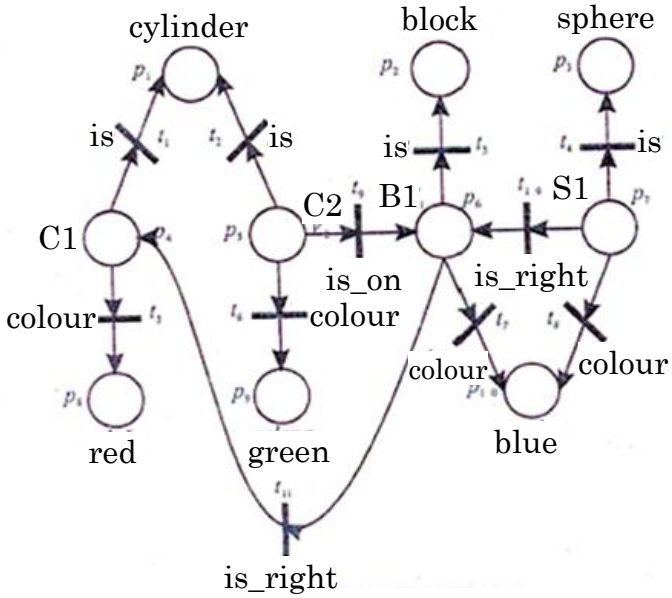
```
<WineSugar rdf:ID="Dry" />

<WineSugar rdf:ID="Sweet">
  <owl:differentFrom rdf:resource="#Dry"/>
</WineSugar>
```

Petri Nets

- Petri nets are used for modelling concurrent, distributed, asynchronous behaviour in a discrete system.
- These networks have proven to be a successful graphical and mathematical tool for modelling automated manufacturing systems, communication protocols as well as knowledge-based systems.







Questions

- Describe the progression of knowledge from data to wisdom.
- What is the difference between implicit and explicit knowledge and between procedural and declarative knowledge?
- Which knowledge representation schemes have been proposed?
- What is the Semantic Web?
- Give some examples of knowledge representation in a selected knowledge representation scheme.