GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

Module 1: Introduction to Artificial Intelligence and Knowledge based Systems

Course Title & Code: Principles of Artificial Intelligence/CSE 228

Semester: B.Tech

Department of Computer Science & Engineering,
School of Engineering, Presidency University,
Bengaluru

Contents

Artificial Intelligence

- What is AI ?
- Foundations of Artificial Intelligence
- History of Artificial Intelligence
- Applications of AI

Agents

- Agents: definition, structure and types
- Different Agent Types relation with environment

Definition and Importance of Knowledge

- Knowledge-Based Systems
- Representation of Knowledge
- Knowledge Organization
- Semantic Network

- Frame Structures
- Conceptual graphs

Introduction to Artificial Intelligence, Definitions, foundation, History and Applications

Introduction to Artificial Intelligence

- Homo Sapiens : The name is Latin for "*wise man*“

- Philosophy of AI - *“Can a machine think and behave like humans do?”*

- In Simple Words - *Artificial Intelligence is a way of* ***making a computer, a computer-controlled robot, or a software think intelligently****, in the similar manner the intelligent humans think.*

- **Artificial intelligence** (**AI**) is an area of computer science that emphasizes the creation of **intelligent** machines that work and react like humans.

- AI is accomplished by studying how human brain thinks, and how humans learn, decide, and work while trying to solve a problem, and then using the outcomes of this study as a basis of developing intelligent software and systems.

What is AI?

Views of AI fall into four categories:

1. Thinking humanly
2. Thinking rationally
3. Acting humanly
4. **Acting rationally**

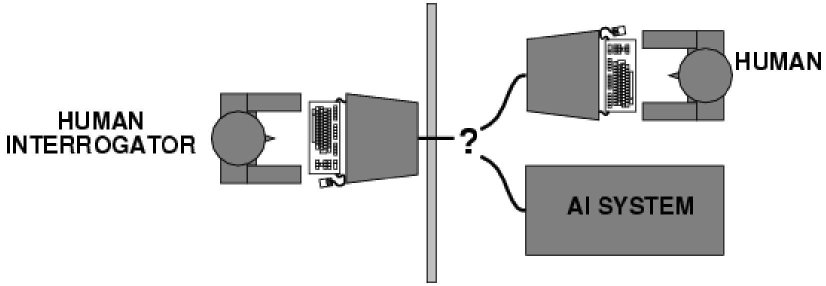
The textbook advocates "acting rationally"

What is AI?

Thinking Humanly “The exciting new effort to make computers think … machines with minds, in the full and literal sense.” “Activities that we associate with human thinking, activities such as decision-making, problem solving, learning…”	**Thinking Rationally** “The study of mental faculties through the use of computational models.” “The study of the computations that make it possible to perceive, reason and act.”
Acting Humanly “The art of creating machines that perform functions that require intelligence when performed by people.” “The study of how to make computers do things at which, at the moment, people are better.”	**Acting Rationally** “Computational Intelligence is the study of the design of intelligent agents.” “AI … is concerned with intelligent behavior in artifacts.”

Acting humanly: Turing Test

- Turing (1950) developed "Computing machinery and intelligence":
- "Can machines think?" "Can machines behave intelligently?"
- Operational test for intelligent behavior: **the Imitation Game**



A computer passes the test if a human interrogator, after posing some written questions, cannot tell whether the written responses come from a person or from a machine.

- Suggested major components of AI: knowledge, reasoning, language understanding, learning

Acting humanly: Turing Test

The computer would need to posses the following capabilities:

- **Natural Language Processing:**

  To enable it to communicate successfully in English.
- **Knowledge representation:**

  To store what it knows or hear.
- **Automated reasoning:**

  To use the stored information to answer questions and to draw new conclusions.
- **Computer vision:**

  To perceive objects.
- **Robotics:**

  To manipulate objects and move about.

Thinking humanly: Cognitive Modeling

- If we are going to say that ***given program thinks like a human***, we must have some way of determining how humans think.

- We need to get inside the actual working of human minds.

- There are 3 ways to do it:

  1. **Through introspection**

     Trying to catch our own thoughts as they go

  2. **Through psychological experiments**

     Observing a person in action

  3. **Through brain imaging**

     Observing the brain in action

Thinking humanly: Cognitive Modeling

- Once we have a sufficiently precise theory of the mind, it becomes possible to express the theory as a computer program.

- If the program’s input-output behavior matches corresponding human behavior, that is evidence that the program’s mechanisms could also be working in humans

Thinking Rationally: “Laws of Thought"

- Aristotle: one of the first to attempt to codify **“right thinking”**. Mathematical representation.

- His **syllogisms** provided patterns for **argument structures** that always yielded correct conclusions when given premises are correct.

Example – Socrates is a man

All men are mortal

Therefore

Socrates is mortal

Acting Rationally: Rational Agent

- An agent is an entity that perceives and acts
- A system is rational if it does the “right thing,” given what it knows.
- This course is about designing rational agents
- Rational agent is one that acts so as to achieve the best outcome or, when there is uncertainty, the best expected outcome.

- Abstractly, an agent is a function from percept histories to actions:

$$[f: P^{*} \boxed{?} A]$$

Behave Rationally……….

- **What means "behave rationally" for a person/system:**
  - Take the right/ best action to achieve the goals, based on his/its knowledge and belief

  Example: Assume I don't like to get wet in rain (my goal), so I bring an umbrella (my action). Do I behave rationally?

  - The answer is dependent on my knowledge and belief
  - If I've heard the forecast for rain and I believe it, then bringing the umbrella is rational.
  - If I've not heard the forecast for rain and I do not believe that it is going to rain, then bringing the umbrella is not rational

- **"Behave rationally" does not always achieve the goals successfully**

  Example:

  - My goals – (i) do not get wet if rain; (ii) do not looked stupid (such as bring an umbrella when not raining)
  - My knowledge/belief – weather forecast for rain and I believe it
  - My rational behaviour – bring an umbrella
  - The outcome of my behaviour: If rain, then my rational behaviour achieves both goals; If no rain, then my rational behaviour fails to achieve the 2nd goal

- **The successfulness of "behave rationally" is limited by my knowledge and belief**

Cryptarithmetic

Solve the following crypt arithmetic:

```
   T W O
 + T W O
 --------
 F O U R
```

Solution :

Variables: F,T,U,W,R,O,C1,C2,C3.

Domains: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9} (same domain for all)

Sample constraints: alldif (F,T,U,W,R,O) or F ≠T , F≠ U.

Rules for Cryptarithmetic Problem

Digit Ranges From **0 to 9**.

Each Variables should have **Unique value**.

Each Letter symbol represents **only one digit** throughout the problem.

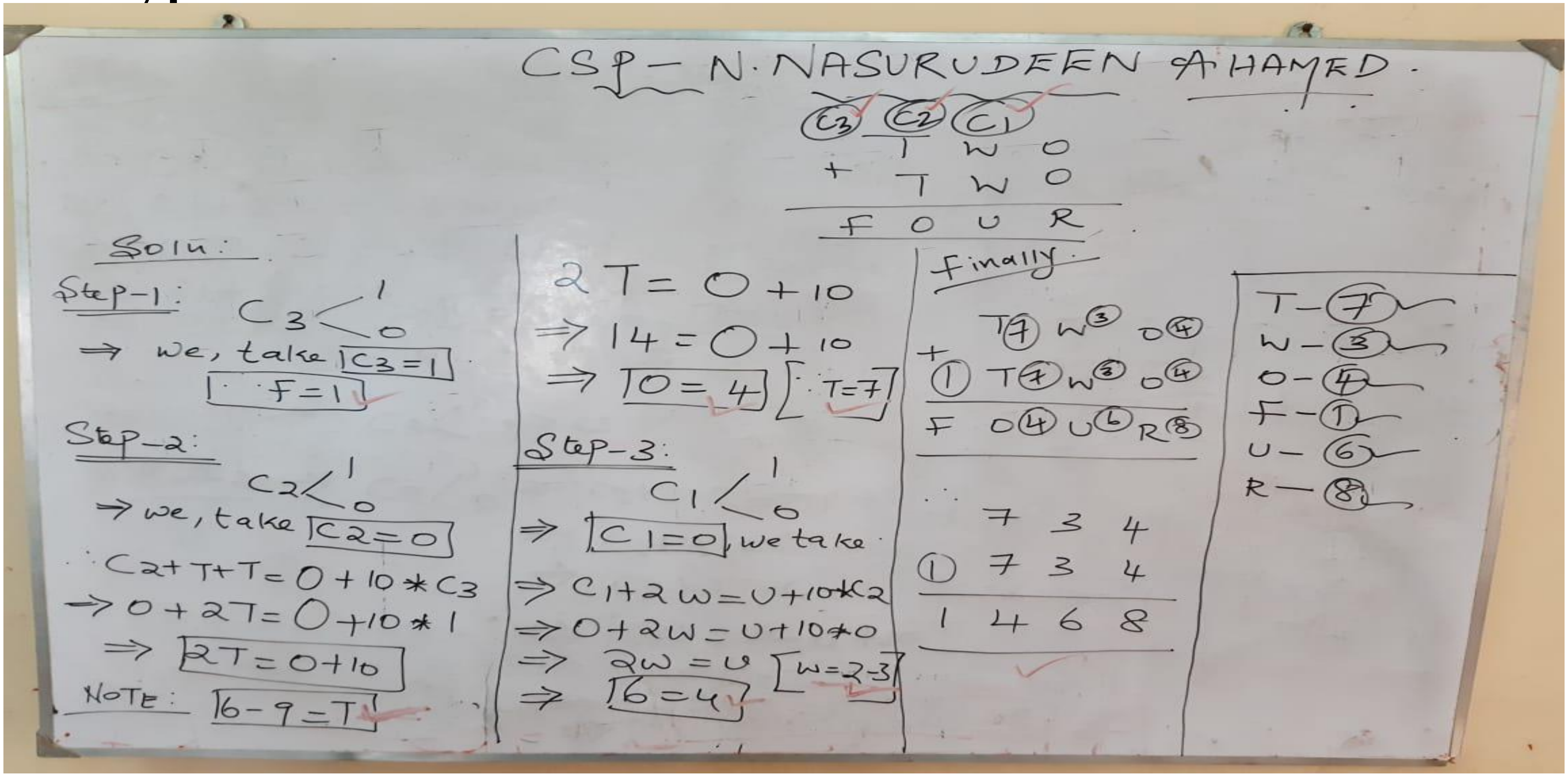
You have to find the value of letter in the CSP.

There must be **only one solution** to the problem.

The numerical base unless specifically stated is 10.

Numbers must **not begin** with **zero.**

Cryptarithmetic



CSP – N. NASURUDEEN AHAMED.

```
    C3  C2  C1
        T   W   O
  +     T   W   O
  ---------------
    F   O   U   R
```

Soln:

Step-1: $C_3 < {}^{1}_{0}$

⇒ we, take $\boxed{C_3 = 1}$

$\boxed{\therefore F = 1}$ ✓

Step-2: $C_2 < {}^{1}_{0}$

⇒ we, take $\boxed{C_2 = 0}$

$\therefore C_2 + T + T = O + 10 * C_3$

$\Rightarrow 0 + 2T = O + 10 * 1$

$\Rightarrow \boxed{2T = O + 10}$

NOTE: $\boxed{6 - 9 = T}$ ✓

$2T = O + 10$

$\Rightarrow 14 = O + 10$

$\Rightarrow \boxed{O = 4}$ ✓ $[\because T = 7]$ ✓

Step-3: $C_1 < {}^{1}_{0}$

$\Rightarrow \boxed{C_1 = 0}$, we take

$\Rightarrow C_1 + 2W = U + 10 * C_2$

$\Rightarrow 0 + 2W = U + 10 * 0$

$\Rightarrow 2W = U$ $[W = 2\text{-}3]$ ✓

$\Rightarrow \boxed{6 = 4}$ ✓

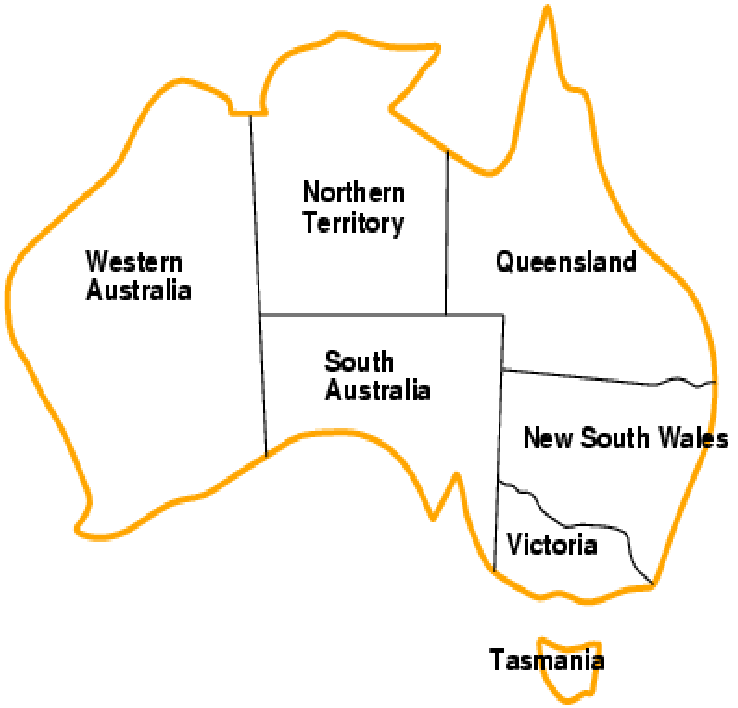
Finally.

```
       T(7)  W(3)  O(4)
  + (1) T(7)  W(3)  O(4)
  ----------------------
   F   O(4)  U(6)  R(8)
  ----------------------
  ∴      7    3    4
     (1) 7    3    4
  ----------------------
     1   4    6    8
  ----------------------
```

✓

T –	7
W –	3
O –	4
F –	1
U –	6
R –	8

Map – Coloring Problem



Hints :

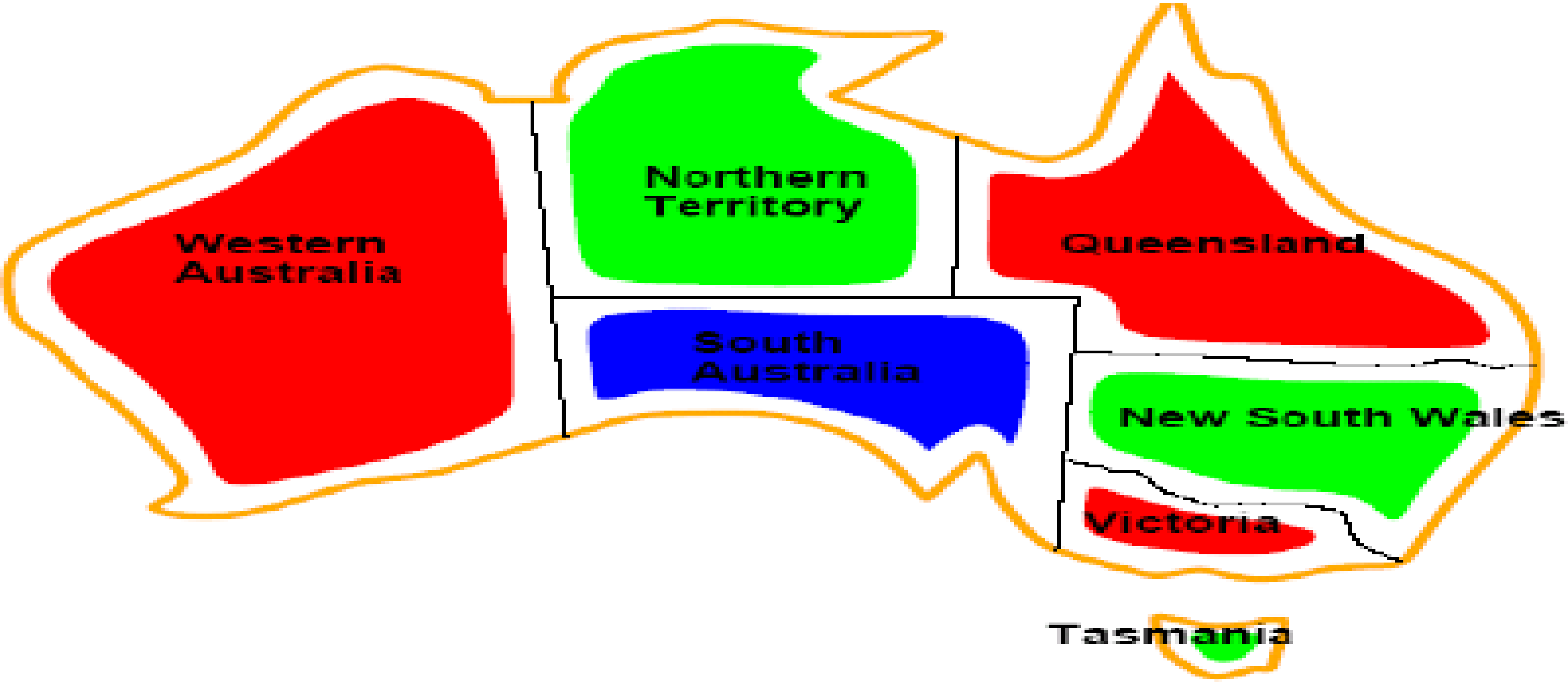
Variables ***WA, NT, Q, NSW, V, SA, T***

Domains Di = {red,green,blue}

Constraints: **adjacent regions must have different colors**
e.g., WA ≠ NT

Notes : Western Australia (WA)

Map – Coloring Problem



Definition of AI

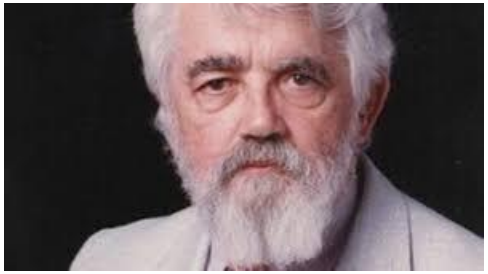
- Existing definitions advocate everything from replicating human intelligence to simply solving knowledge-intensive tasks.

Examples:

“Artificial Intelligence is the design, study and construction of computer programs that behave intelligently.” -- Tom Dean.

“Artificial Intelligence is the enterprise of constructing a physical symbol system that can reliably pass the Turing test.” -- Matt Ginsberg.

History of AI



- 1943 McCulloch & Pitts developed **Boolean circuit model of brain**
- 1950 Turing's **"Computing Machinery and Intelligence"**
- 1956 Dartmouth meeting: **"Artificial Intelligence"** adopted
- 1952—69 McCarthy referred **“Look, Ma, no hands!”** era People thought “Only arithmetic can be done and no more before 1952”. Astonishing if something is done remotely clever.
- 1950s Early AI programs, including
  - » **Samuel's checkers program**,
  - » **Newell & Simon's Logic Theorist** : It was the first program deliberately engineered to mimic the problem solving skills of a human being and is called "the first artificial intelligence program". It would eventually prove 38 of the first 52 theorems in Whitehead and Russell's *Principia Mathematica*
  - » **Gelernter's Geometry Engine**
- 1965 Robinson's complete algorithm for logical reasoning

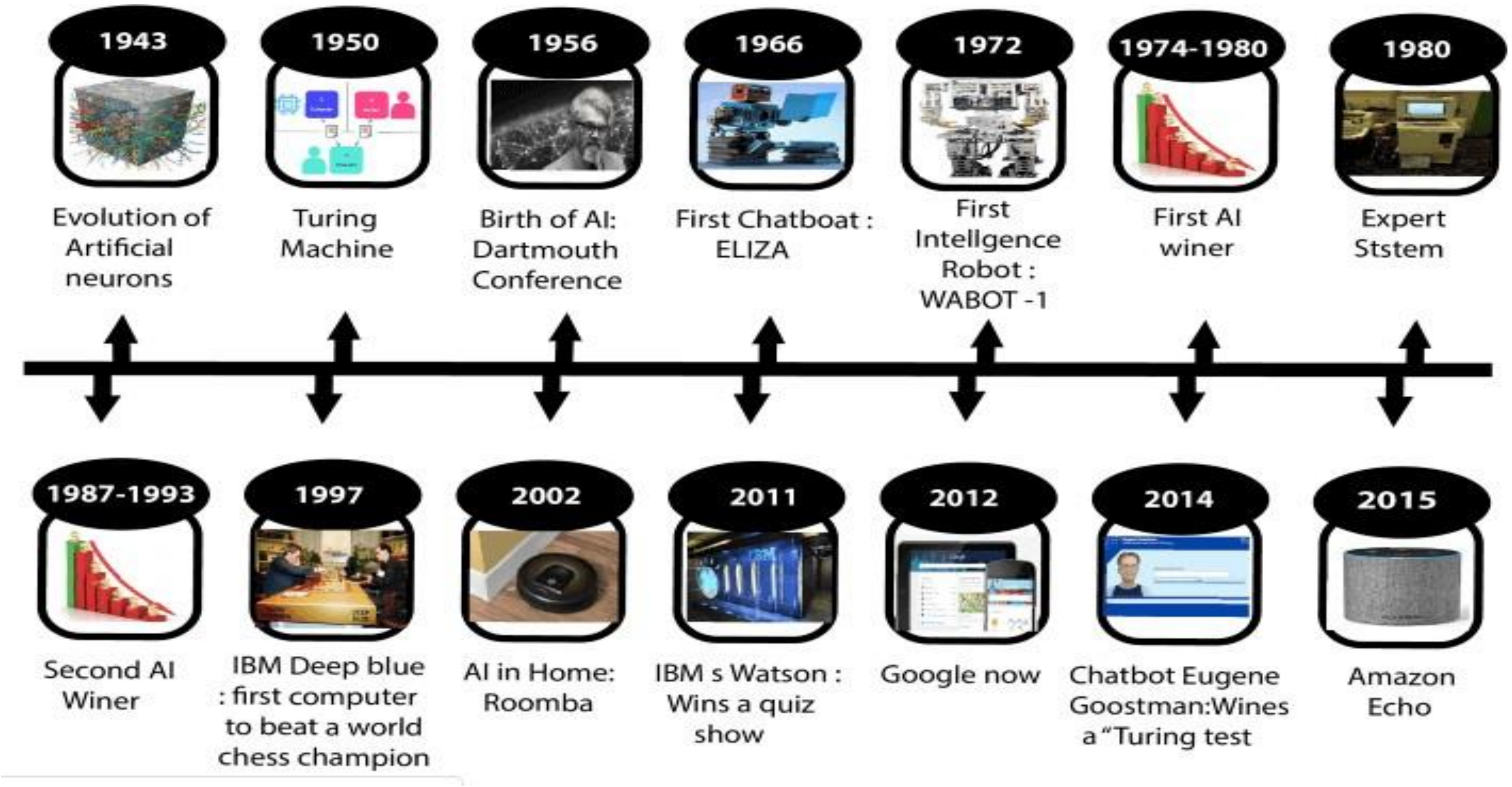
AI Winter

- Funding Reduces
- P Vs NP
- Exponential time and complexity explosions
- Limited Computational Capabilities
  - RAM
  - Processing Power
  - Data
- Moravec's paradox@ CMU
  - High-level reasoning requires very little computation, but low- level subconscious tasks require enormous computational resources

History of AI

- 1966—73 AI discovers **computational complexity Neural network research** almost **disappears**
- 1969—79 Early development of **knowledge-based systems**
- 1980-- AI becomes an **industry**
- 1986-- **Neural networks** return to popularity AI
- 1987-- becomes a **science**
- 1995-- The emergence of **intelligent agents**

History of AI



1943
Evolution of Artificial neurons

1950
Turing Machine

1956
Birth of AI: Dartmouth Conference

1966
First Chatboat : ELIZA

1972
First Intellgence Robot : WABOT -1

1974-1980
First AI winer

1980
Expert Ststem

1987-1993
Second AI Winer

1997
IBM Deep blue : first computer to beat a world chess champion

2002
AI in Home: Roomba

2011
IBM s Watson : Wins a quiz show

2012
Google now

2014
Chatbot Eugene Goostman:Wines a "Turing test

2015
Amazon Echo

Ref : https://www.javatpoint.com/history-of-artificial-intelligence

Foundations of Artificial Intelligence

Below are the disciplines that contributed **ideas**, **viewpoints** and **techniques** to AI:

1. Philosophy
2. Mathematics
3. Economics
4. Neuroscience
5. Psychology
6. Computer Engineering
7. Control theory
8. Linguistics

Foundations of Artificial Intelligence

- Philosophy

  Can formal rules be used to draw valid conclusions?
  How does the mind arise from a physical brain?
  Where does knowledge come from?
  How does knowledge lead to action?

Aristotle was the first to formulate a precise set of laws governing the **rational part of the mind**. He developed an informal system of **syllogisms** for **proper reasoning**, which in principle allowed one to generate **conclusions** mechanically, given initial **premises**.

All dogs are animals; all animals have four legs; therefore all dogs have four legs

Descartes was a strong advocate of the **power of reasoning** in understanding the world, philosophy now called as **rationalism**.

Foundations of Artificial Intelligence

- Mathematics

  What are the formal rules to draw valid conclusions?
  What can be computed?
  How do we reason with uncertain information?

Formal representation and proof algorithms: Propositional logic
Computation: Turing tried to characterize exactly which functions are computable - capable of being computed.

(un)decidability: Incompleteness theory showed that in any formal theory, there are **true statements that are undecidable** i.e. they have no proof within the theory.

“ a line can be extended infinitely in both directions”

(in)tractability: A problem is called intractable if the time required to solve instances of the problem grows exponentially with the size of the instance. **probability:** Predicting the future.

Foundations of Artificial Intelligence

- **Economics**

How should we make decisions so as to maximize payoff?

Economics is the study of how people make choices that lead to **preferred outcomes**(utility).

Decision theory: It combines **probability theory** with **utility theory**, provides a formal and complete framework for decisions made under uncertainty.

- **Neuroscience**

How do brains process information?

Neuroscience is the study of the **nervous system,** particularly brain.

Brain consists of nerve cells or **neurons**. 10^11 neurons.

Neurons are considered as **Computational units**.

Foundations of Artificial Intelligence

- **Psychology**

  How do Humans and animals think and act?

- **Computer engineering**

  How can we build an efficient computer?

  Building fast computers

- **Control theory**

  How can artifacts operate under their own control?

  Design systems that maximize an objective function over time

- **Linguistics**

  How does the language relate to thought?

  knowledge representation, grammar

Applications of AI

Applications:

- **Deep Blue(chess-playing computer)** defeated the world chess champion Garry Kasparov in 1997
- During the 1991 Gulf War, US forces deployed an **AI logistics planning and scheduling program** that involved up to 50,000 vehicles, cargo, and people

  Planning – How to use resources? Scheduling – When to use the resources?
- NASA's on-board autonomous planning program controlled the **scheduling of operations for a spacecraft**

- **Google duplex**
- The GPS developed in 1957 by Alan Newell and Hervert Simon, embodied a grandiose vision

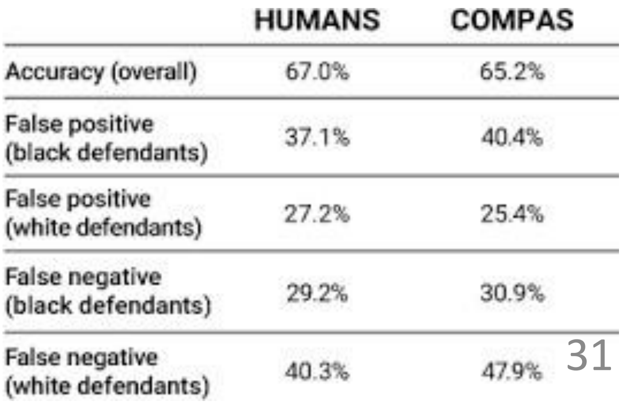
Future Perspective

(1) Reducing the time and cost of development is a big plan for AI.
(2) To develop applications towards strong AI.
(3) Allowing students to work collaboratively is another plan from Researchers.

Perfect rationality: the classical notion of rationality in decision theory.
Bounded optimality: A bounded optimal agent behaves as well as possible given its computational resources.
Game theory studies decision problems in which the utility of a given action depends not only on chancing events in the environment but also on the actions of other agents.

Major Concerns

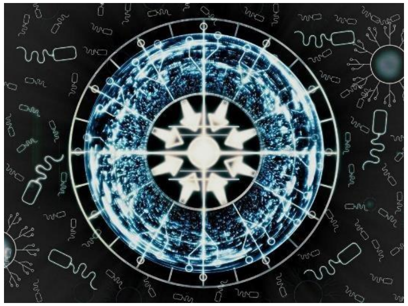
TAY



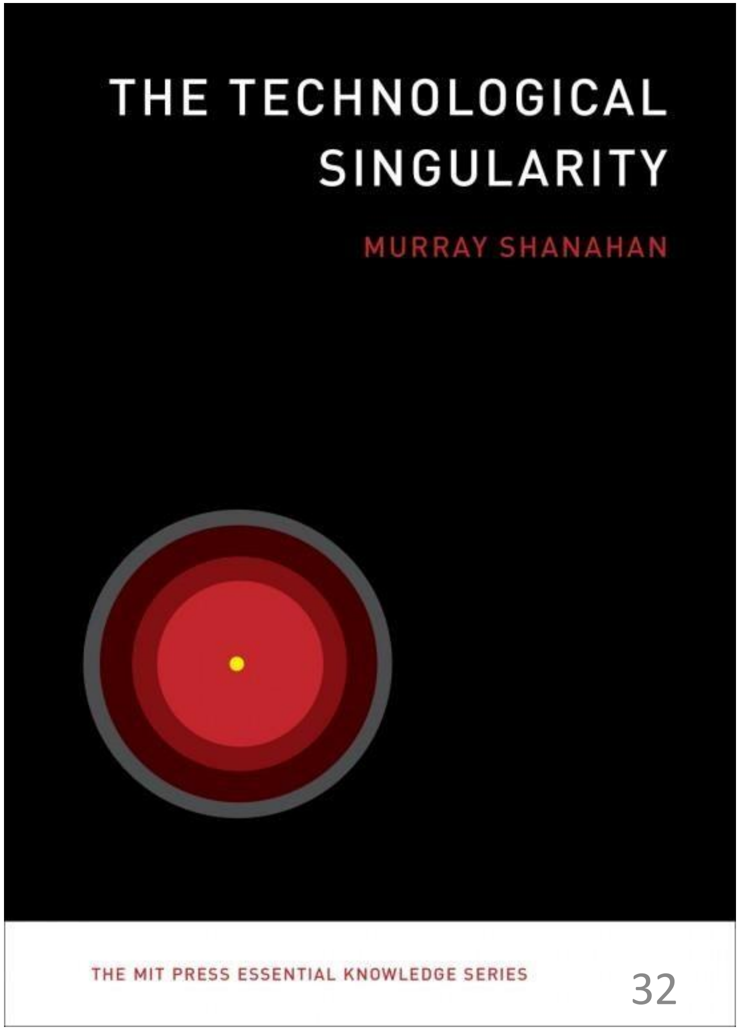
	HUMANS	COMPAS
Accuracy (overall)	67.0%	65.2%
False positive (black defendants)	37.1%	40.4%
False positive (white defendants)	27.2%	25.4%
False negative (black defendants)	29.2%	30.9%
False negative (white defendants)	40.3%	47.9%

COMPAS - Correctional Offender Management Profiling for Alternative Sanctions

Singularity



Singularity is a hypothetical future point in time at which technological growth becomes uncontrollable and irreversible, resulting in unfathomable changes to human civilization



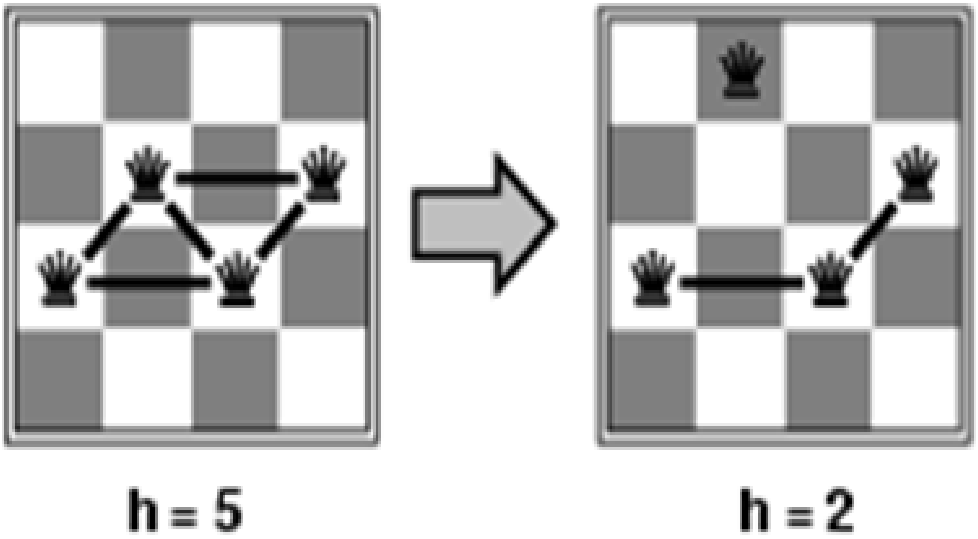
THE TECHNOLOGICAL
SINGULARITY
MURRAY SHANAHAN

THE MIT PRESS ESSENTIAL KNOWLEDGE SERIES

Hints:

States: **4 queens in 4 columns (4^4 = 256 states)**

Goal test: **no attacks**



AGENTS

Agents in Artificial Intelligence

Artificial intelligence is defined as a study of rational agents. A rational agent could be anything which makes decisions, as a person, firm, machine, or software. It carries out an action with the best outcome after considering past and current percepts(agent's perceptual inputs at a given instance).

An AI system is composed of an **agent and its environment**. The agents act in their environment. The environment may contain other agents. An agent is anything that can be viewed as :

- perceiving its environment through **sensors** and
- acting upon that environment through **actuators**

The Structure of Intelligent Agents

Agent’s structure can be viewed as −

- Agent = Architecture + Agent Program
- Architecture = the machinery that an agent executes on.
- Agent Program = an implementation of an agent function.

The Structure of Intelligent Agents

To understand the structure of Intelligent Agents, we should be familiar with *Architecture* and *Agent Program*. **Architecture** is the machinery that the agent executes on. It is a device with sensors and actuators, for example : a robotic car, a camera, a PC. **Agent program** is an implementation of an agent function. An **agent function** is a map from the percept sequence(history of all that an agent has perceived till date) to an action.

Agent Terminology

Performance Measure of Agent − It is the criteria, which determines how successful an agent is.

Behavior of Agent − It is the action that agent performs after any given sequence of percepts.

Percept − It is agent’s perceptual inputs at a given instance.

Percept Sequence − It is the history of all that an agent has perceived till date.

Agent Function − It is a map from the precept sequence to an action.

Rationality

Rationality is nothing but status of being reasonable, sensible, and having good sense of judgment.

Rationality is concerned with expected actions and results depending upon what the agent has perceived. Performing actions with the aim of obtaining useful information is an important part of rationality.

What is Ideal Rational Agent?

An ideal rational agent is the one, which is capable of doing expected actions to maximize its performance measure, on the basis of −

- Its percept sequence
- Its built-in knowledge base

Rationality of an agent depends on the following −

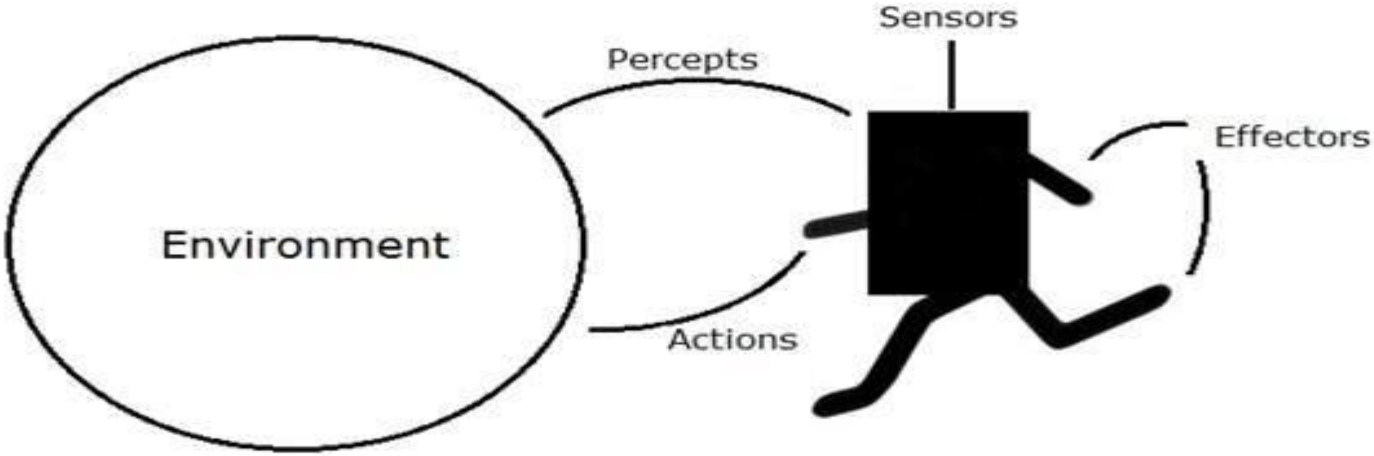
- The **performance measures**, which determine the degree of success.
- Agent's **Percept Sequence** till now.
- The agent's **prior knowledge about the environment**.
- The **actions** that the agent can carry out.

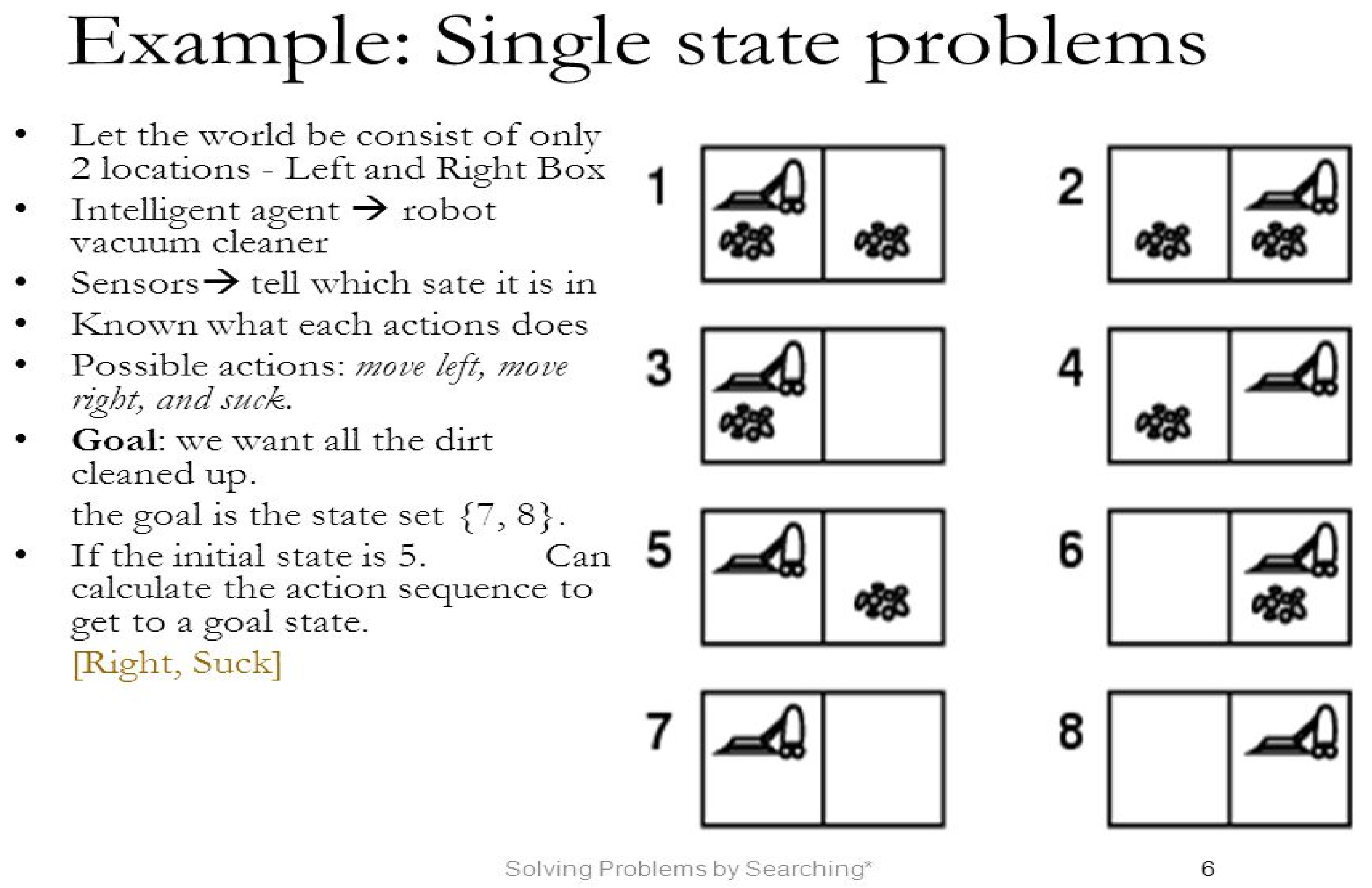
A rational agent always performs right action, where the right action means the action that causes the agent to be most successful in the given percept sequence. The problem the agent solves is characterized by Performance Measure, Environment, Actuators, and Sensors (PEAS).

Examples of Agent:-

An **agent** is anything that can perceive its environment through **sensors** and acts upon that environment through **effectors.**

- A **human agent** has sensory organs such as eyes, ears, nose, tongue and skin parallel to the sensors, and other organs such as hands, legs, mouth, for effectors.
- A **robotic agent** replaces cameras and infrared range finders for the sensors, and various motors and actuators for effectors.
- A **software agent** has encoded bit strings as its programs and actions.





Example: Single state problems

- Let the world be consist of only 2 locations - Left and Right Box
- Intelligent agent → robot vacuum cleaner
- Sensors→ tell which sate it is in
- Known what each actions does
- Possible actions: *move left, move right, and suck*.
- **Goal**: we want all the dirt cleaned up.

  the goal is the state set {7, 8}.
- If the initial state is 5. Can calculate the action sequence to get to a goal state.

  [Right, Suck]

Types of Agents

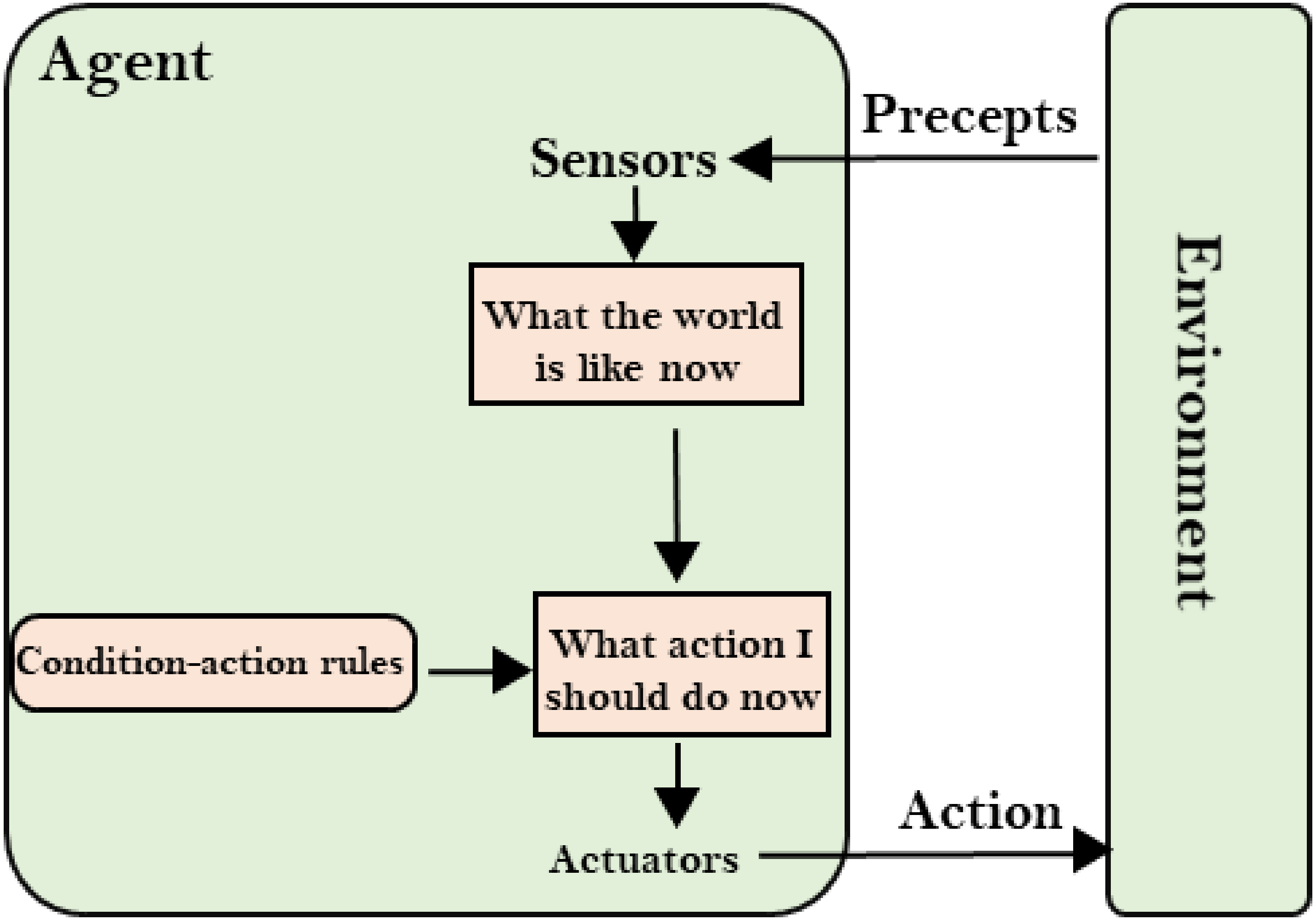
Agents can be grouped into five classes based on their degree of perceived intelligence and capability. All these agents can improve their performance and generate better action over the time. These are given below:

- Simple Reflex Agent
- Model-based reflex agent
- Goal-based agents
- Utility-based agent
- Learning agent

Simplex Agent

- The Simple reflex agents are the simplest agents. These agents take decisions on the basis of the current percepts and ignore the rest of the percept history.
- These agents only succeed in the fully observable environment.
- The Simple reflex agent does not consider any part of percepts history during their decision and action process.
- The Simple reflex agent works on Condition-action rule, which means it maps the current state to action. Such as a Room Cleaner agent, it works only if there is dirt in the room.
- Problems for the simple reflex agent design approach:
  - They have very limited intelligence
  - They do not have knowledge of non-perceptual parts of the current state
  - Mostly too big to generate and to store.
  - Not adaptive to changes in the environment.

Simplex Agent



Model-based reflex agent

The Model-based agent can work in a partially observable environment, and track the situation.

A model-based agent has two important factors:

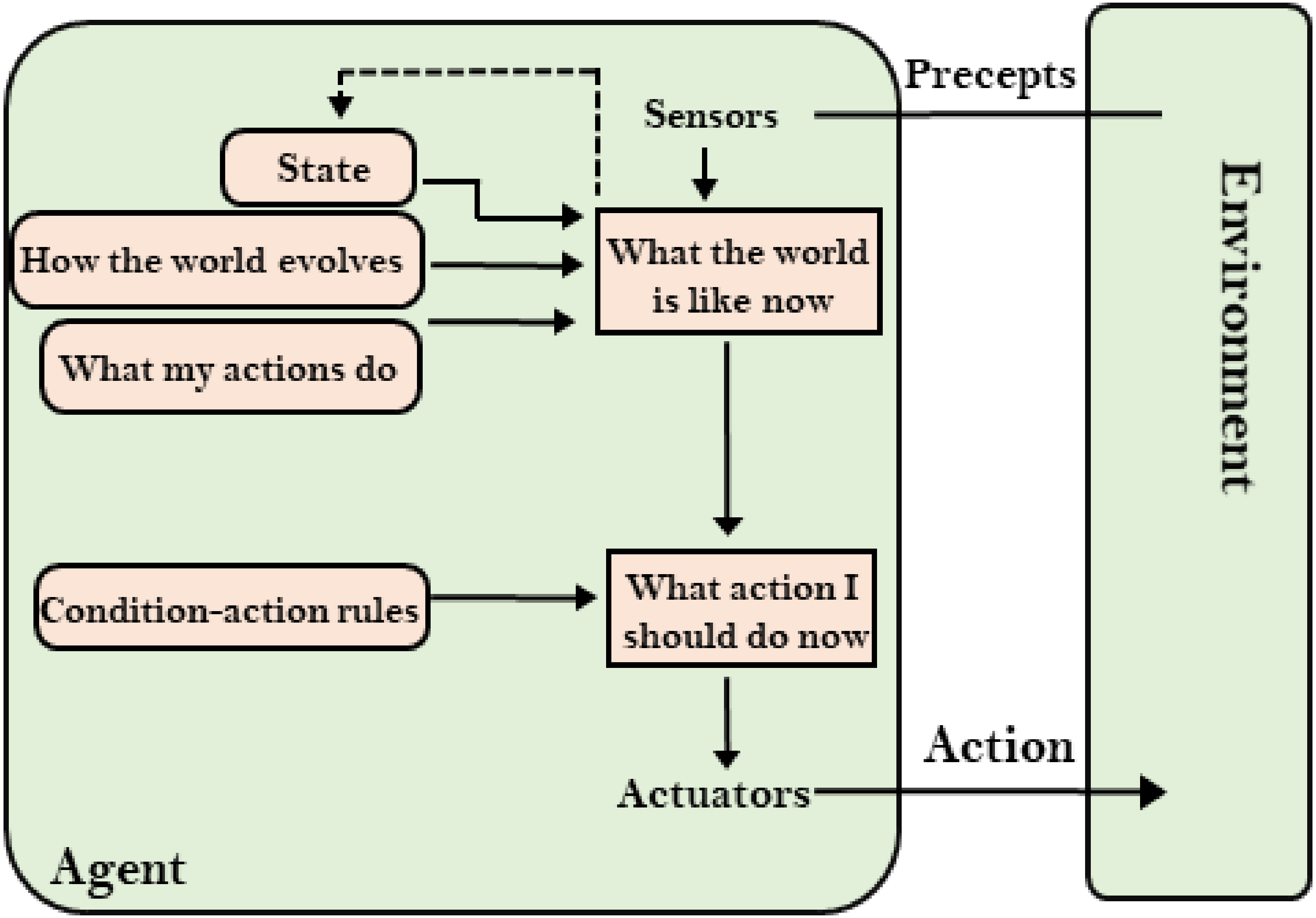
- **Model:** It is knowledge about "how things happen in the world," so it is called a Model-based agent.
- **Internal State:** It is a representation of the current state based on percept history.

These agents have the model, "which is knowledge of the world" and based on the model they perform actions.

Updating the agent state requires information about:

- How the world evolves
- How the agent's action affects the world.

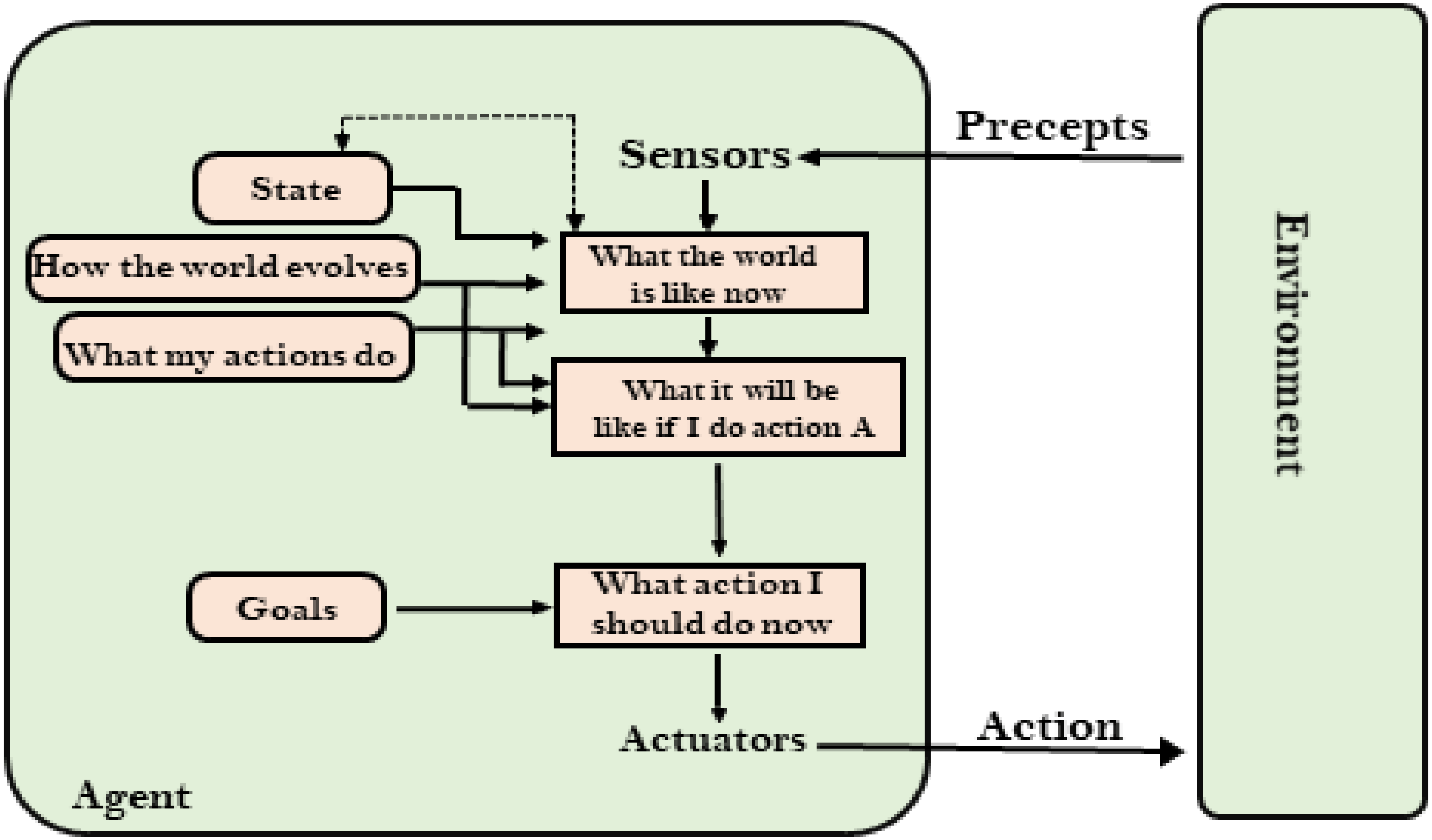
Model-based reflex agent



Goal-based agents

- The knowledge of the current state environment is not always sufficient to decide for an agent to what to do.
- The agent needs to know its goal which describes desirable situations.
- Goal-based agents expand the capabilities of the model-based agent by having the "goal" information.
- They choose an action, so that they can achieve the goal.
- These agents may have to consider a long sequence of possible actions before deciding whether the goal is achieved or not. Such considerations of different scenario are called searching and planning, which makes an agent proactive.

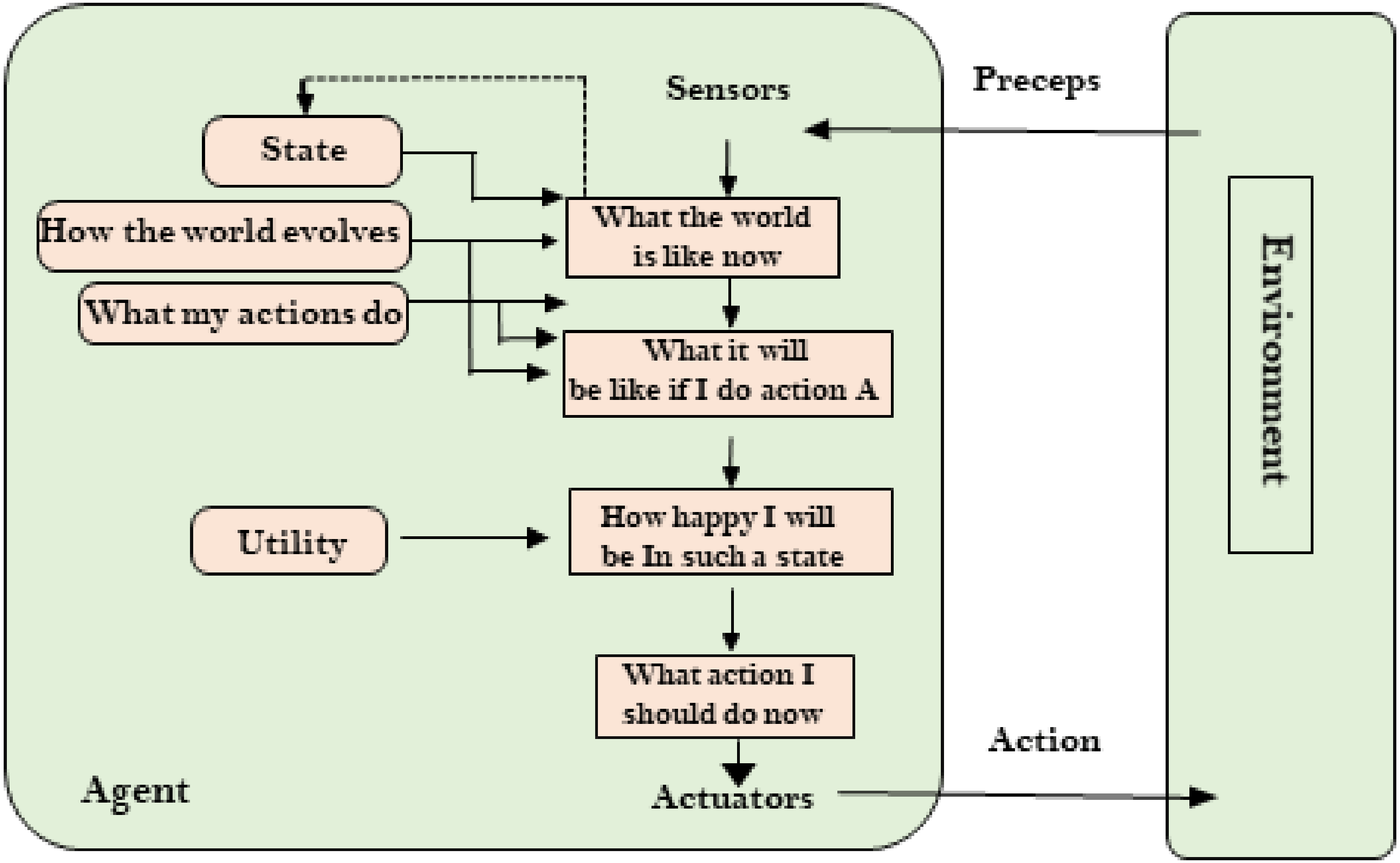
Goal-based agents



Utility-based agents

- These agents are similar to the goal-based agent but provide an extra component of utility measurement which makes them different by providing a measure of success at a given state.
- Utility-based agent act based not only goals but also the best way to achieve the goal.
- The Utility-based agent is useful when there are multiple possible alternatives, and an agent has to choose in order to perform the best action.
- The utility function maps each state to a real number to check how efficiently each action achieves the goals.

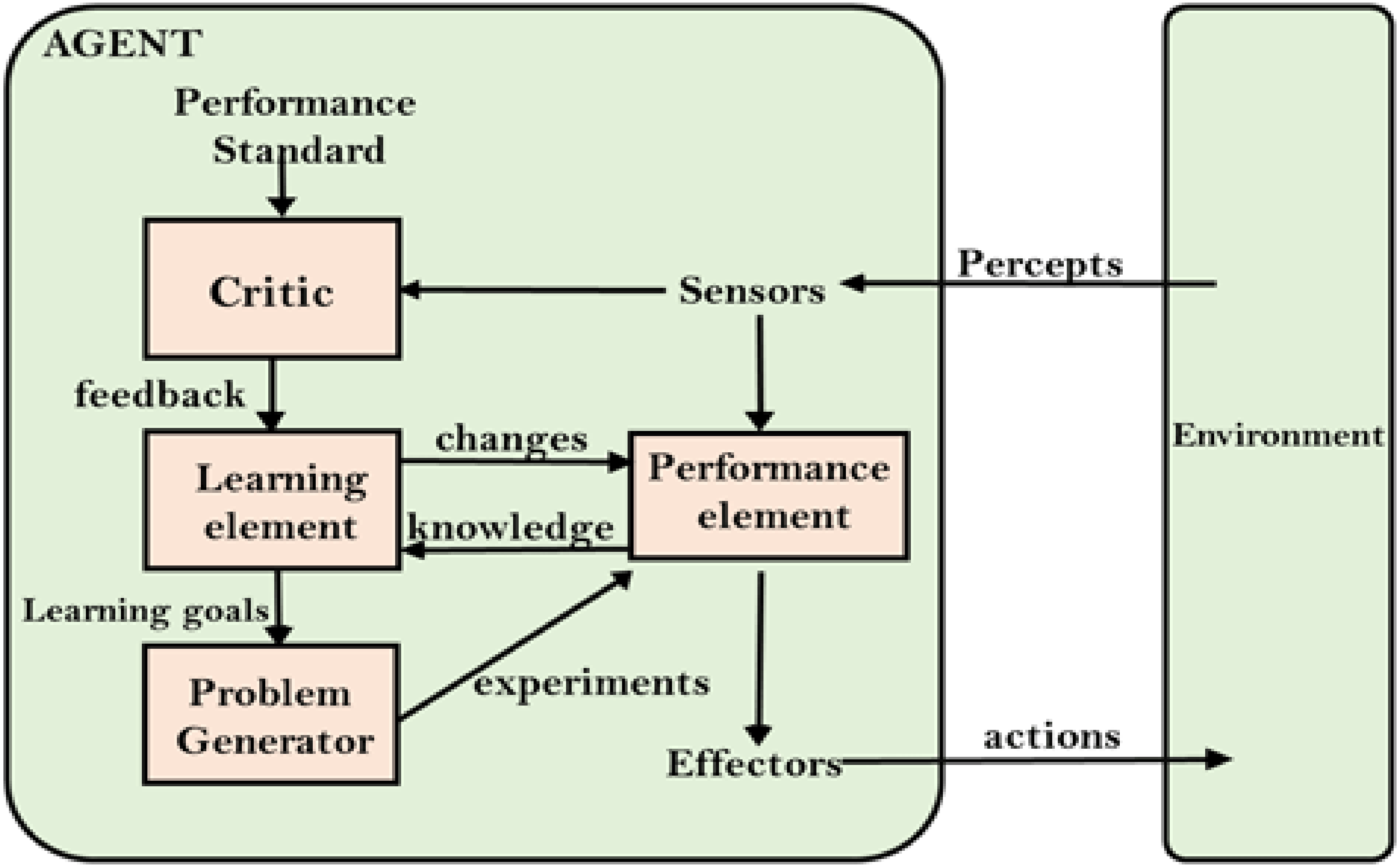
Utility-based agents


Sensors
Preceps
State
How the world evolves
What my actions do
What the world is like now
What it will be like if I do action A
Utility
How happy I will be In such a state
What action I should do now
Action
Agent
Actuators
Environment

Learning Agents

- A learning agent in AI is the type of agent which can learn from its past experiences, or it has learning capabilities.
- It starts to act with basic knowledge and then able to act and adapt automatically through learning.
- A learning agent has mainly four conceptual components, which are:
  - **Learning element:** It is responsible for making improvements by learning from environment
  - **Critic:** Learning element takes feedback from critic which describes that how well the agent is doing with respect to a fixed performance standard.
  - **Performance element:** It is responsible for selecting external action
  - **Problem generator:** This component is responsible for suggesting actions that will lead to new and informative experiences.
- Hence, learning agents are able to learn, analyze performance, and look for new ways to improve the performance.

Learning Agents



KNOWLEDGE

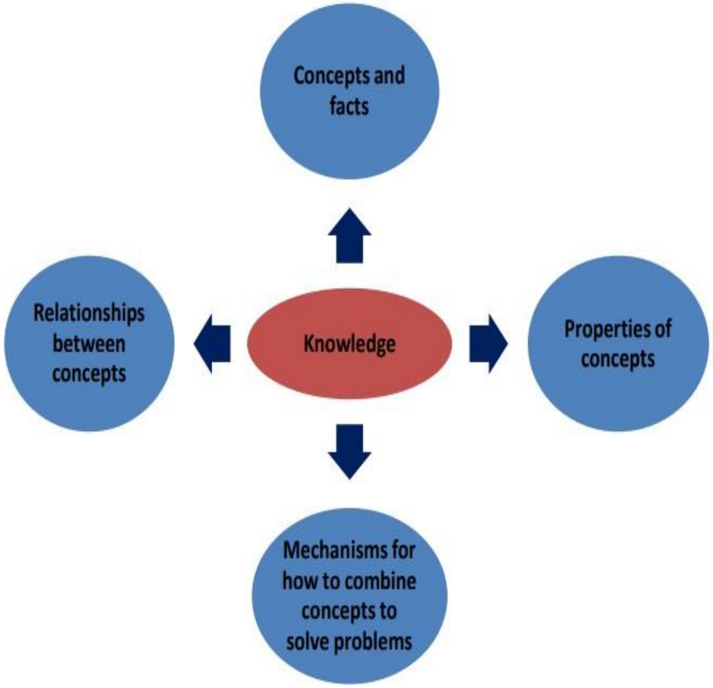
Knowledge

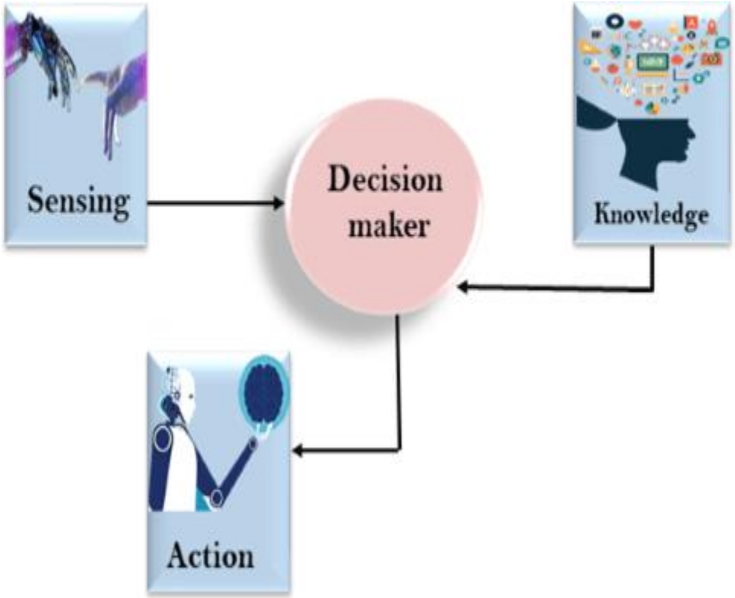
- Definition and Importance of Knowledge
- Knowledge-Based Systems
- Knowledge Organization
- Representation of Knowledge
  - Logic
  - Associative Networks
  - Frame Structures
  - Conceptual graphs

What is Knowledge ?

- Knowledge is the sort of information that people use to solve problems.
- **Knowledge is having familiarity with the language, concepts, procedures, rules, ideas, places, customs, facts, and associations.**

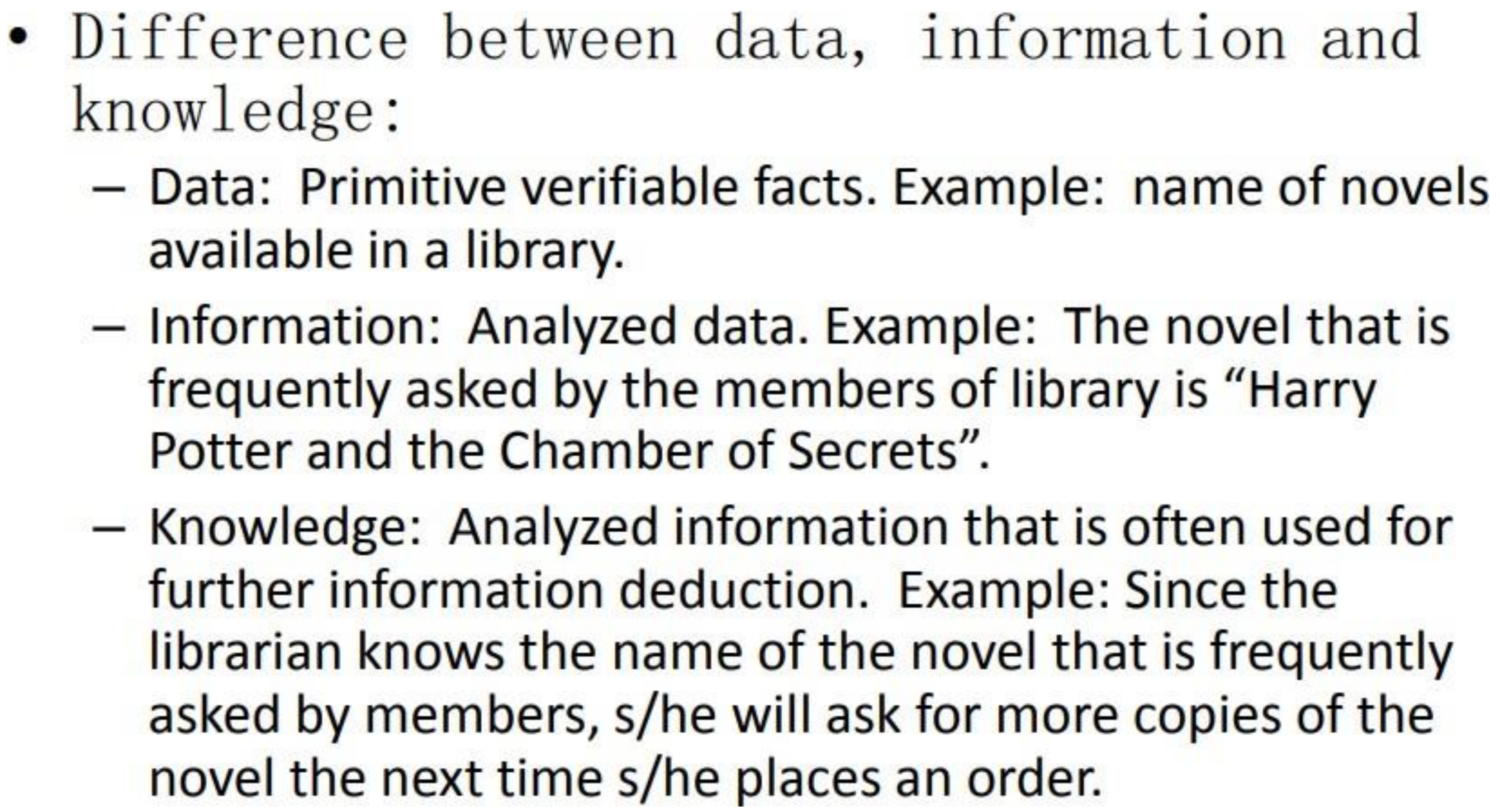
- Knowledge is understanding of a subject area.


Concepts and facts
Relationships between concepts
Knowledge
Properties of concepts
Mechanisms for how to combine concepts to solve problems



Ref : https://nptel.ac.in/courses/126104006/LectureNotes/Week-3_Knowledge%20Representation.pdf
https://www.javatpoint.com/knowledge-representation-in-ai

What is Knowledge?



- Difference between data, information and knowledge:
  - Data: Primitive verifiable facts. Example: name of novels available in a library.
  - Information: Analyzed data. Example: The novel that is frequently asked by the members of library is “Harry Potter and the Chamber of Secrets”.
  - Knowledge: Analyzed information that is often used for further information deduction. Example: Since the librarian knows the name of the novel that is frequently asked by members, s/he will ask for more copies of the novel the next time s/he places an order.

Ref : https://nptel.ac.in/courses/126104006/LectureNotes/Week-3_Knowledge%20Representation.pdf

Importance of Knowledge

- Knowledge is Power
- Knowledge is the primary factor that distinguishes the human race from animals
- knowledge is even what prevents us from making the same mistakes we made in the past.
- With knowledge, one can improve their abilities of thinking critically
- To make program intelligent provide it with lots of high quality specific knowledge about some problem area.
- AI systems are divided into a Knowledge Base with facts about the world and rules and an inference engine that applies the rules to the knowledge base in order to answer questions and solve problems.

Importance of Knowledge

- Common way to represent knowledge is in the form of **written language**.
- For example, some facts and relations represented in English are:

1. **Joe is tall.**
2. **Bill likes Sue.**
3. **Sam has learned to use recursion to manipulate linked list in several programming languages.**

Importance of Knowledge

Joe is tall.

- knowledge above expresses simple fact.
- an attribute possessed by a person.

Bill likes Sue.

- knowledge above expresses complex binary relation between two persons.

Sam has learnt to use recursion to manipulate linked list in several programming languages.

- knowledge above is the most complex.
- Expressing relations between a person and more abstract programming concepts.

Important points on Knowledge

Knowledge may be declarative or procedural

- **Procedural Knowledge**

  - It is a compiled knowledge related to the performance of some task.

  For example, the steps used to solve an algebraic equations are expressed as procedural knowledge.

- **Declarative Knowledge**

  - Passive knowledge expressed as statements of facts about the world.

  For example, personnel data in a database.

Important points on Knowledge

Knowledge should not be confused with data

Example:

A physician treating a patient uses both knowledge and data.

Data: Patient's record, including patient history, drugs given, responses to drugs, and so on.

Knowledge: What the physician has learnt in medical school and in the years of internship, specialization and practice.

It consists of facts, prejudices and beliefs.

Important points on Knowledge

- Knowledge includes and requires the use of data and information.

- It combines relationships, correlations and dependencies.

Important points on Knowledge

Distinction between Knowledge and other concepts such as belief and hypotheses

Belief: Any meaningful and coherent expression that can be represented. It may be true or false.

E.g. It is not good to have lunch after 4.30.

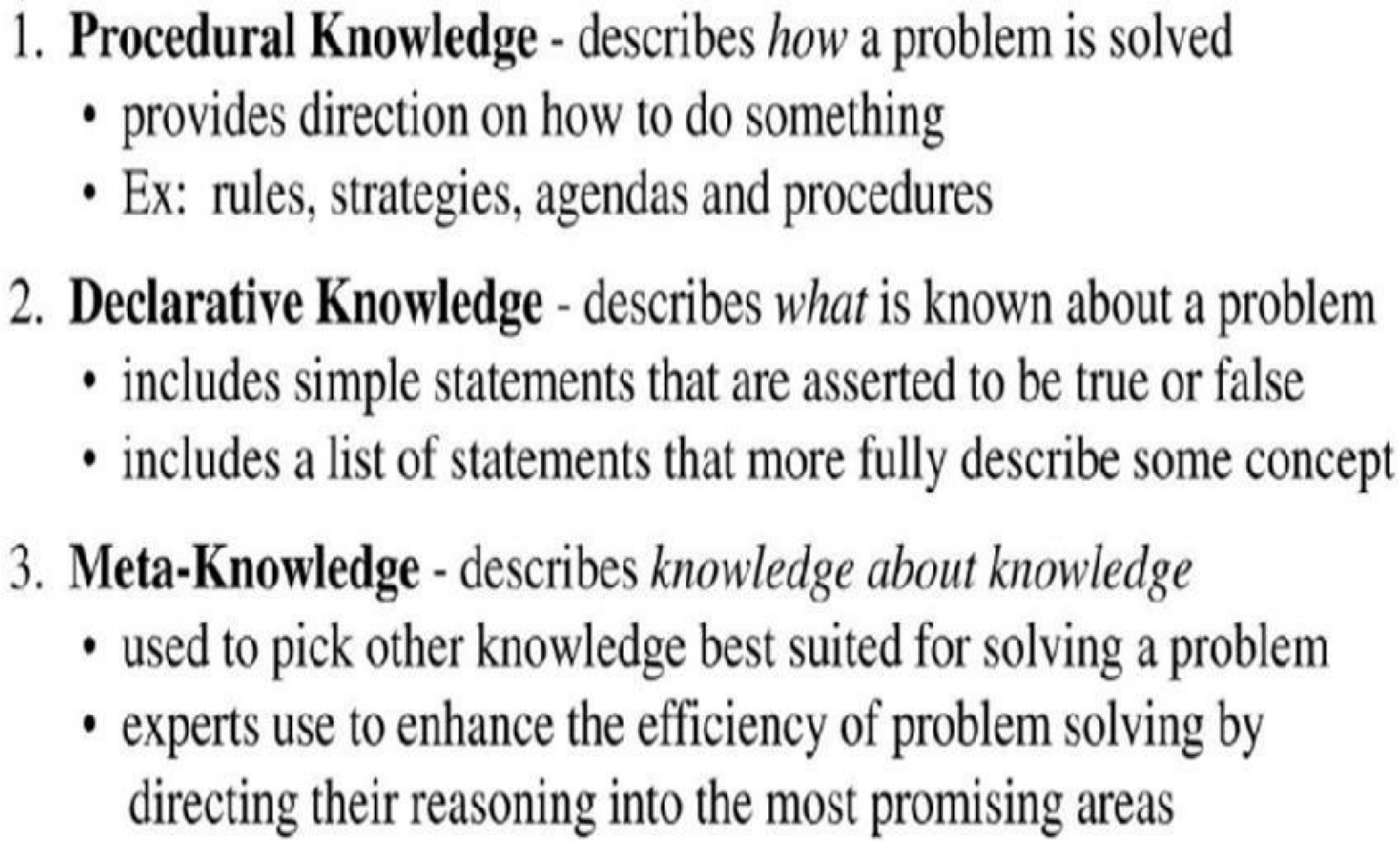
Hypotheses: A Justified belief that is not known to be true.

E.g. Tomorrow guests are coming; so prepare Fried Rice for lunch. They may like it.

Knowledge: True justified belief.

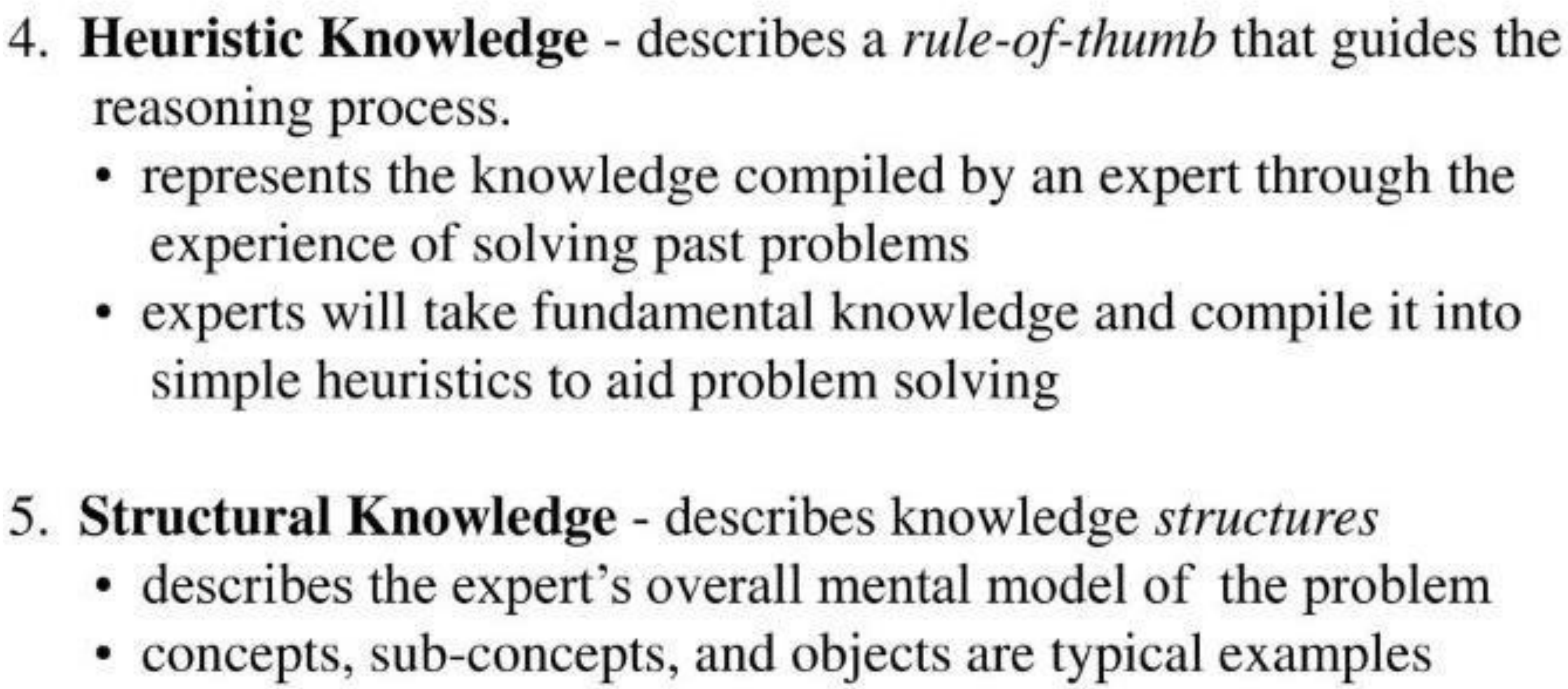
E.g. If you want to get better marks, you need to practice more. By practicing more, you can remember the concepts.

Types Of Knowledge



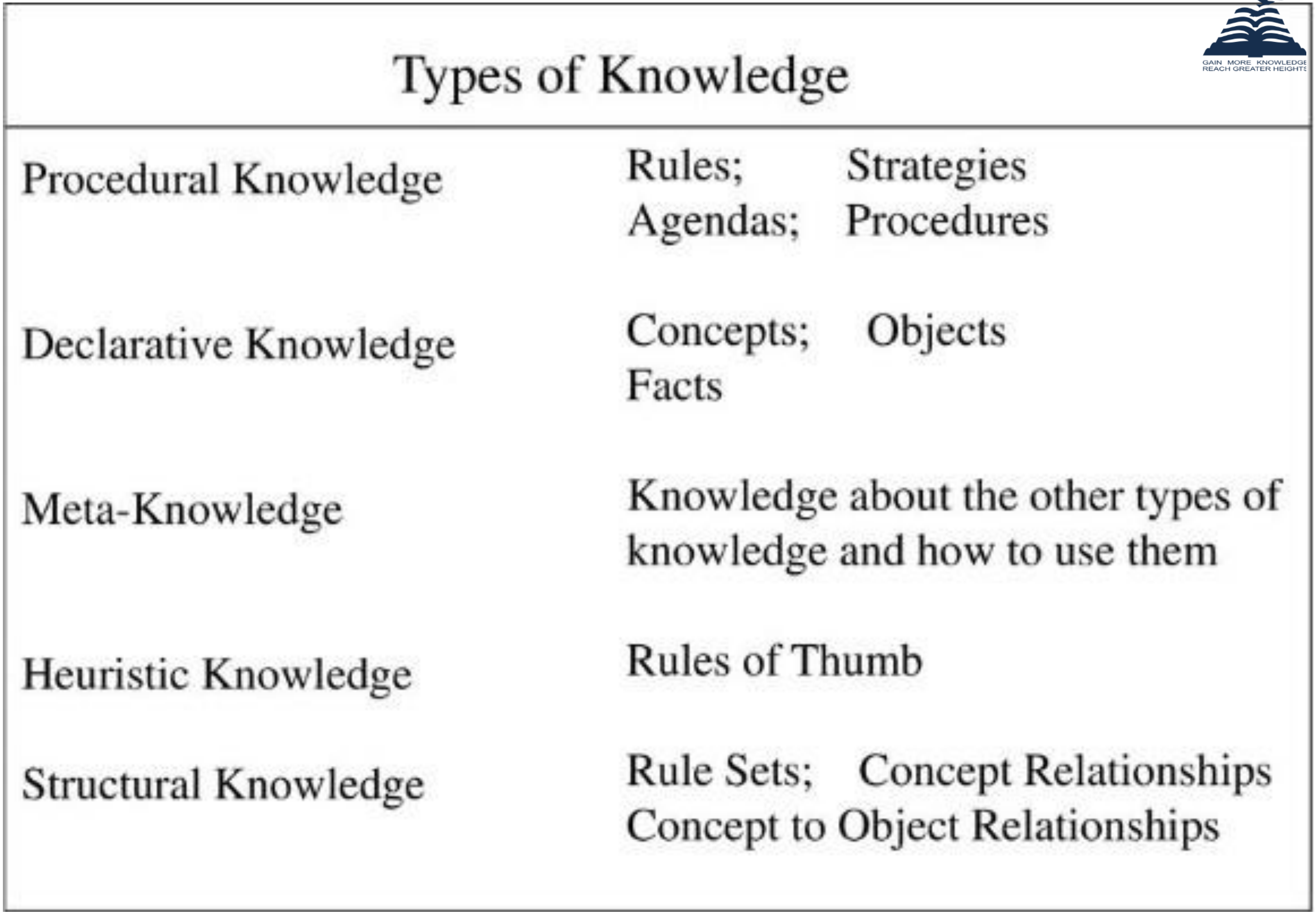
1. **Procedural Knowledge** - describes *how* a problem is solved
   - provides direction on how to do something
   - Ex: rules, strategies, agendas and procedures
2. **Declarative Knowledge** - describes *what* is known about a problem
   - includes simple statements that are asserted to be true or false
   - includes a list of statements that more fully describe some concept
3. **Meta-Knowledge** - describes *knowledge about knowledge*
   - used to pick other knowledge best suited for solving a problem
   - experts use to enhance the efficiency of problem solving by directing their reasoning into the most promising areas

Types Of Knowledge



4. **Heuristic Knowledge** - describes a *rule-of-thumb* that guides the reasoning process.
    - represents the knowledge compiled by an expert through the experience of solving past problems
    - experts will take fundamental knowledge and compile it into simple heuristics to aid problem solving

5. **Structural Knowledge** - describes knowledge *structures*
    - describes the expert's overall mental model of the problem
    - concepts, sub-concepts, and objects are typical examples



Types of Knowledge

Procedural Knowledge	Rules; Strategies Agendas; Procedures
Declarative Knowledge	Concepts; Objects Facts
Meta-Knowledge	Knowledge about the other types of knowledge and how to use them
Heuristic Knowledge	Rules of Thumb
Structural Knowledge	Rule Sets; Concept Relationships Concept to Object Relationships

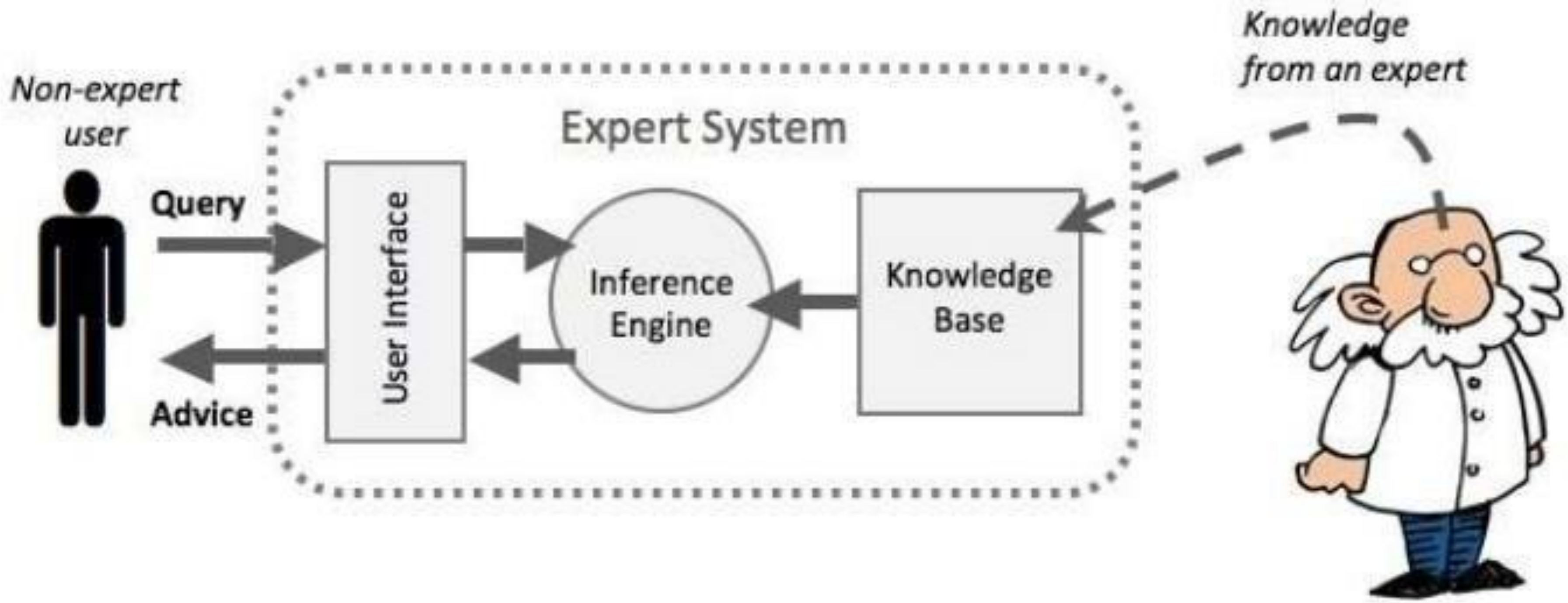
Knowledge Based Systems (KBS)

- A knowledge-based system (KBS) is a computer program that reasons and uses a knowledge base to solve complex problems
- A system which is built around a knowledge base. i.e. a collection of knowledge, taken from a human, and stored in such a way that the system can reason with it
- Uses AI to solve problems within a specialized domain that ordinarily requires human expertise
- Uses Heuristic (cause and effect) rather than algorithms
- E.g.
  - Expert Systems
  - Clinical decision-support systems
- MYCIN, for example, was an early knowledge-based system created to help doctors diagnose diseases

KBS Examples

- **Expert Systems**
  - One in which the knowledge, stored in the knowledge base, has been taken from an expert in some particular field
  - **Expert systems** are designed to solve complex problems by reasoning through bodies of knowledge, represented mainly as if–then rules rather than through conventional procedural code
  - Therefore, an expert system can, to a certain extent, act as a substitute for the expert from whom the knowledge was taken
- **Clinical decision-support systems**
  - MYCIN, for example, was an early knowledge-based system created to help doctors diagnose diseases

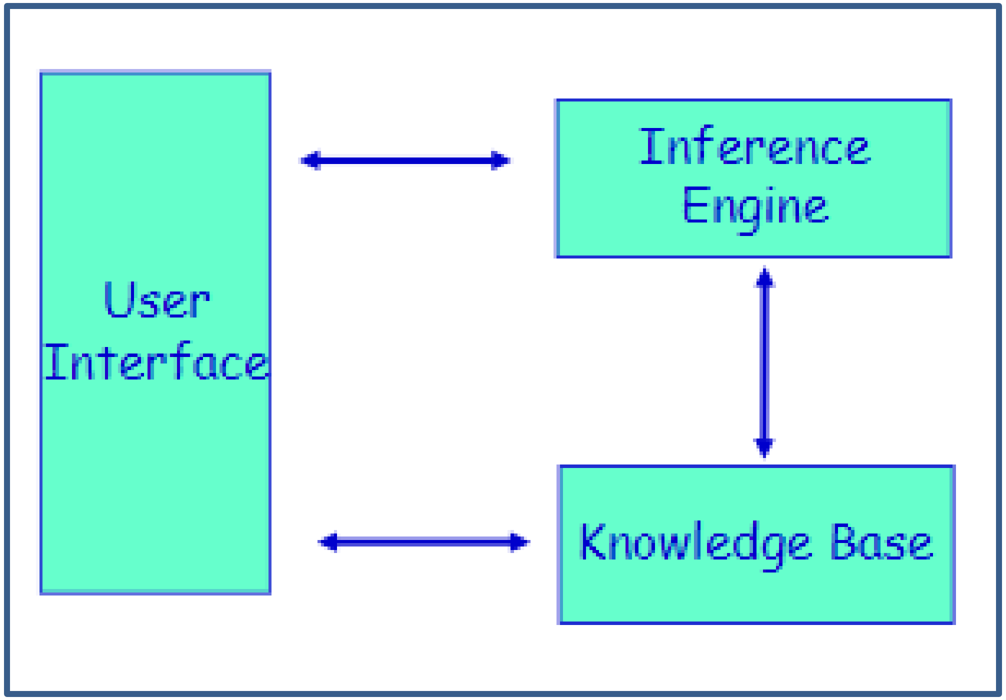
KBS Architecture



https://www.researchgate.net/figure/Expert-System-Architecture-23-Knowledge-Base_fig1_319208444

KBS Architecture

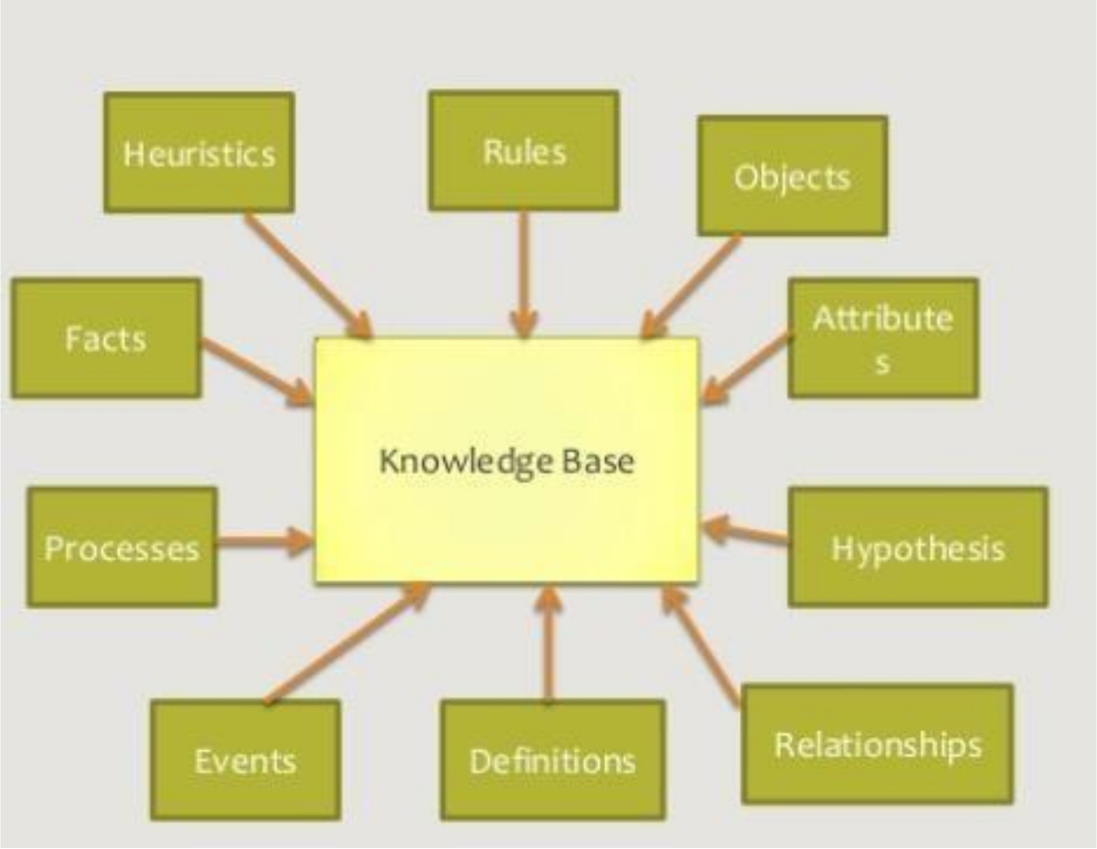
- **User Interface**
  - Enables the user to communicate with KBS



KBS Architecture

KBS = Knowledge-Base + Inference Engine

- **Knowledge Base**
  - The component of KBS that contains the system’s knowledge organized in collection of facts about the system’s domain



Knowledge base System

Storing knowledge inside the program

```
#include<stdio.h>
int main()
{
   char dob="20/08/1992";                         #Knowledge
   .............
   .............
}
```

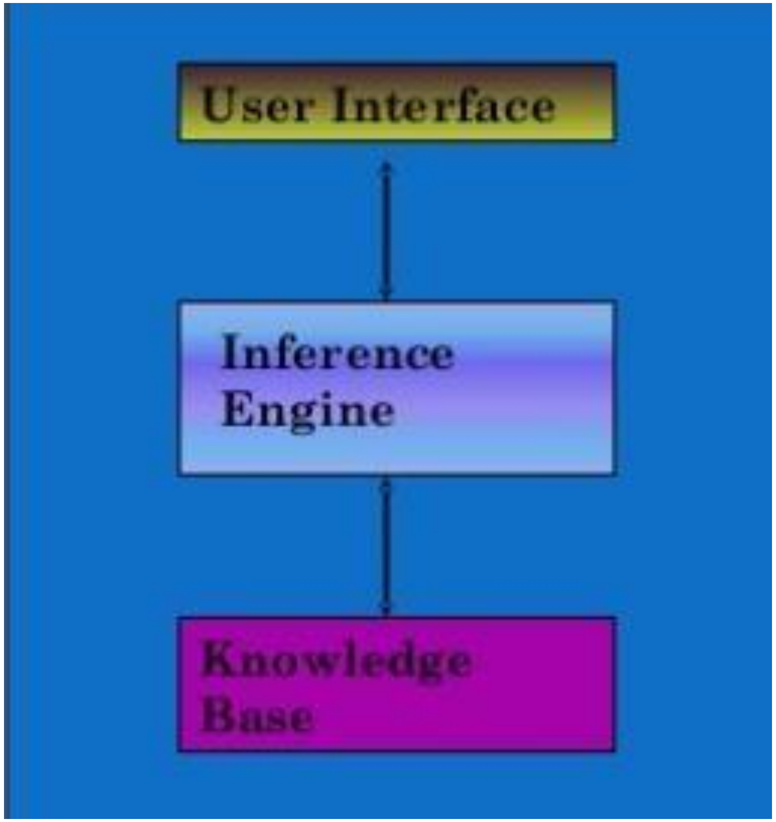
Instead write the dob in text file and access the date of birth from the text file

Text file: dob.txt

20/08/1992

KBS Architecture

- **Inference Engine :**
  - Tries to derive answers from knowledge base
  - Brain of KBS that provides a methodology for reasoning about the information in the knowledge base and for formulating conclusions



Example 1 for AI system: Gender Identification Problem

- Male and female names have some distinctive characteristics.
- Names ending in *a*, *e* and *i* are likely to be female.
- Names ending in *k*, *n*, *r*, *s* and *t* are likely to be male.

Input File: male.txt
Amit
Prasan
Ashok
Ankit
Amar
Chetan
Shashank
Sumant

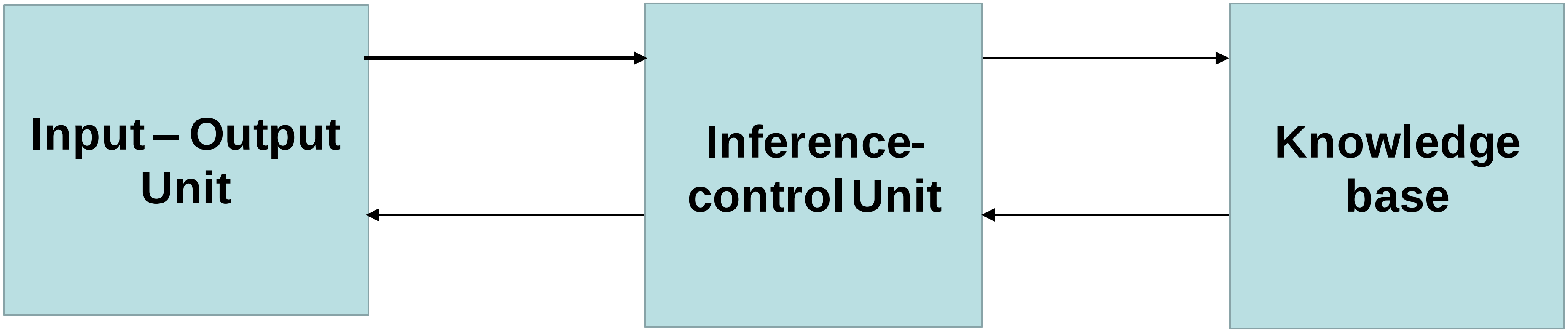
Input File: female.txt
Reshma
Akshata
Vani Sita
Bhavani
Lalita
Ankita
Harika

Output: Predict the Gender for the following names:

- Karan
- Sameera

Architecture of AI

Components of Knowledge base System



Input-output: male.txt and female.txt

Knowledge base:

From the last character in the name identification of gender is possible.

Inference-control Unit: AI Algorithm Implementation – Programs

Example: Naïve Bayes Algorithm, Decision Tree Algorithm

List of Common Algorithms:

- Naive Bayes
- Decision Trees
- Linear Regression
- Support Vector Machines (SVM)
- Neural Networks

Example 2 for AI system: Movie Rating

Airlift Movie rating – Reviews (**Movie site**, **Facebook**, **Blog**)

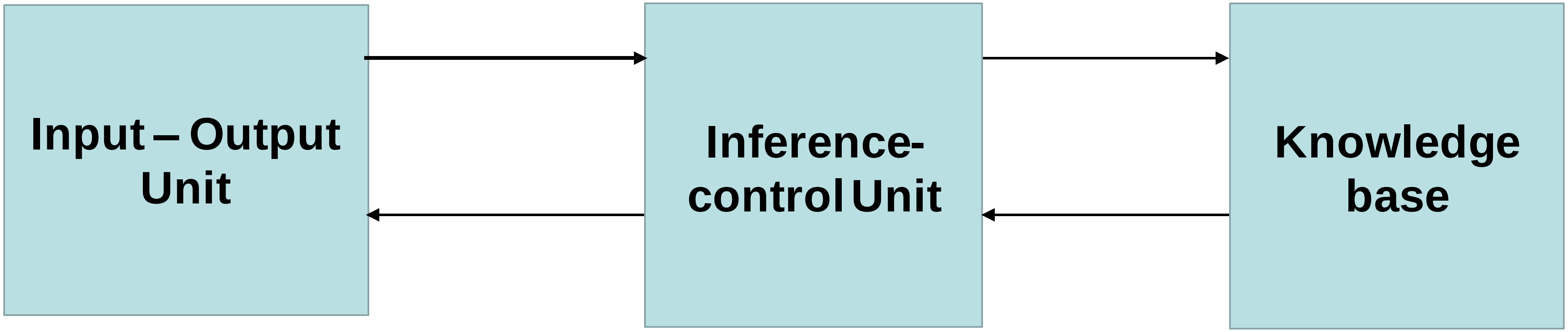
Post - How is Airlift Movie?

Comments from the people who watched movie -

- Airlift Movie is nice.
- It's boring.
- Yesterday I went to the movie. I enjoyed it.
- Superb.

Architecture of AI

Components of Knowledge base System



Input-output: Facebook, twitter and movie site comments about the movie.

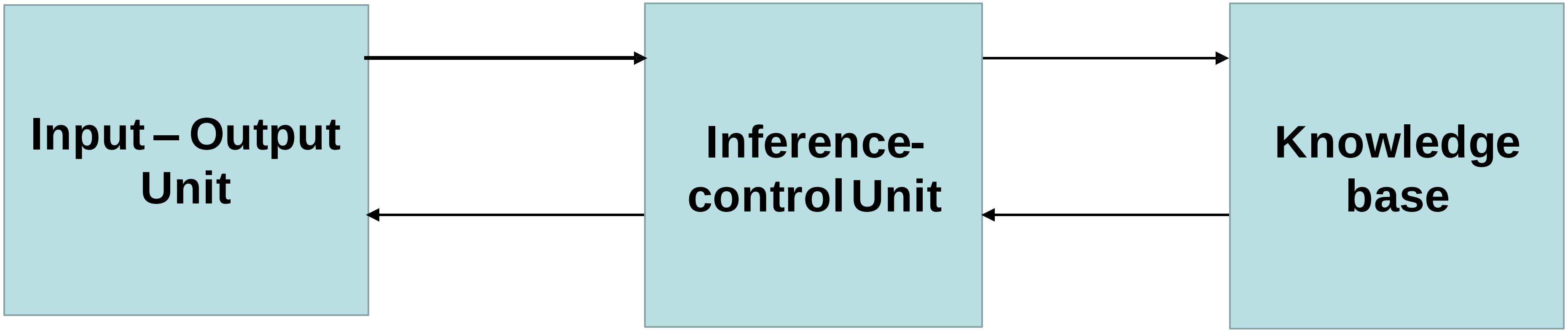
Knowledge base: Contains the **positive**, **negative** and **neutral** keywords (Dictionary).

Inference-control Unit: AI Algorithm Implementation – Programs

Example: NLP Algorithms, Naïve Bayes Algorithm, Support Vector Machines

Architecture of AI

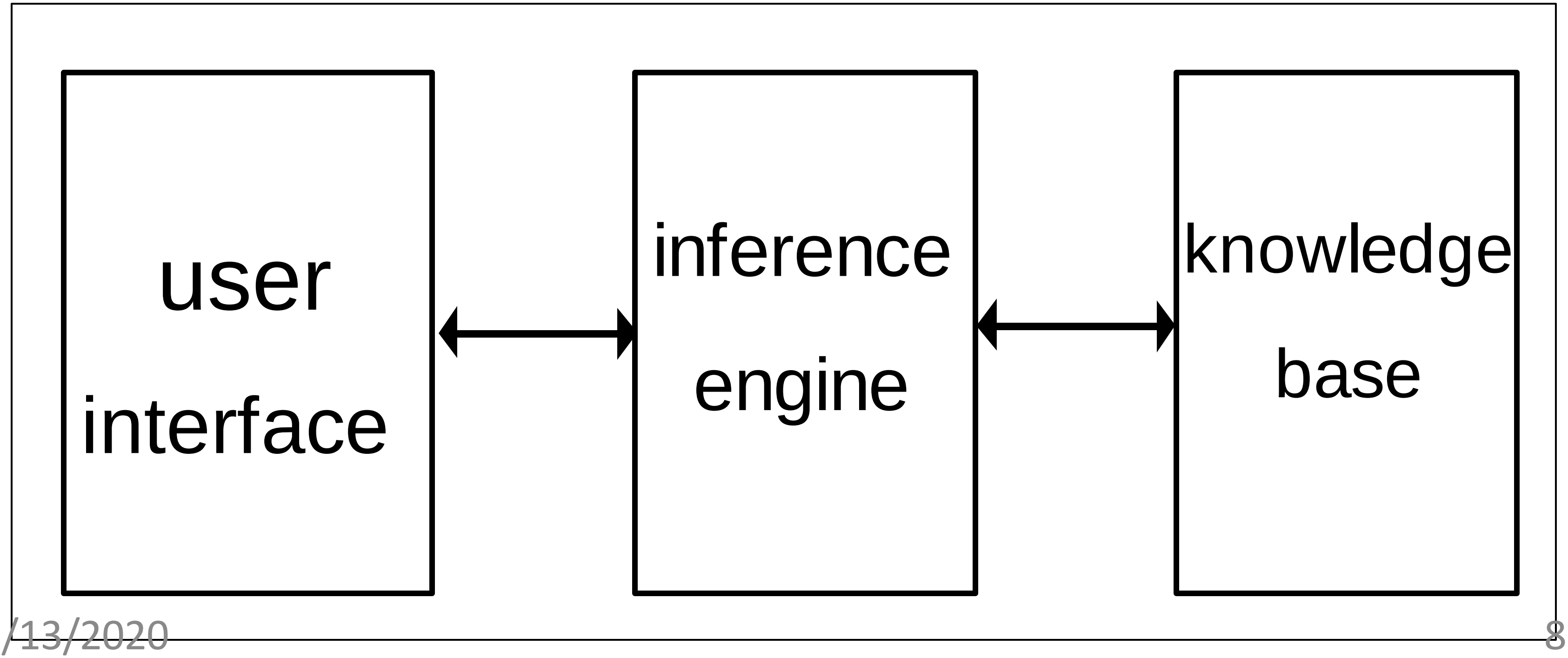
Components of Knowledge base System



- Knowledge-based systems – get the power from expert knowledge that has been coded into facts, heuristics, and procedures.
- The knowledge is stored in knowledge base separate from the control and inference components.
- This makes possible to add new knowledge or refine existing knowledge without recompiling the control and inference programs.

KBS architecture (1)

- The typical architecture of an KBS is often described as follows:

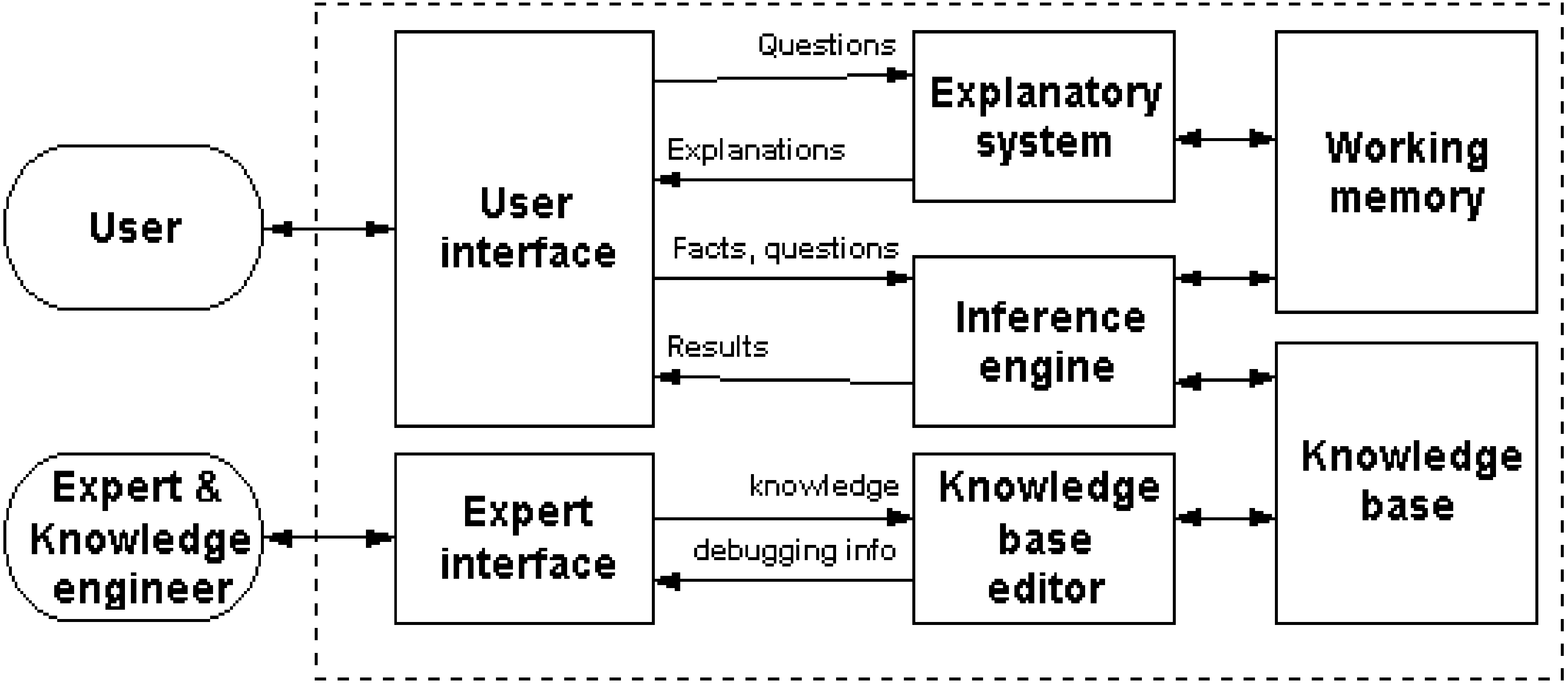


KBS architecture (2)

- However, it is reasonable to produce **a richer, more elaborate, description** of the typical KBS.

- A more **elaborate description**, which still includes the components that are to be found in almost any real-world system, would look like this:

KBS architecture (2)



KBS architecture (2)

- The system holds a collection of ***general principles*** which can potentially be applied to any problem - these are stored in the ***knowledge base***.

- The system also holds a collection of ***specific details*** that apply to the **current problem** (including **details of how the current reasoning process is progressing**) - these are held in ***working memory***.

- Both these sorts of information are processed by the ***inference engine***.

KBS architecture (2)

- **Expert System:** This artificial intelligence system is a database which attempts to reason like a person by using **logic** to extract new information from a set of information.

- Any practical expert system needs an **explanatory facility**. It is essential that an expert system should be able to explain its **reasoning**. This is because:

  it gives the user **confidence** in the system;

  it makes it easier to **debug** the system.

KBS architecture (2)

- It is not unreasonable to include an **expert interface & a knowledge base editor**, since any practical KBS is going to need a **mechanism for efficiently building and modifying the knowledge base**.

Knowledge Organization

- Knowledge organization involves activities that "*classify, map, index, and categorize knowledge for navigation, storage, and retrieval*"

Knowledge Organization

- Organization of knowledge in memory is key to **efficient processing**.
- Knowledge-based systems may require tens of thousands of **facts and rules** to perform their intended tasks.
- It is essential that appropriate facts and rules be easy to **locate and retrieve**.
- Otherwise, much **time will be wasted** in **searching and testing** large number of items in the memory.

Knowledge Organization

- Knowledge can be organized in memory for easy access by a method known as **indexing**.
- It amounts to **grouping the knowledge** in a way that keywords can be used to access the group.

Example:

Positive:

Nice

Beautiful

Superb

Negative:

Bad Doubt

Guilt

Dataset

A collection of related sets of information that is composed of separate elements but can be manipulated as a unit by a computer.

Input File: female.txt

```
Reshma
Akshata
Vani  Sita
Bhavani
Lalita
Ankita
Harika
```

Dataset

- Size of the dataset should be as large as possible, which increases the prediction accuracy.

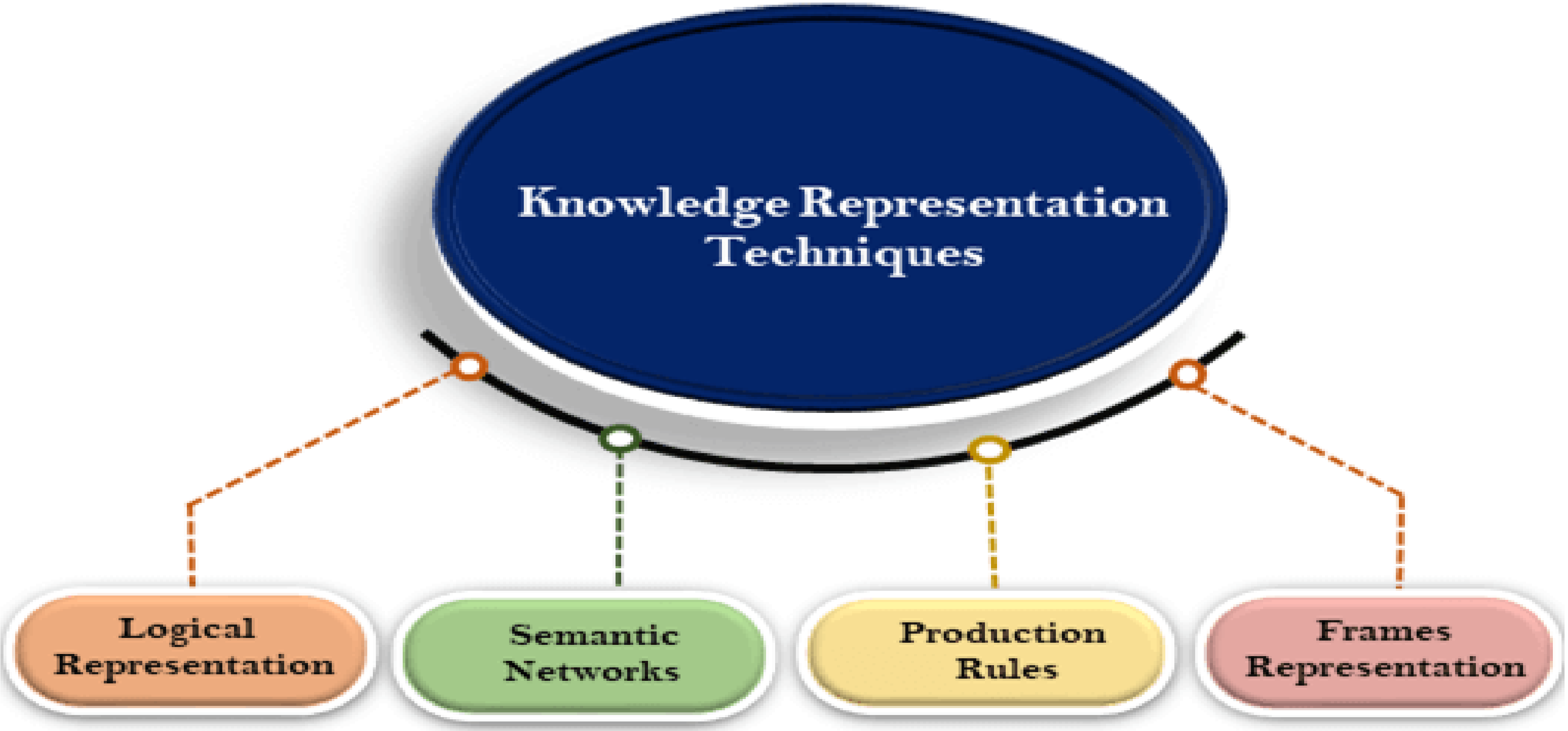
- Dataset should not contain duplicate elements.

Knowledge Manipulation

- Knowledge manipulation involves the following activities:
  - **Knowledge acquisition**
    - It is the process of gathering, structuring and organizing knowledge of a particular topic or a domain in order to prepare it to be put into the computer memory.
  - **Knowledge Storing**
    - The process of putting the knowledge into the computer in a suitably encoded format.
  - **Knowledge Retrieval**
    - The inverse process of getting the knowledge back whenever it is needed.
  - **Reasoning**
    - This is the most important part of knowledge representation.
    - It includes deriving new knowledge from the existing knowledge by means of an intelligent program.
    - The newly derived knowledge is known as conclusion, inference or explanation.

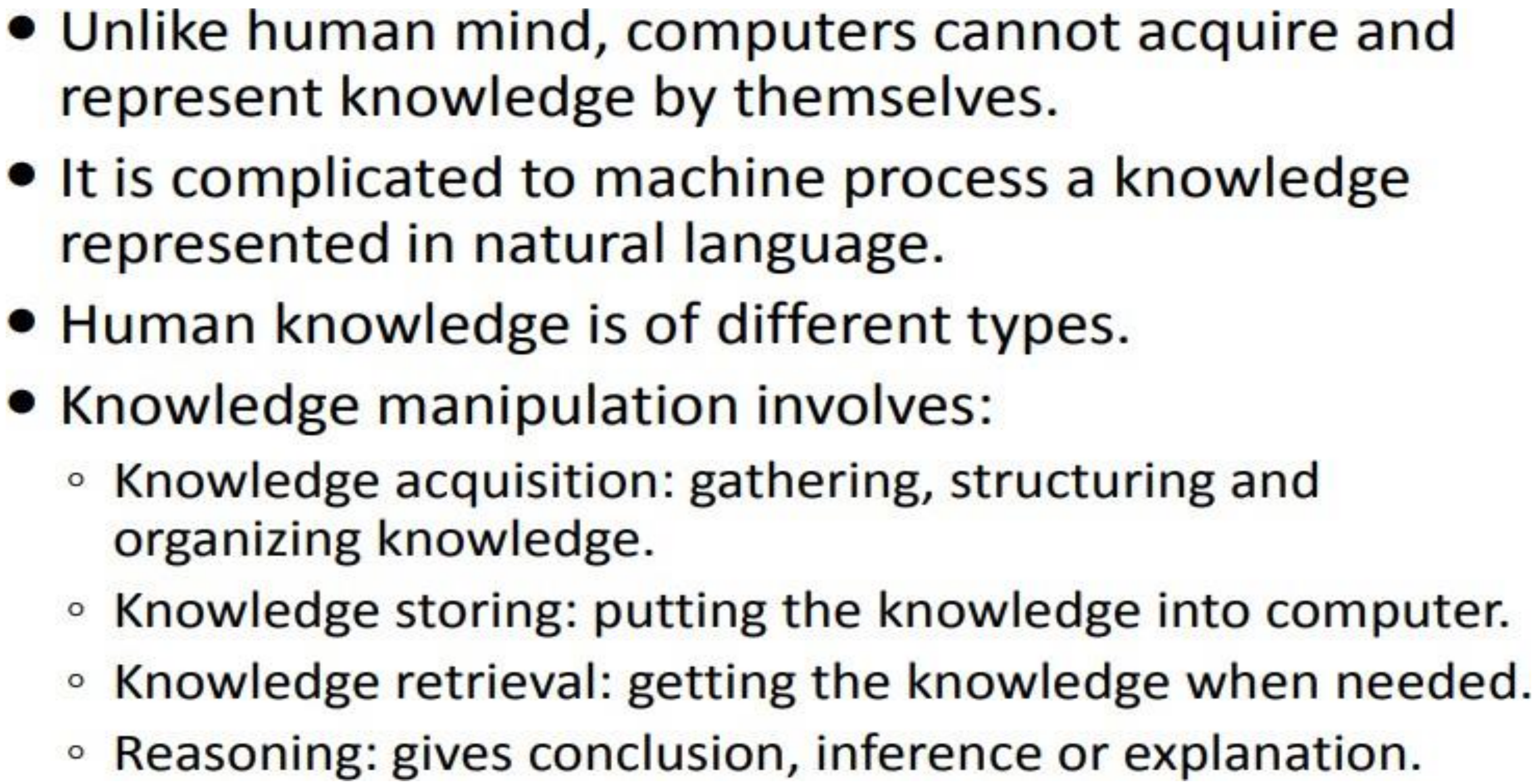
Knowledge Representation (KR)

- The method used to encode knowledge in an KBS's Knowledge base
- The field of AI dedicated to representing information about the world in a form that a computer system can utilize to solve complex tasks



https://www.javatpoint.com/ai-techniques-of-knowledge-representation

Why do we need Knowledge Representation?



- Unlike human mind, computers cannot acquire and represent knowledge by themselves.
- It is complicated to machine process a knowledge represented in natural language.
- Human knowledge is of different types.
- Knowledge manipulation involves:
  - Knowledge acquisition: gathering, structuring and organizing knowledge.
  - Knowledge storing: putting the knowledge into computer.
  - Knowledge retrieval: getting the knowledge when needed.
  - Reasoning: gives conclusion, inference or explanation.

Ref : https://nptel.ac.in/courses/126104006/LectureNotes/Week-3_Knowledge%20Representation.pdf

Knowledge Representation Schemas

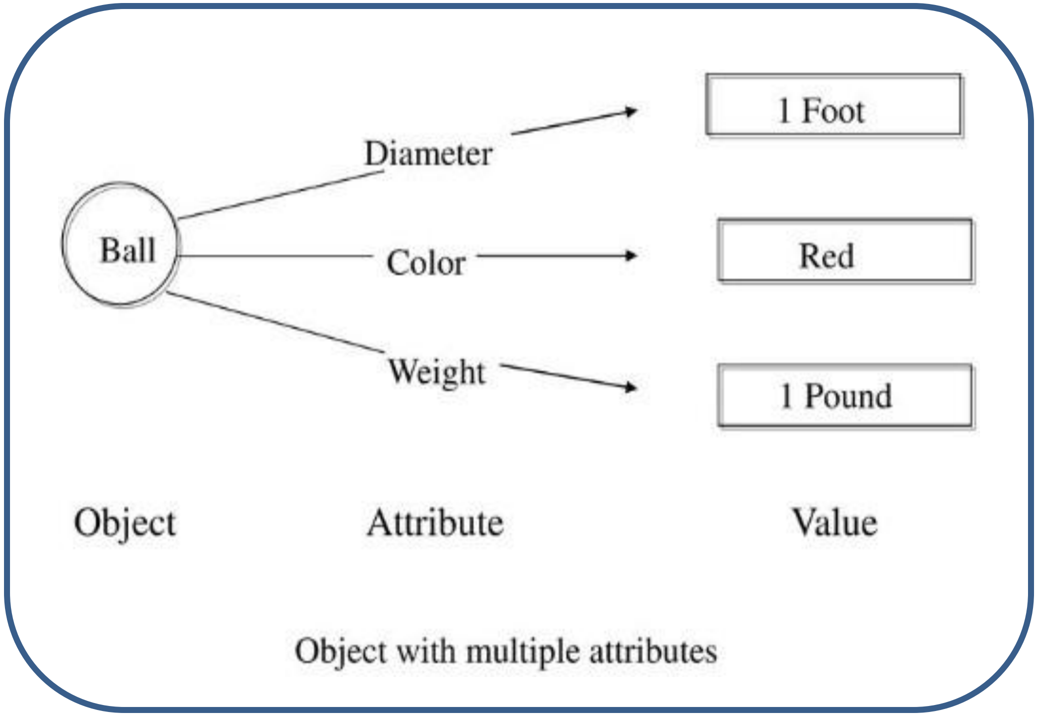
• **Logical schemas** - First-order logic - Higher-order logic	• **Network schemas** - Semantic networks - Conceptual graphs
• **Procedural schemas** - Rule-based systems	• **Structural schemas** - Frames - Scripts

Common Techniques of KR

- Object – Attribute – Value Triplets (O-A-V)
- Rules
- Semantic Networks
- Frames
- Logic

O-A-V

- *The Ball is round in shape*
  - **Object** : Ball
  - **Attribute** : Shape
  - **Value** : Red



Object with multiple attributes

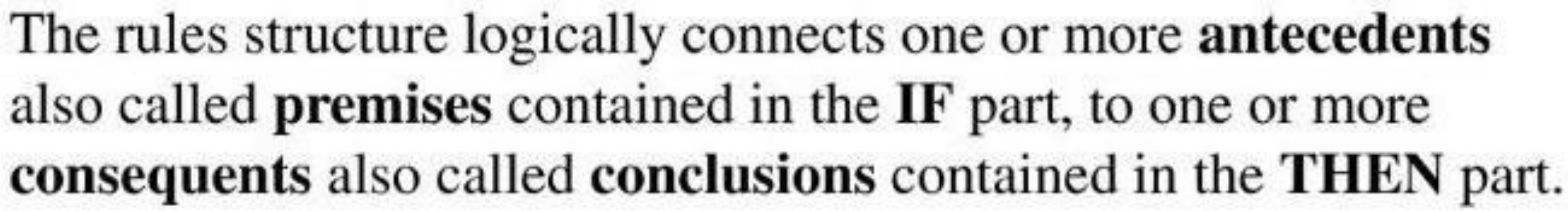
Rules

- **Rule**: A knowledge structure that relates some known information to other information that can be concluded or inferred

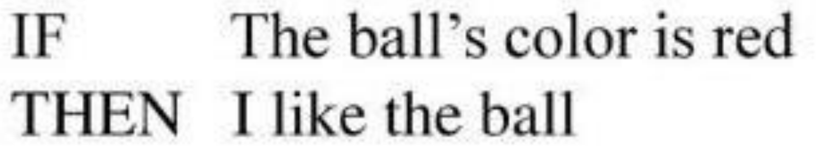
- A rule describes how to solve a problem

- Expert systems employing rules are called rule-based expert systems

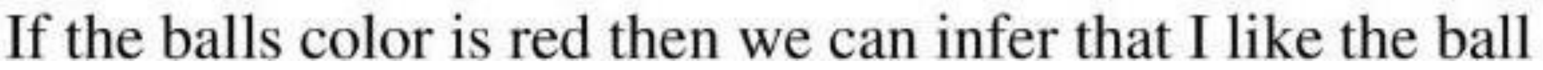
Structure of Rule



The rules structure logically connects one or more **antecedents** also called **premises** contained in the **IF** part, to one or more **consequents** also called **conclusions** contained in the **THEN** part.

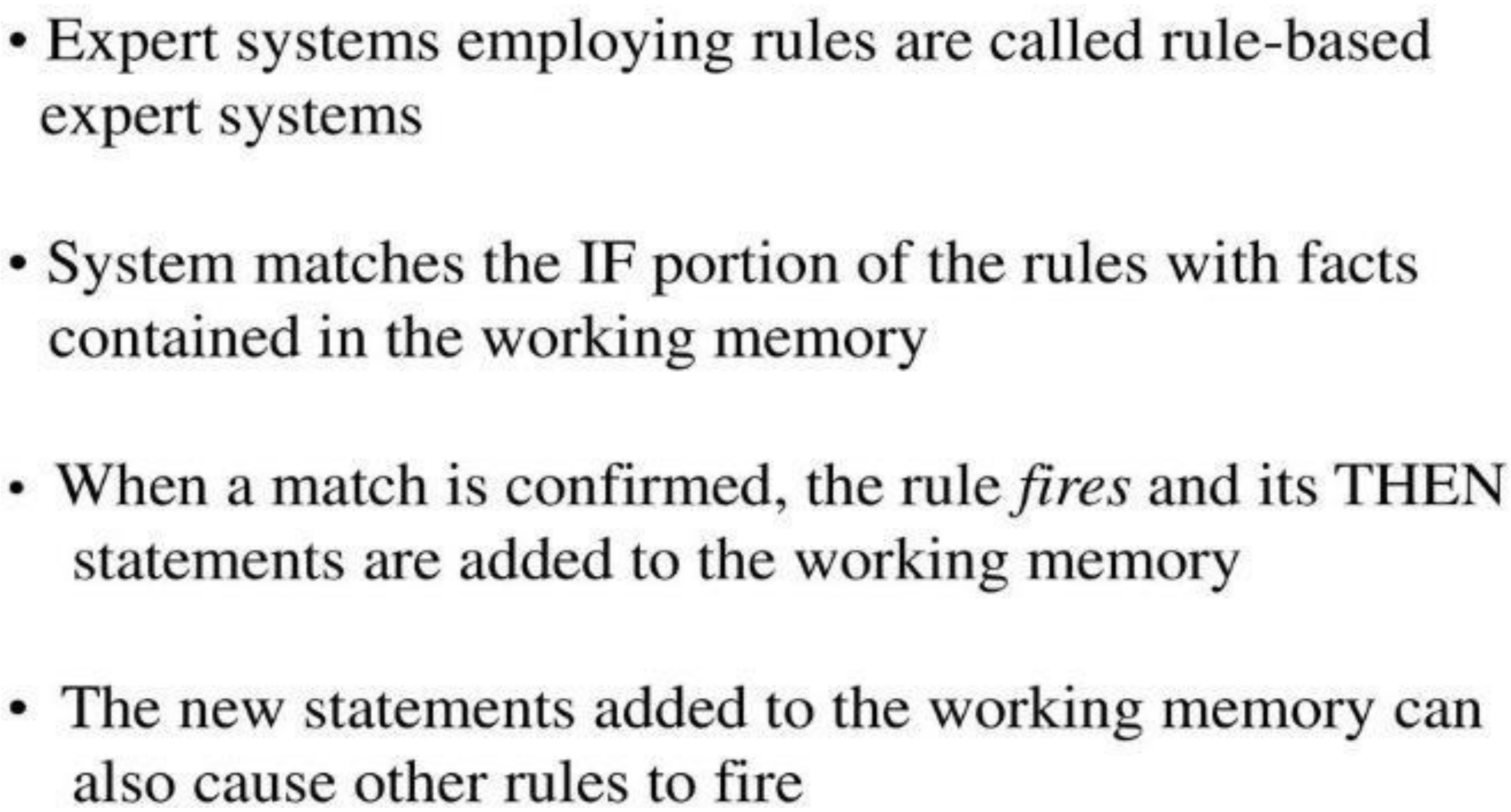


IF The ball’s color is red
THEN I like the ball



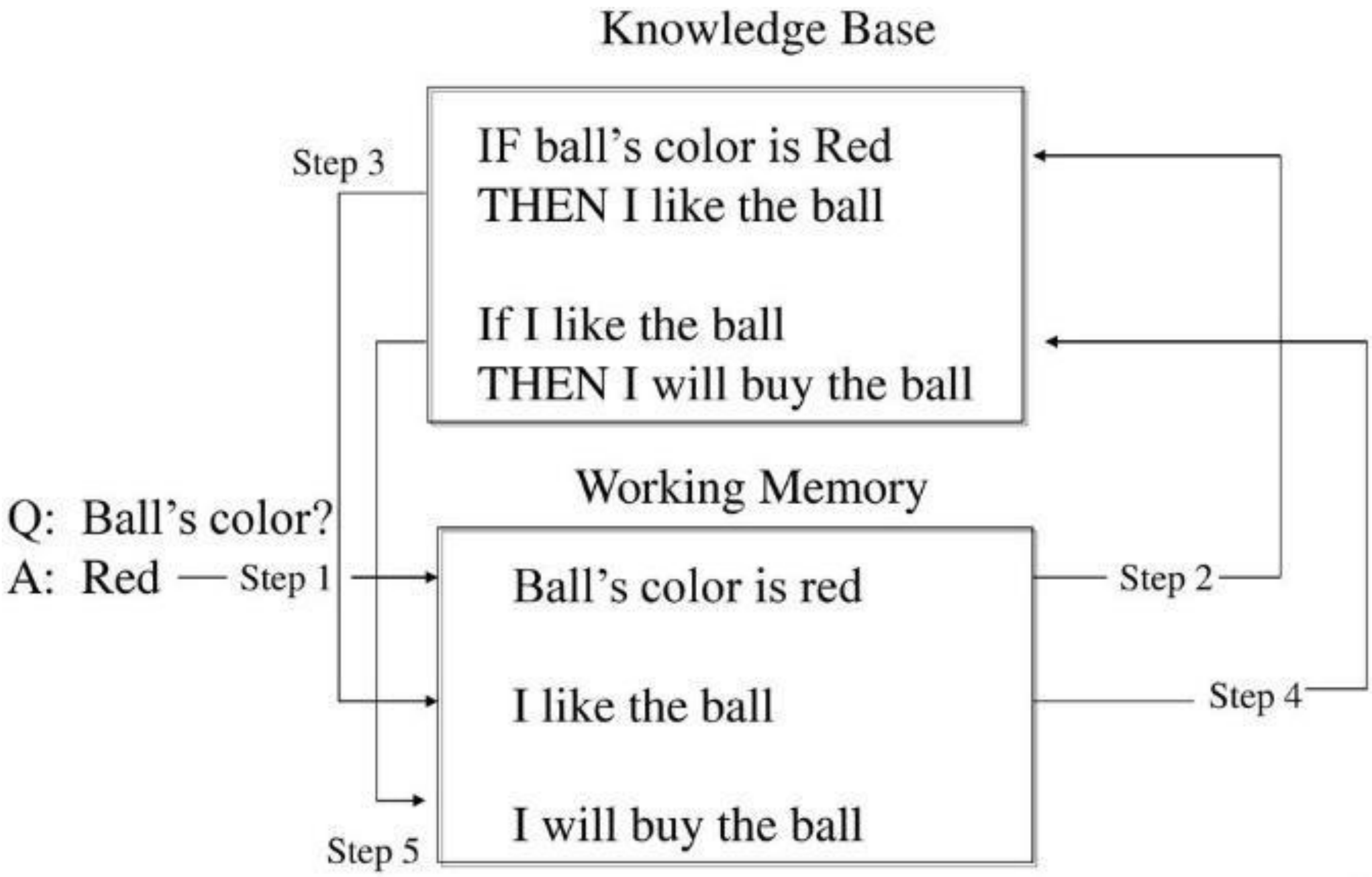
If the balls color is red then we can infer that I like the ball

Working Memory



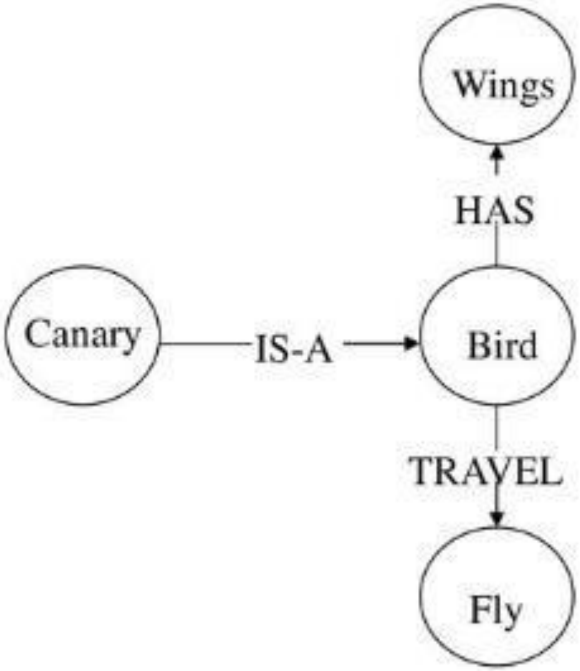
- Expert systems employing rules are called rule-based expert systems
- System matches the IF portion of the rules with facts contained in the working memory
- When a match is confirmed, the rule *fires* and its THEN statements are added to the working memory
- The new statements added to the working memory can also cause other rules to fire

Inference Engine



Semantic Networks

DEFINITION: Semantic Network - A method of knowledge representation using a graph made up of nodes and arcs where the nodes represent objects and the arcs the relationships between the objects



IS-A (Canary, Bird)

Semantic network of a bird

Associative Networks

- Semantic networks consist of **nodes, links (edges)** and **link labels**.
- **nodes** appear as circles or ellipses or rectangles to represent objects such as physical objects, concepts or situations.
- **Links** appear as arrows to express the relationships between objects .
- **link labels** specify particular relations .
- As nodes are associated with other nodes semantic nets are also referred to as **Associative Networks.**
- Semantic Networks, Frames and Scripts are sometimes called as **Associative Networks.**

Associative Networks

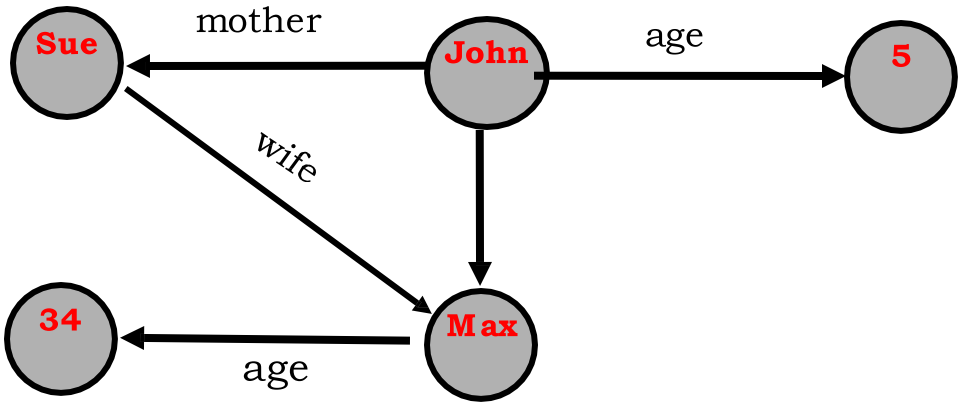
Example :

mother(john,sue)

age(john,5)

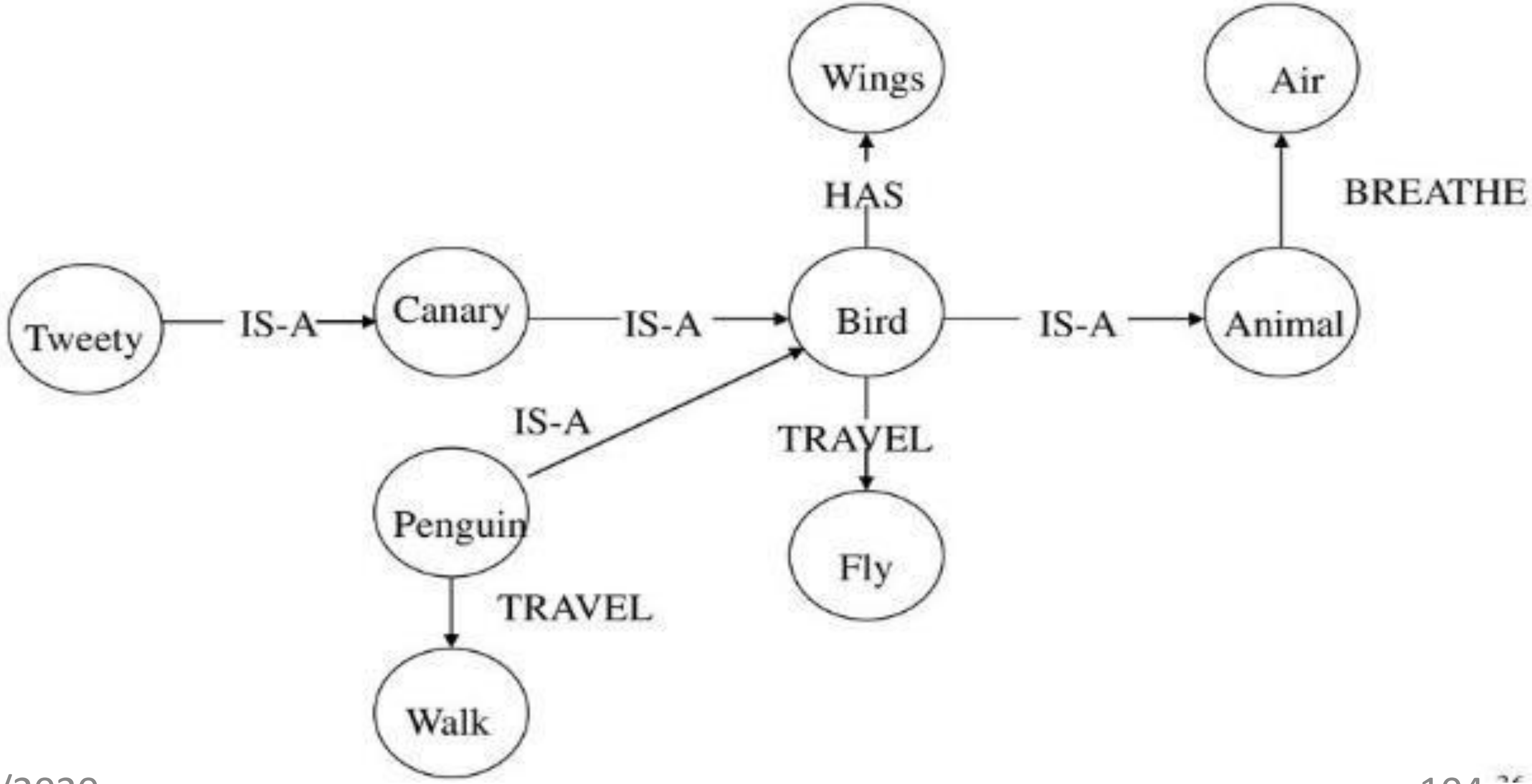
wife(sue,max)

age(max,34)



Semantic Networks

You can add a new object node by 1) a similar object, 2) a more specific object, or (3) a more general object.



Basics of Associative Networks

- It's defined as various kinds of links between the **concepts**.

- “has-part” or aggregation.

- “is-a” or specialization.

- More specialized depending on domain.

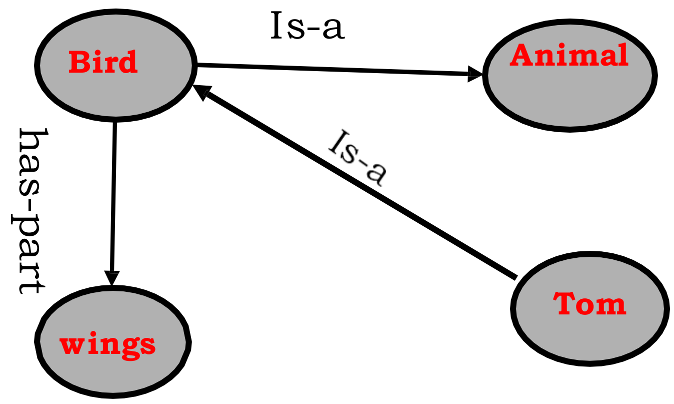
- It typically also includes **Inheritance** and some kind of procedural attachment.

Example :

- Tom is a Bird.
- Bird is a Animal.
- Bird has part Wings.

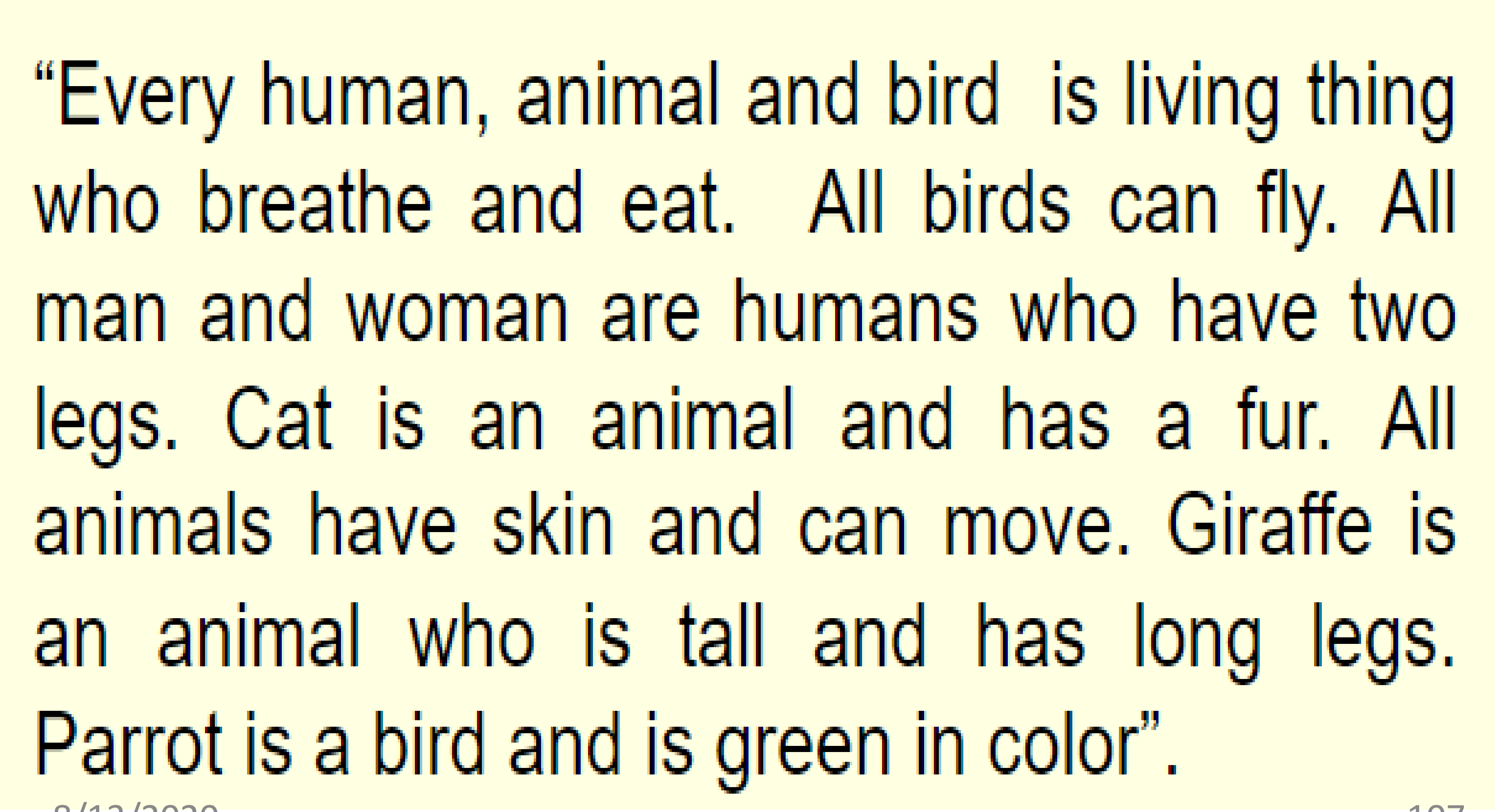
Basics of Associative Networks

Example : Tom is a Bird.
Bird is a Animal.
Bird has part Wings.

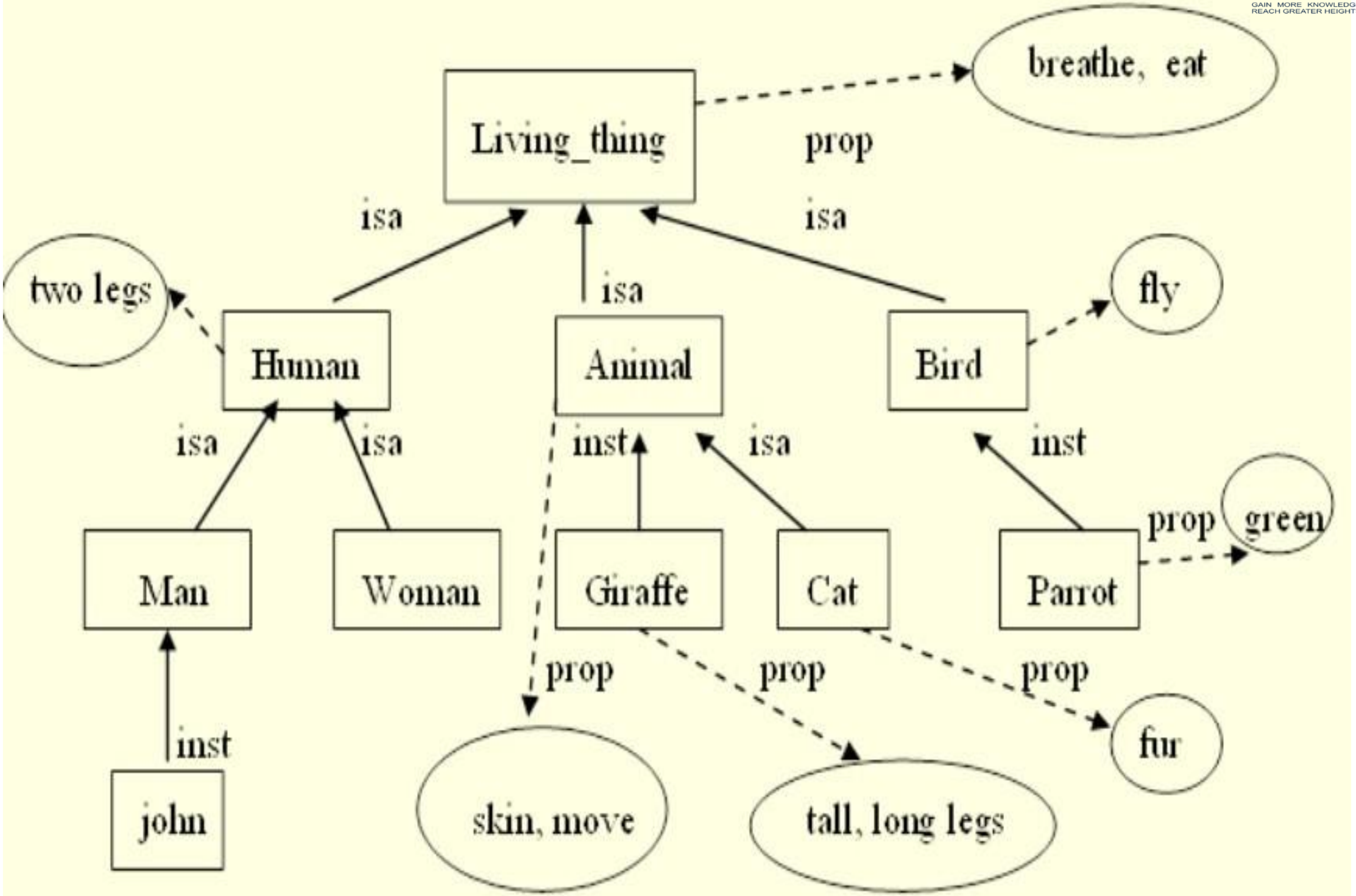


- The ISA (**is-a**) or AKO (a-kind-of) relation is often used to link instances to classes, classes to super classes
- Some links (e.g. **has Part**) are inherited along ISA paths.
- The semantics of a semantic net can be relatively informal or very formal
  - often defined at the implementation level

Draw a Semantic Network



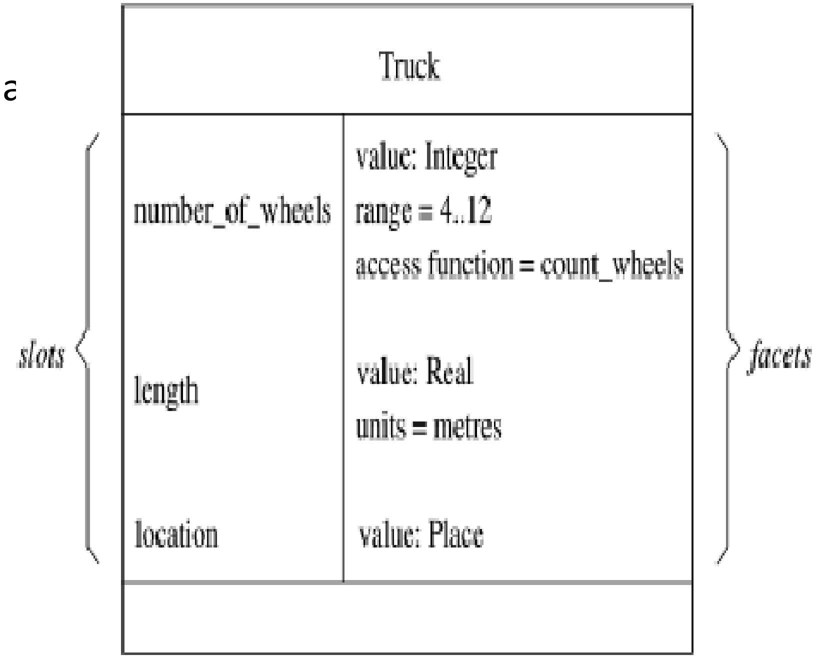
“Every human, animal and bird is living thing who breathe and eat. All birds can fly. All man and woman are humans who have two legs. Cat is an animal and has a fur. All animals have skin and can move. Giraffe is an animal who is tall and has long legs. Parrot is a bird and is green in color”.



https://www.slideserve.com/kacy/knowledge-representation-techniques

FRAME STRUCTURES

- Semantic networks morphed into Frame representation Languages in the ‘70s and ‘80s.
- A frame is a lot like the notion of an object in OOP, but ha more meta-data.
- Represents related knowledge about a subject
- A frame has a set of slots.
- A slot represents a relation to another frame (or value)
- A slot has one or more facets.
- A facet represents some aspect of the relation.
- Facet : A slot in a frame holds more than a value



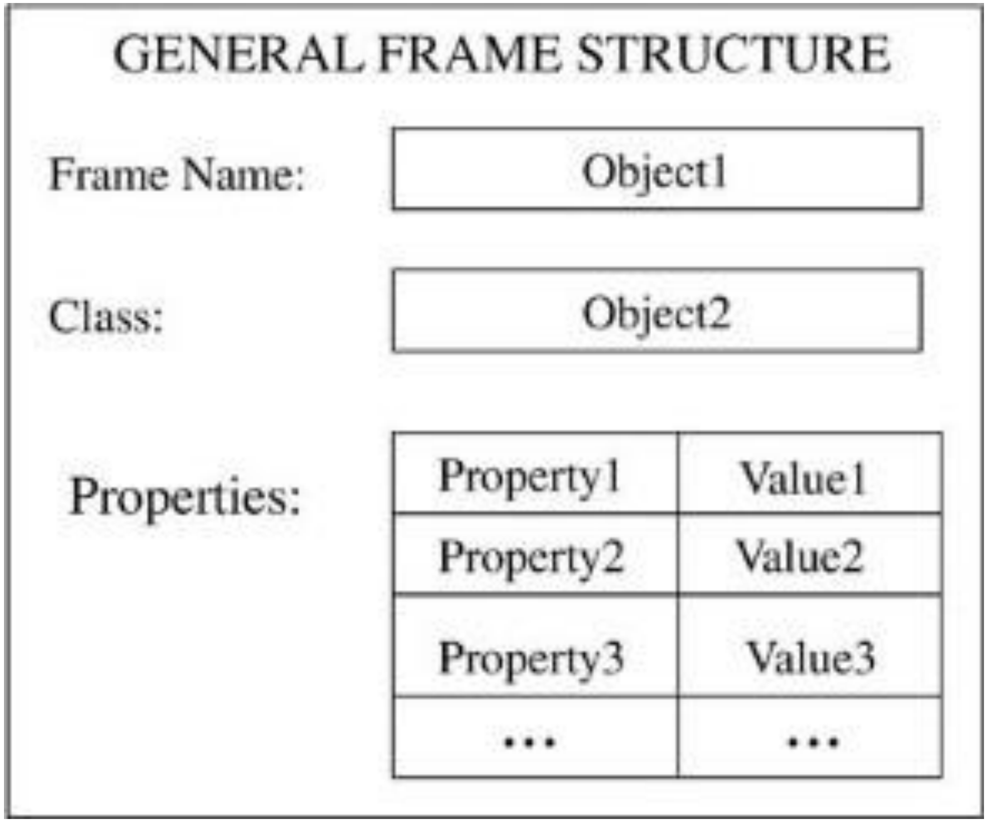
Truck	
number_of_wheels	value: Integer range = 4..12 access function = count_wheels
length	value: Real units = metres
location	value: Place

Frame

- A data structure for representing stereotypical knowledge of some concept

- A frame is a collection of attributes and associated values that describe some entity in the world.

- Frames are general record like structures which consist of a collection of slots and slot values.



GENERAL FRAME STRUCTURE

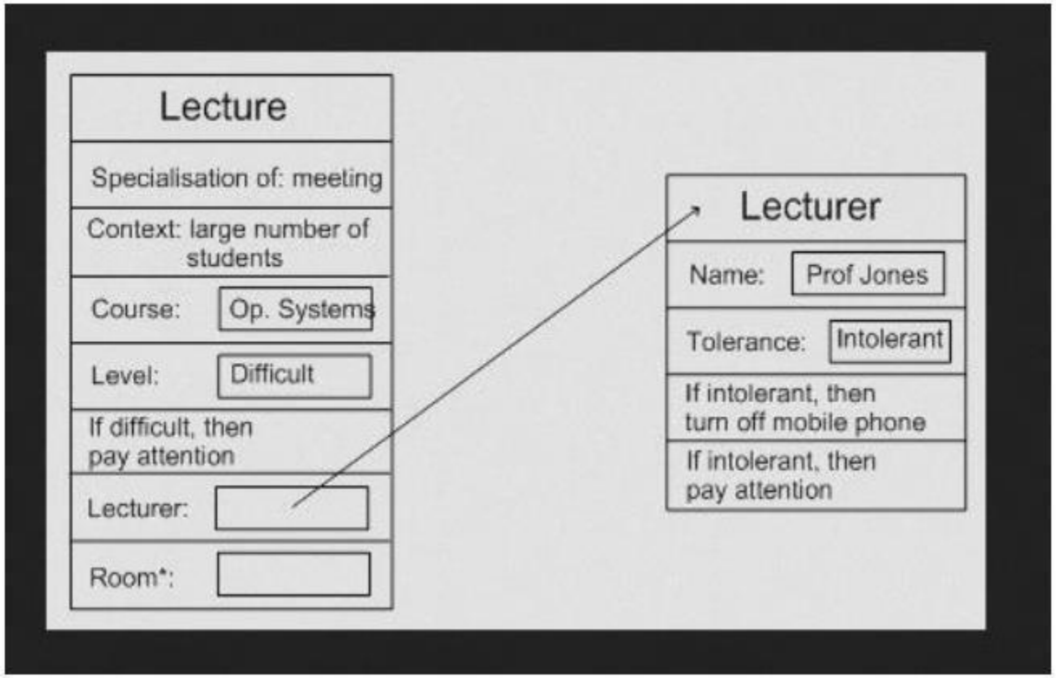
Frame Name: Object1

Class: Object2

Properties:

Property1	Value1
Property2	Value2
Property3	Value3
...	...

Frame Structures



Lecture

Specialisation of: meeting

Context: large number of students

Course: Op. Systems

Level: Difficult

If difficult, then pay attention

Lecturer:

Room*:

Lecturer

Name: Prof Jones

Tolerance: Intolerant

If intolerant, then turn off mobile phone

If intolerant, then pay attention

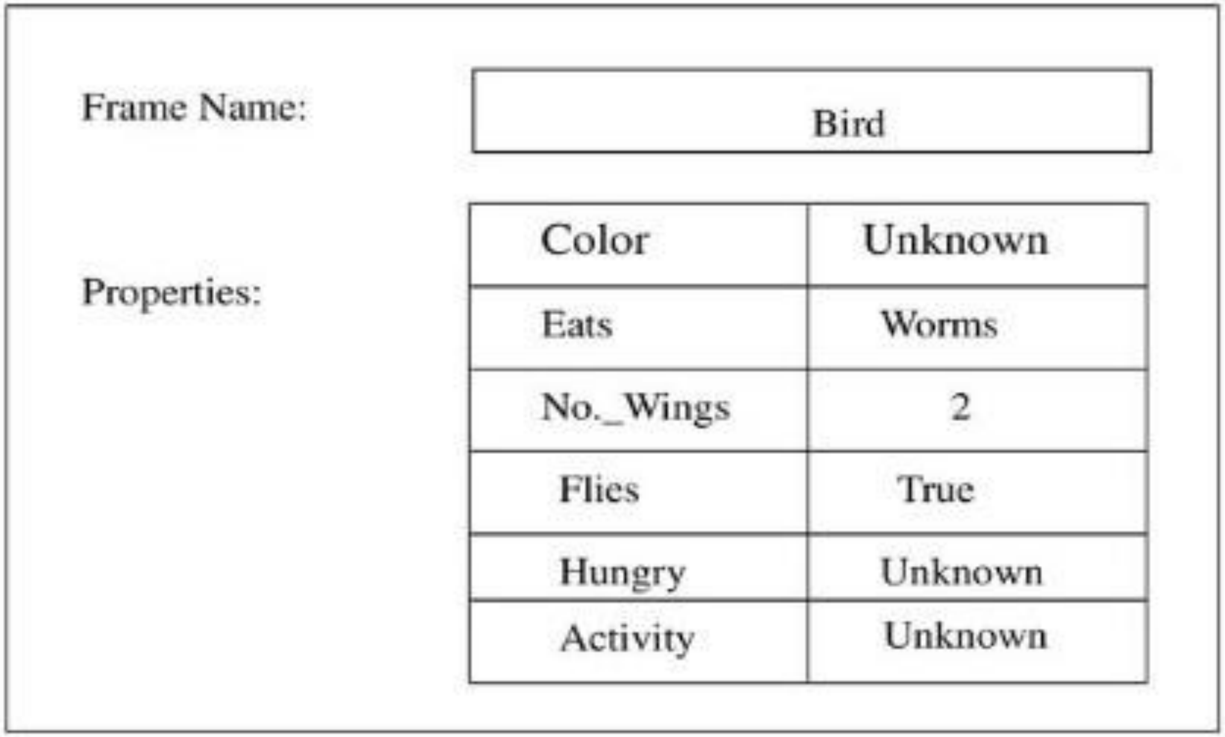
Example :

- (Jones)
- (Profession (Value Lecturer))
- (Age (Value 25))
- (City (Value Yelahanka))
- (State (Value Karnataka))

* Note : value is a Keyword.

Class Frame

A class frame represents the general characteristics of some set of common objects.



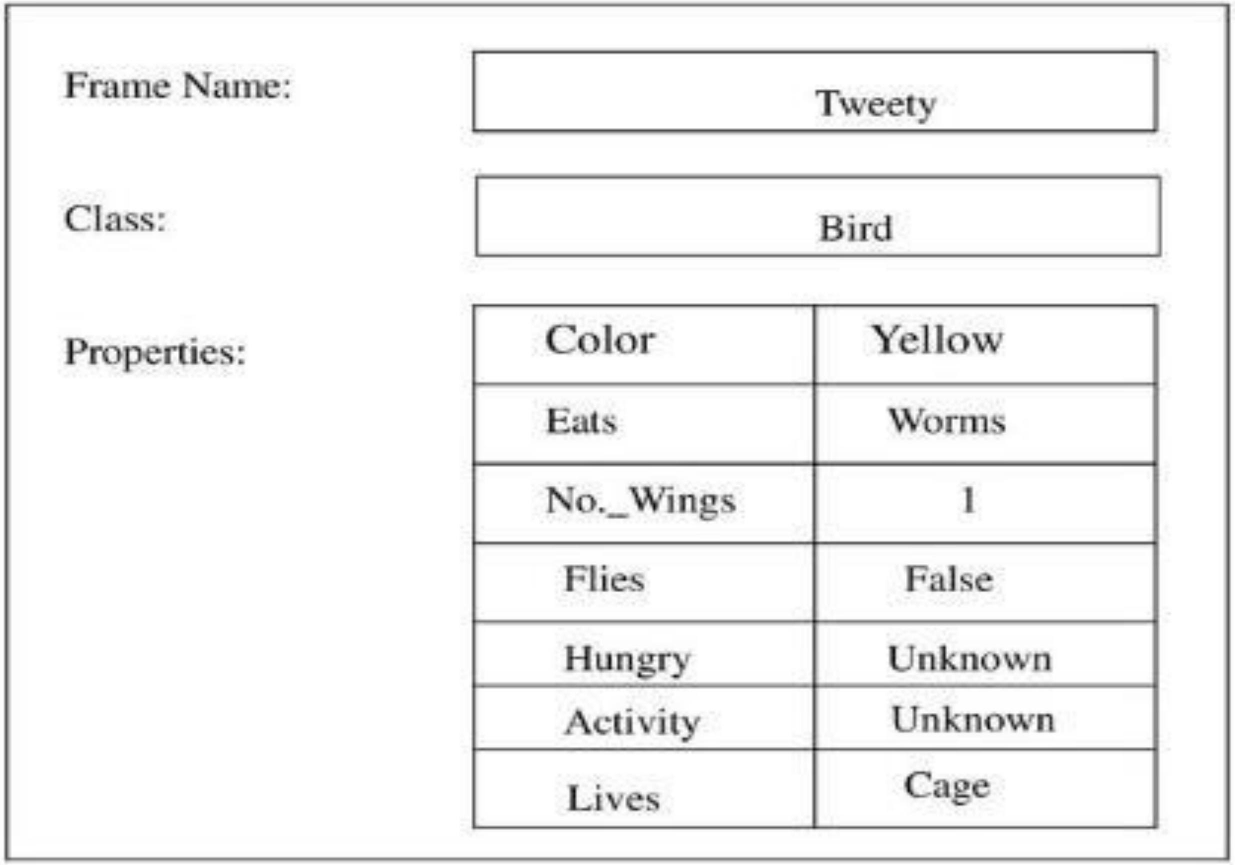
Frame Name: Bird

Properties:

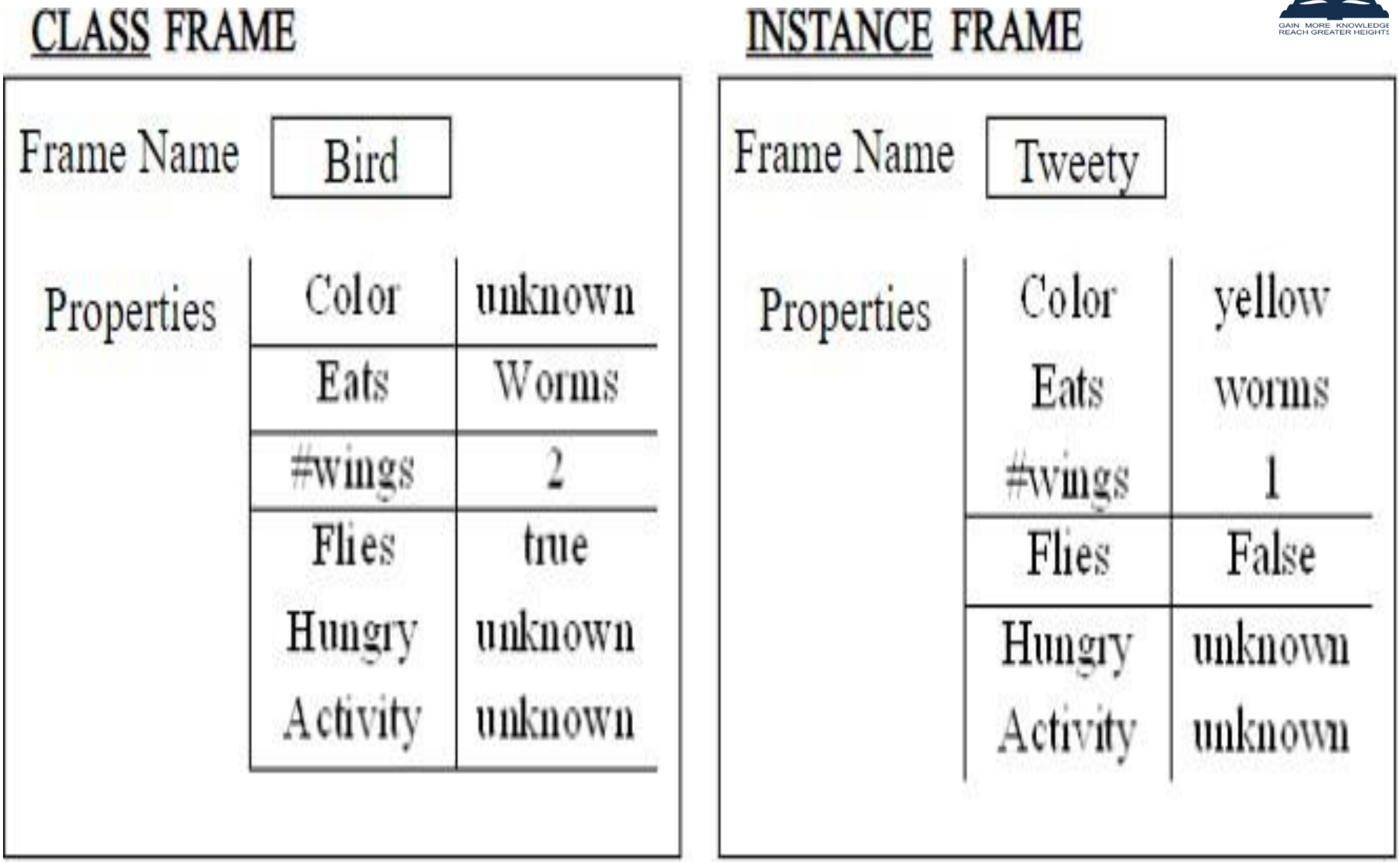
Color	Unknown
Eats	Worms
No._Wings	2
Flies	True
Hungry	Unknown
Activity	Unknown

Instance Frame

Instance Frame describes a specific instance of a class frame. The frame inherits both properties and property values from the class.



Frame Name:	Tweety	
Class:	Bird	
Properties:	Color	Yellow
	Eats	Worms
	No._Wings	1
	Flies	False
	Hungry	Unknown
	Activity	Unknown
	Lives	Cage



CLASS FRAME

Frame Name: Bird

Properties

Color	unknown
Eats	Worms
#wings	2
Flies	true
Hungry	unknown
Activity	unknown

INSTANCE FRAME

Frame Name: Tweety

Properties

Color	yellow
Eats	worms
#wings	1
Flies	False
Hungry	unknown
Activity	unknown

https://slideplayer.com/slide/6072316/

Conceptual Graphs

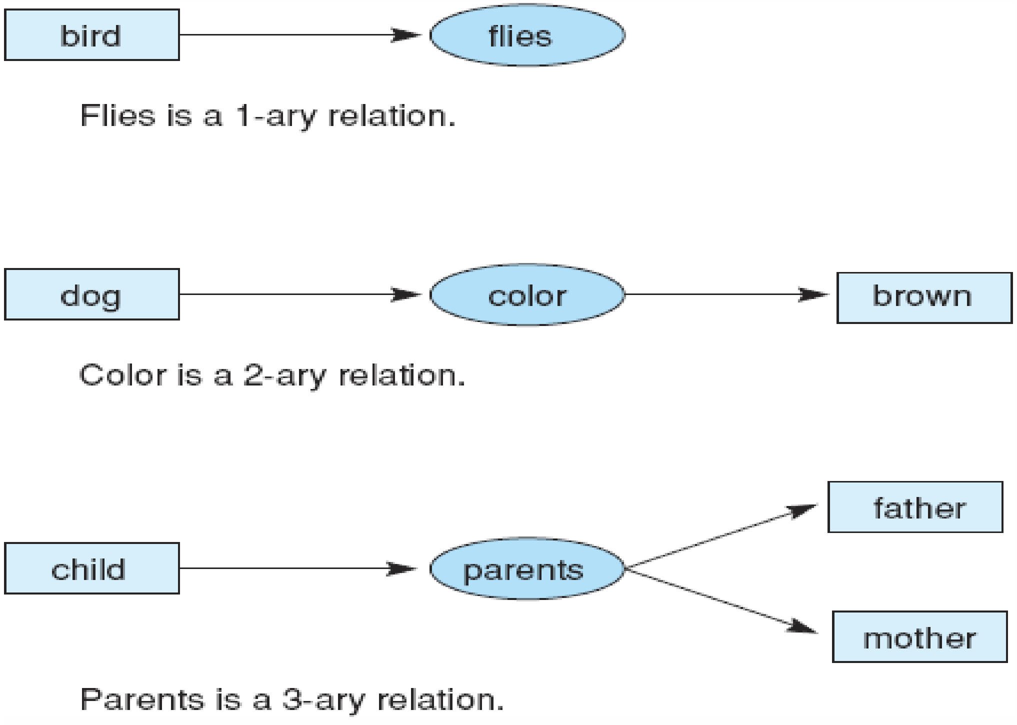
- Semantic Network where each graph represents a single preposition
- Consists of basic concepts and the relationship between them
- KB consists of set of Conceptual Graphs
- It tries to capture the concepts about the events and represents them in the form of a graph.
- A concept may be individual or generic.
- An individual concept has a type field followed by a reference field.
- A single conceptual graph is roughly equivalent to a graphical diagram of a natural language sentence where the words are depicted as concepts and relationships.

Conceptual Graphs

- **Conceptual graph**
  - A finite, connected, bipartite graph.
  - No arc labels, instead the conceptual relation nodes represent relations between concepts
  - Concepts are represented as boxes and conceptual relations as ellipses
  - Nodes
    - **Concept Nodes – box nodes**
      - Concrete concepts:
        - » These concepts are characterized by our ability to form an image of them in our minds.
        - » cat, telephone, classroom
        - » Concrete concepts include generic concepts such as cat or book along with concepts of specific cats and books
      - Abstract objects:
        - » Abstract Concepts that do not correspond to images in our minds
        - » love, beauty, loyalty
    - **Conceptual Relation Nodes – ellipse nodes**
      - Relations involving one or more concepts
      - Some special relation nodes, namely, agent, recipient, object, experiencer, are used to link a subject and the verb
      - Arity – number of box nodes linked to

Conceptual Graphs

- **Example :**



Flies is a 1-ary relation.

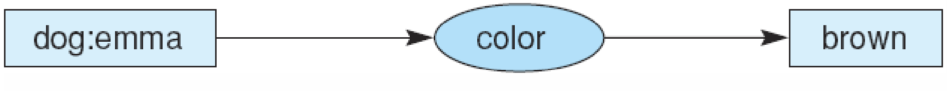
Color is a 2-ary relation.

Parents is a 3-ary relation.

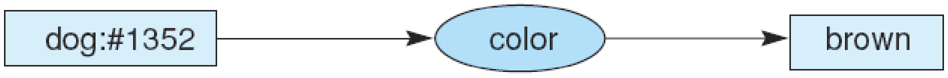
Conceptual Graphs

- **Example :**

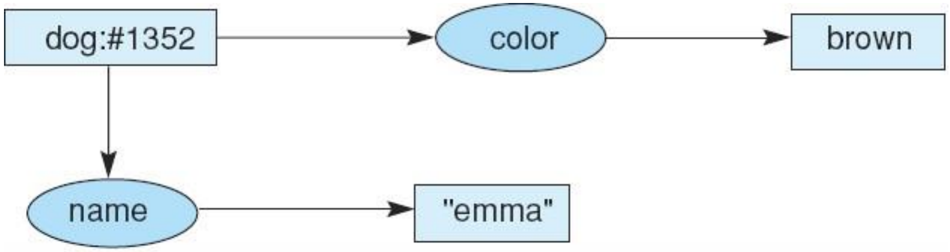
Conceptual graph indicating that the dog named Emma is brown.



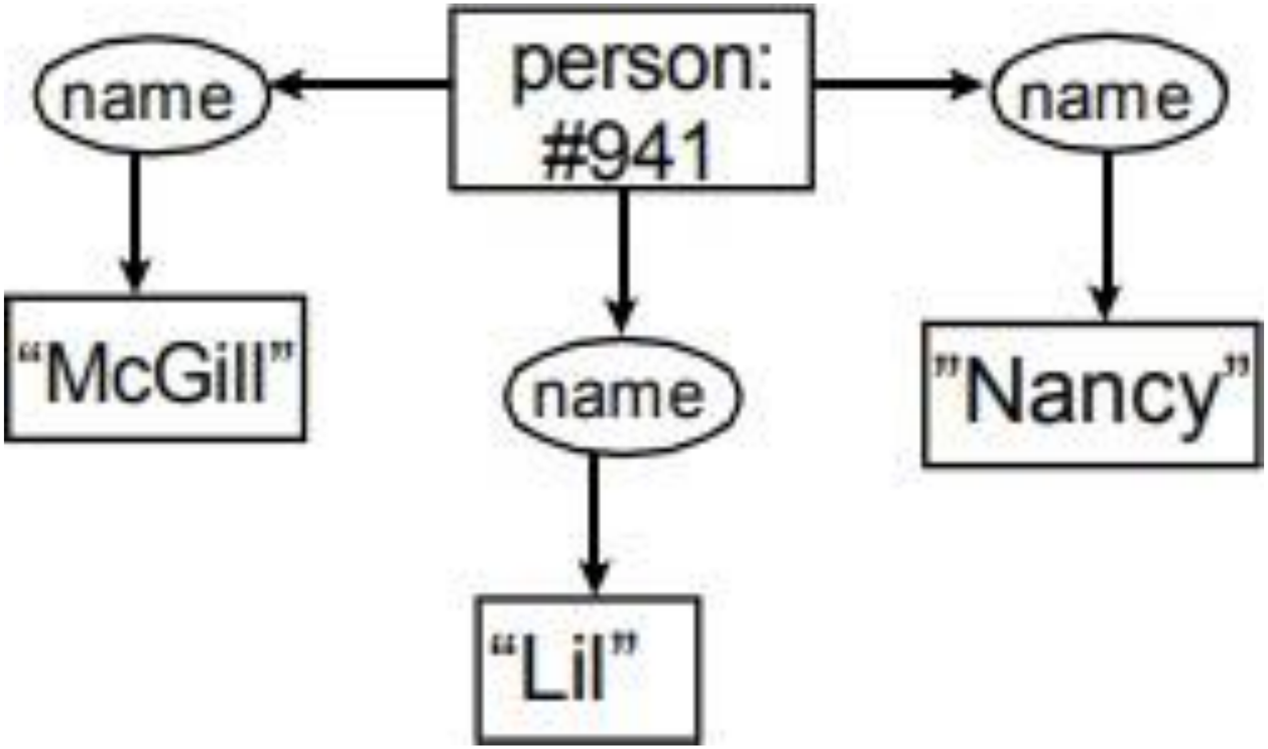
Conceptual graph indicating that a particular (but unnamed) dog is brown.



Conceptual graph indicating that a dog named Emma is brown.



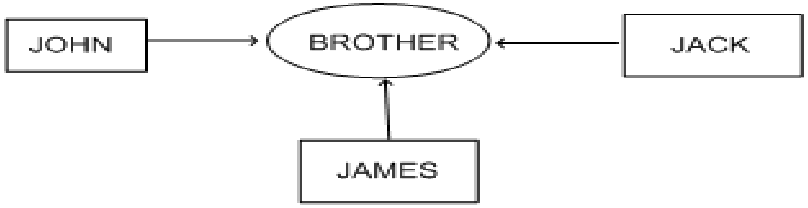
Example: Her name was McGill and she called herself Lil, but everyone knew her as Nancy



Conceptual Graphs

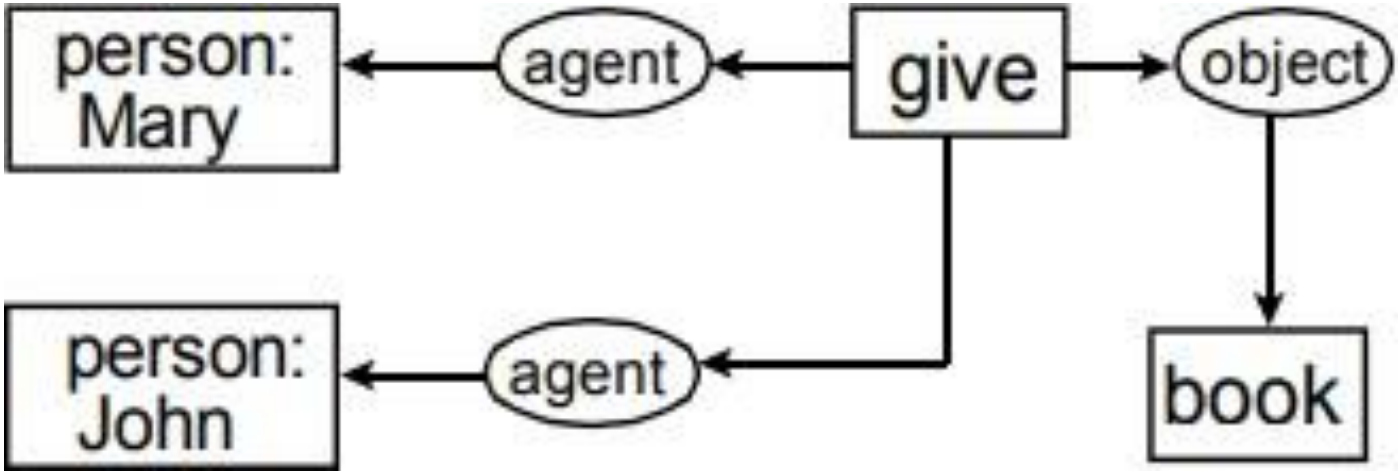
- **Example :**

Each graph represents a single proposition.



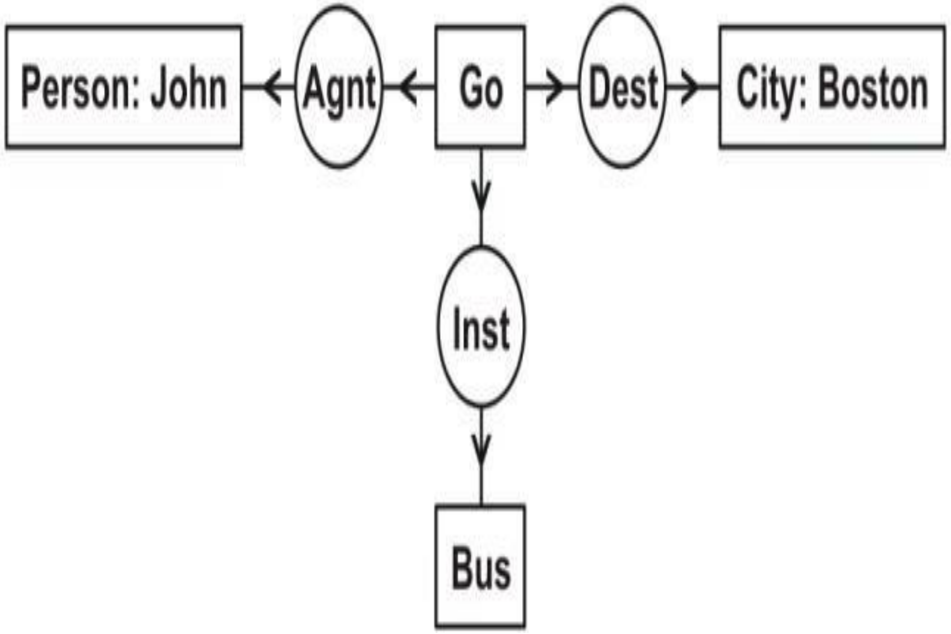
- Advantage:
  - – Single relationship between multiple concepts is easily representable.

Example: Mary gave John the book

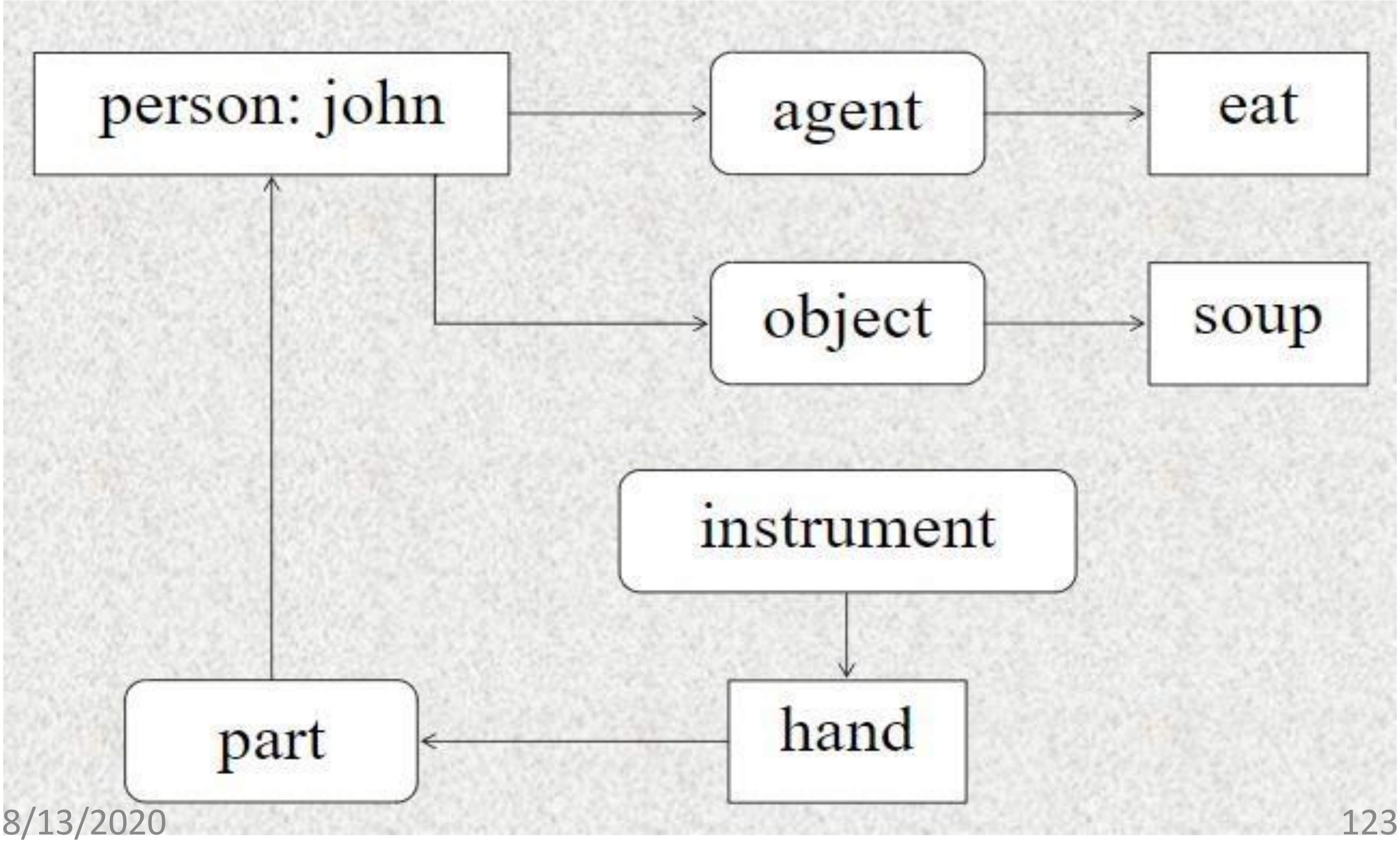


Example: John is going to Boston by bus

- Each of the four concepts has a type label, which represents the type of entity the concept refers to**: Person, Go, Boston, or Bus**.
- Two of the concepts have names, which identify the referent: John or Boston.
- Each of the three conceptual relations has a type label that represents the type of relation: agent (Agnt), destination (Dest), or instrument (Inst).
- The CG as a whole indicates that the person John is the agent of some instance of going, the city Boston is the destination, and a bus is the instrument.



Example: John agent eat object soup instrument hand part



Summary

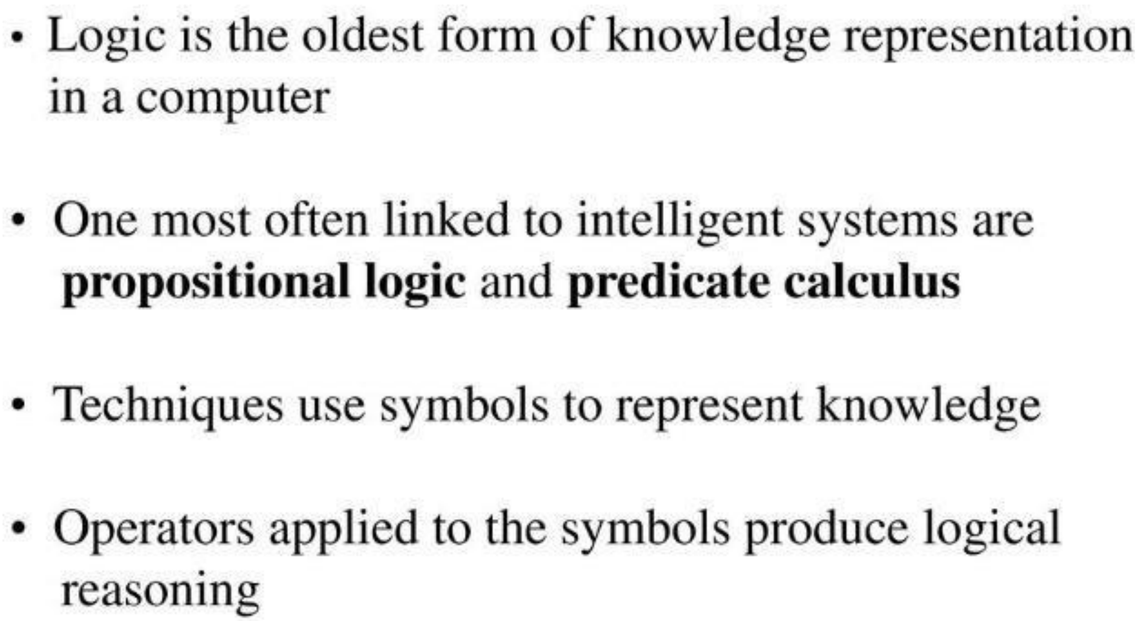
- Graphs are **very easy to store** inside a computer, by structuring knowledge in a useful fashion
- An **ontology** formally represents concepts in a **domain and relationships**

  between those concepts
- The concept originated in philosophy; a model of a **theory of nature** or

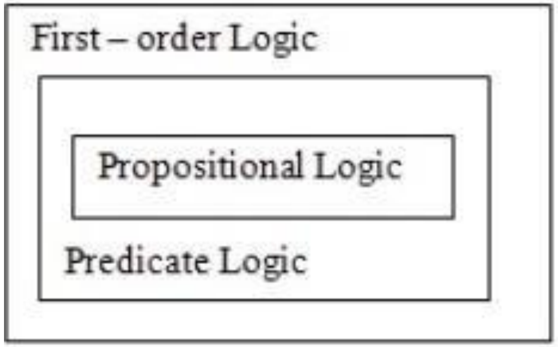
  existence.
- An ontology describes the things we want to talk about, **including both**

  objects and relationships

Logic



- Logic is the oldest form of knowledge representation in a computer
- One most often linked to intelligent systems are **propositional logic** and **predicate calculus**
- Techniques use symbols to represent knowledge
- Operators applied to the symbols produce logical reasoning



- **A logic is a formal language, with precisely defined syntax and semantics, which supports sound inference.**
- **Different logics exist, which allow you to represent different kinds of things, and which allow more or less efficient inference.**

Acknowledgement & Disclaimer

- The content in the slides is amalgamated from web.
- Many websites have been visited and different tutorials have been used
- I want to thank my colleagues and students for being so supportive and understanding.
- If you find any mistake do point the slide no and point.

Thank you!