

Knowledge Management

Prepared by Dr. Mohamed Eassa

IT 383
Knowledge Management

Lecture 4



The background features a series of vertical panels with rounded ends. The leftmost panel shows a person's hands typing on a laptop. The subsequent panels contain various white line-art icons on a semi-transparent background: a graduation cap, a play button, a target with an arrow, an open book, a magnifying glass, and a group of people. The main title is positioned to the right of these panels.

Knowledge Engineering Knowledge Representation

Lecture 4

IT 383

Knowledge Management

Prepared by
Dr. Mohamed Eassa



Knowledge Representation



What is knowledge Representation?

Representing the captured knowledge in the form that can be understood by computers then utilized this knowledge to solve the complex problems.

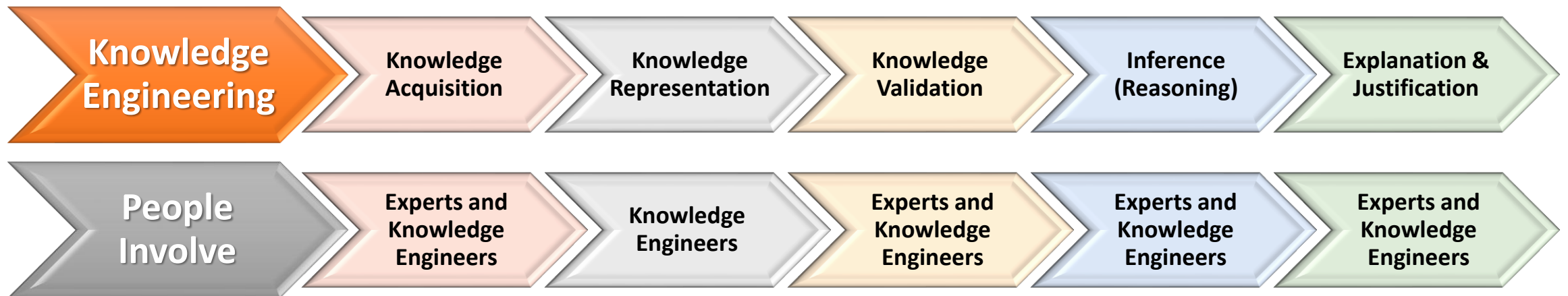


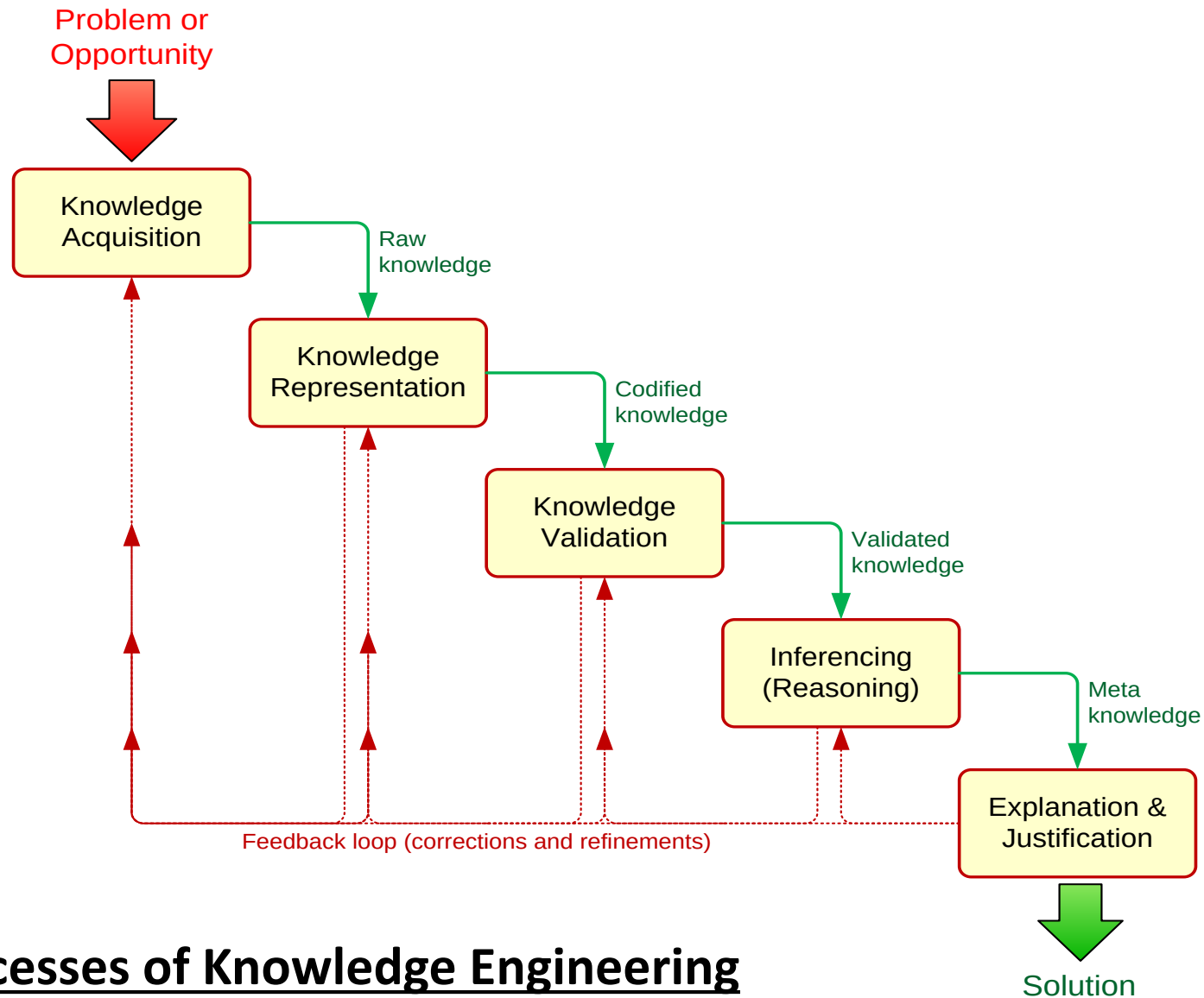
Lecture Objectives

- **Understanding knowledge representation.**
- **Identify most common techniques for knowledge representation.**
- **Understanding Ontology.**

➤ **What is knowledge Engineering?**

A set of intensive activities involved in building, maintaining and using knowledge-based systems.



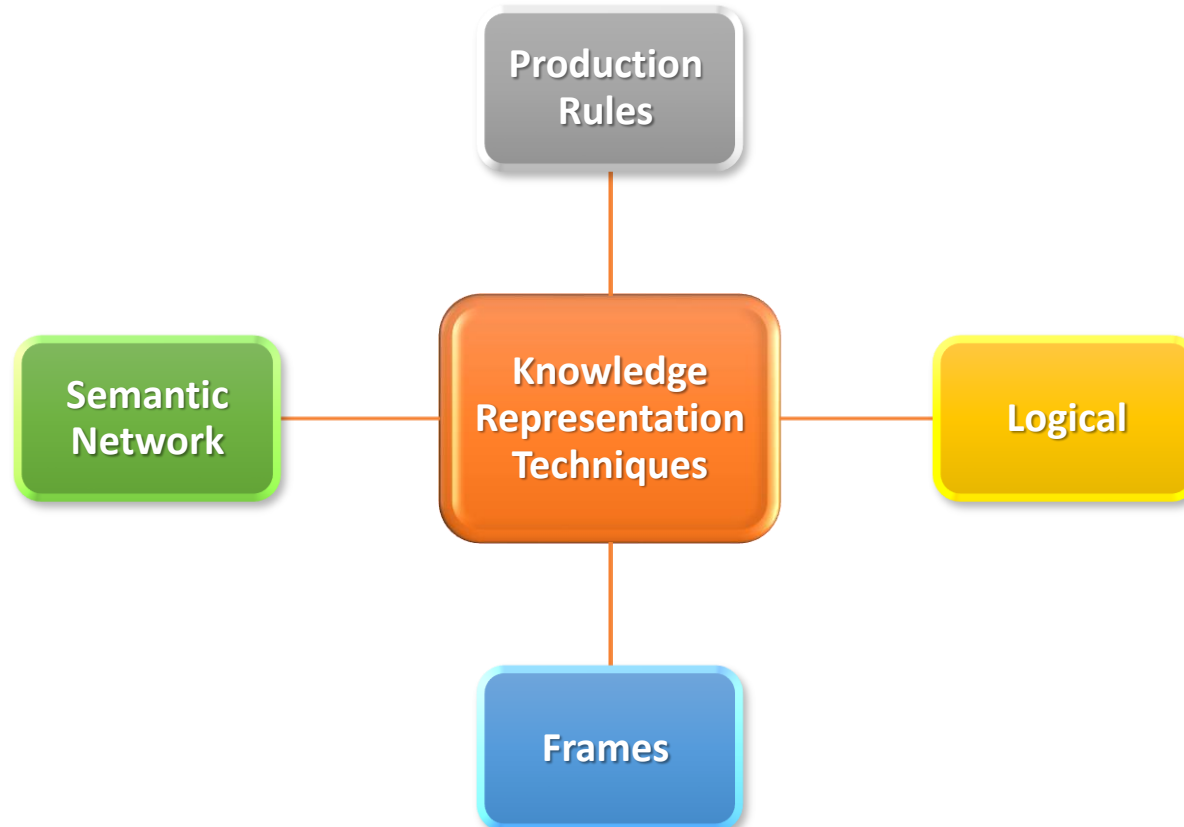


The Processes of Knowledge Engineering

- **Humans are best at understanding, reasoning, and interpreting knowledge. Human knows things, which is knowledge and as per their knowledge they perform various actions in the real world.**
- **Since AI is the science of simulating the human behavior by using machines (computers).**
- **But how can we make machines understand knowledge in order to perform the same action as human does?**

➤ What is knowledge Representation?

Representing the captured knowledge in the form that can be understood by computers then utilized this knowledge to solve the complex problems.



Production Rules

➤ Production Rules:

A set of **if-then rules** - typically state that if certain conditions hold, then some action should be taken.

➤ The rule's structure connects one or more **premises** contained in the **IF part**, to one or more **conclusions** contained in the **THEN part**.

➤ A rule is a form of procedural knowledge.

➤ A rule describes how to solve a problem.

Production Rules

- A rule may have multiple premises joined with **AND** statements, **OR** statements, **NOT** statements or a combination of all.
- In a rule based expert system, domain knowledge is captured in a set of rules and entered in the system's knowledge base.
- the system uses these rules along with information contained in the working memory to solve a problem.
- When the IF portion of the rule matches the information contained in the working memory, the system performs the actions specified in the THEN part of the rule.

Production Rules

- When this happened, 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.

Suppose we have **three facts**: **Raining**, **Outside(Omar)**, $\neg$ **HasUmbrella(Omar)**.

And we have a rule in KB: **IF Raining $\wedge$ Outside(x) $\wedge$ $\neg$ HasUmbrella(x) THEN GetWet(x)**

If **x=Omar** matches the facts, so we are able to infer **GetsWet(Omar)**:

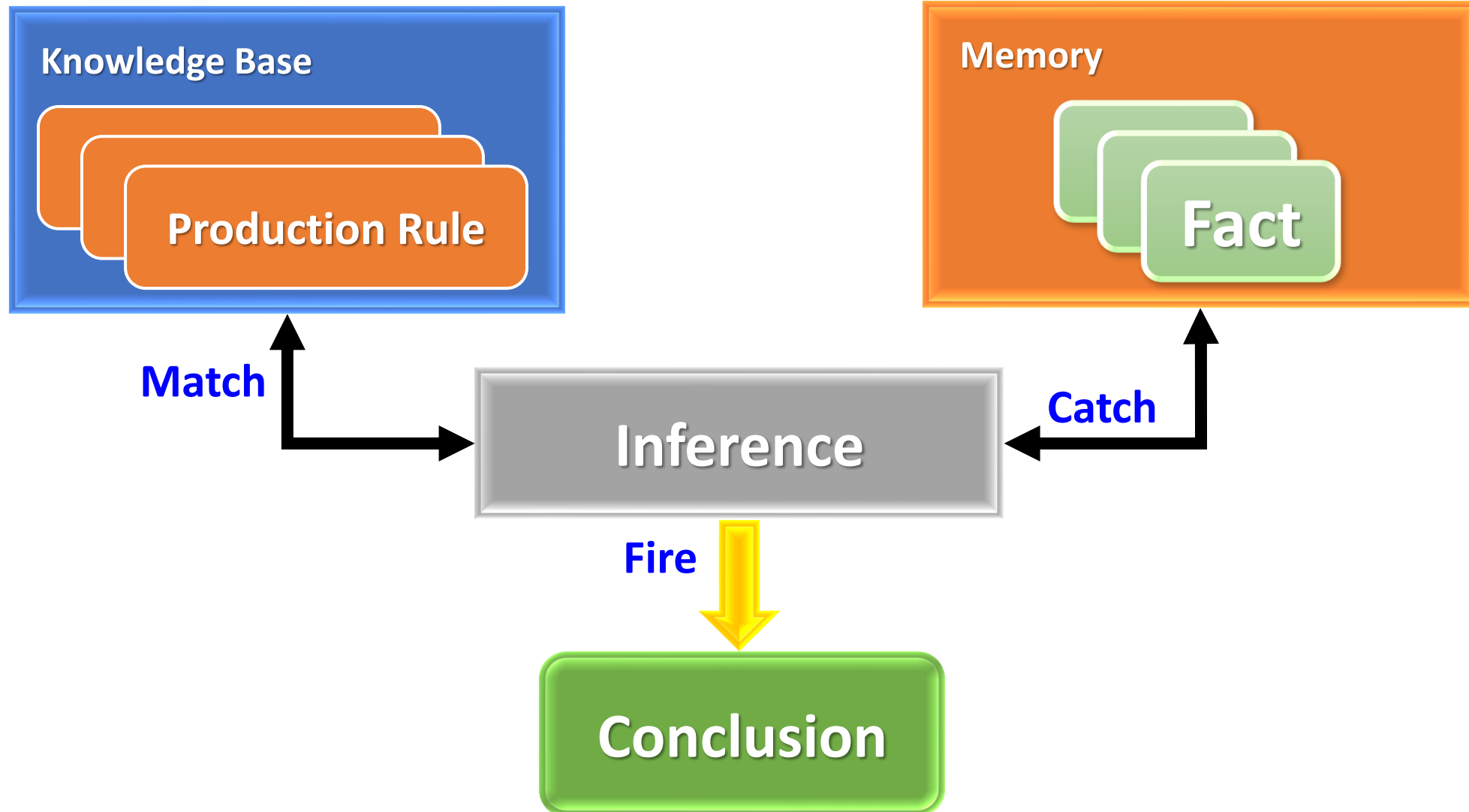
IF Raining $\wedge$ Outside(Omar) $\wedge$ $\neg$ HasUmbrella(Omar) THEN GetWet(Omar)

This means we now have **four facts** in our knowledge base:

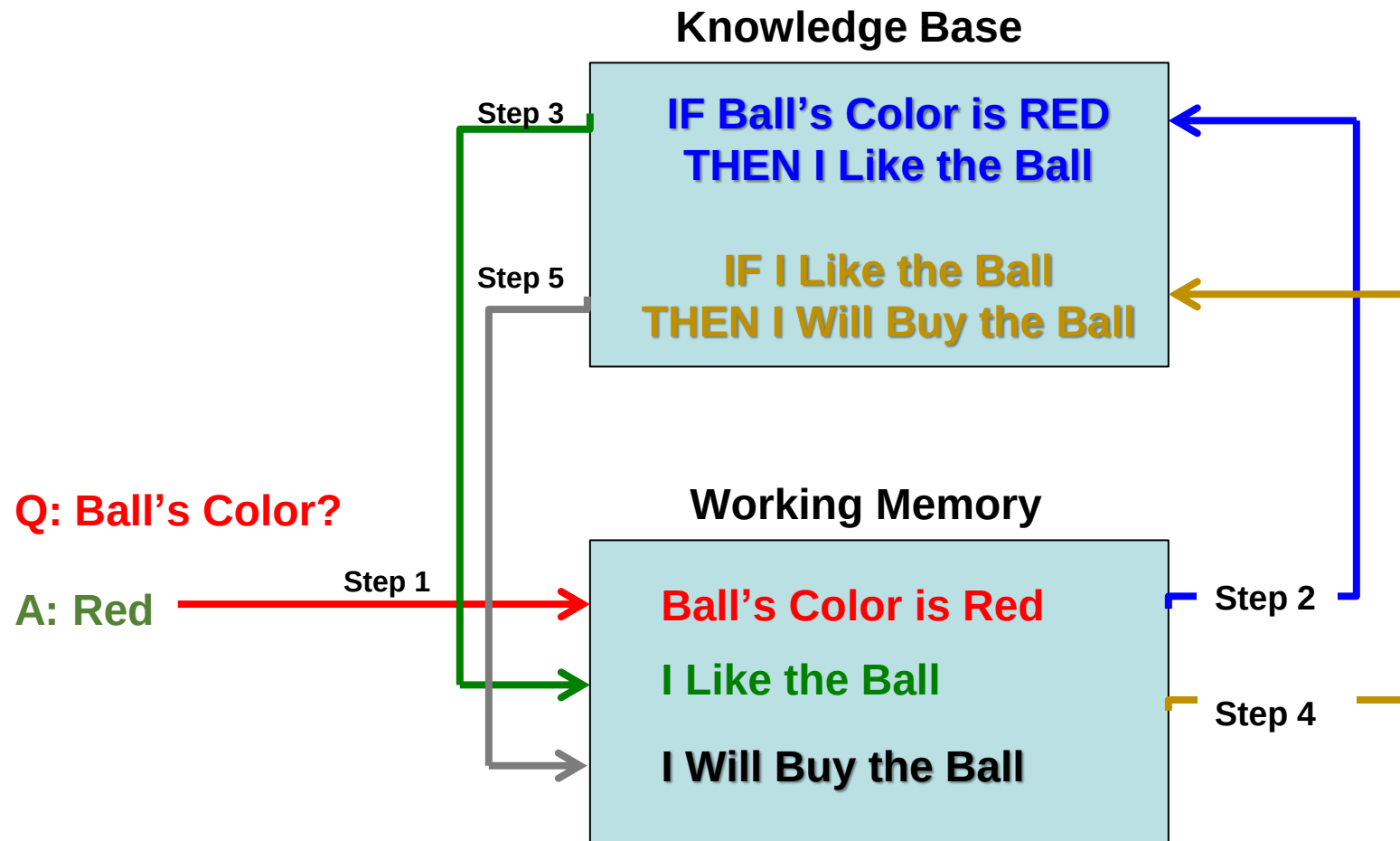
Raining, **Outside(Omar)**, $\neg$ **HasUmbrella(Omar)**, and **GetWet(Omar)**

$\neg A$	NOT A (negation)
$A \wedge B$	A AND B (Conjunction)
$A \vee B$	A OR B (Disjunction)
$A \rightarrow B$	If A then B "A implies B" (Implication)
$A \leftrightarrow B$	A if and only if B (Equivalence)
$\exists A$	There exist A
$\forall A$	For all A (universal quantifier)

Production Rules



Production Rules



Production Rules

- The process of deriving new facts from given facts is called **inference**, and is managed by a module known as “**inference engine**”
- Besides concluding new information, a rule can perform some operations :
IF The area of the square is needed
THEN $\text{area} = \text{Length} * \text{width}$
- In order to perform more complex operations, most rule based systems are designed to access external programs, e.g., Database.

Production Rules

➤ Rules Types:

➤ Relational:

IF the battery is dead.

THEN the car will not start.

➤ Recommendation:

IF the car will not start

Then take a Taxi.

➤ Directive:

IF the car will not start **AND** the fuel system is OK.

THEN Check out the electrical system.

➤ Heuristic:

IF the car will not start **AND** the car is a 1971 VW.

THEN check the battery

Production Rules

➤ Rules can be also categorized according to the nature of the problem solving strategy:

➤ Interpretation problem:

IF Voltage is greater than 220 volts **AND** the Power is 130 WATT.
THEN refrigerator operation is in the normal range.

➤ Diagnosis problem:

IF the patient Symptoms dry cough **AND** temperature is greater than 38 **AND** suffer from asthma **AND** location is China.
THEN the patient has COVID-19.

➤ Design problem:

IF Current task is assigning a power supply **AND** The Position of the power supply in the cabinet is known **AND** There is space in the cabinet for the power supply.
THEN Put the Power Supply in the cabinet .

Logical Representation

➤ **Logical Representation:**

Logical representation is a language with some concrete rules that has no ambiguity in representation. It consists of precisely defined sentences. Each sentence can be translated into logics using syntax and semantics.

First-Order Logic

Propositional Logic

Predicate Logic

Logical Representation

➤ Basics:

$\neg A$	NOT A (negation)
$A \wedge B$	A AND B (Conjunction)
$A \vee B$	A OR B (Disjunction)
$A \rightarrow B$	If A then B “A implies B” (Implication)
$A \leftrightarrow B$	A if and only if B (Equivalence)
$\exists A$	There exist A
$\forall A$	For all A (universal quantifier)

Logical Representation

➤ Propositional logic:

It is the branch of logic that studies ways of joining and/or modifying entire statements or sentences to form more complicated statements or sentences, as well as the logical relationships and properties that are derived from these methods of combining or altering statements.

Example:

A: It is cold

B: It is snowing

C: It is winter

D: It is sunny

To represent the following rules using propositional logic;

R1: “if it is snowing then it is cold”

R2: “if it is cold and it is snowing then it is not sunny”

R1: $B \rightarrow A$

R2: $A \wedge B \rightarrow \neg D$

Logical Representation

➤ Predicate logic:

It is a formal system that uses variables and quantifiers ($\forall, \exists$) to formulate propositions.

Example 1:

1- All humans are mortal.

2- Omar is a human. (Nada is a human, Ahmed is a human,...etc.)

Therefore;

3- Omar is mortal

To represent the previous statements using predicate logic;

(Omar, Nada, and Ahmed are variables, may represent by variable "X"), so,

$\forall X [is_a_human(X) \rightarrow is_mortal(X)]$

Where X is a variable = (Omar, Nada, or Ahmed)

Frame Representation

➤ Frame Representation:

A data structural for representing all knowledge about a particular object.

Content of the frame:

- ❖ It has a name
- ❖ Slots: a set of attributes
- ❖ Facets:
 - attribute values
 - Default
 - Range
 - If needed
 - If changed

Frame Name

Bird

Slots:

color	unknown
Eats	Worms
No_Wings	2
Flies	True
Hungry	Unknown
Activity	Unknown

Frame Representation

➤ Instance (Child) frame:

Describe a specific instance of a class frame. It inherits both properties and property values from the class.

Frame Name:

Tweety

Class:

Bird

Slots:

Color	yellow
Eats	Worms
No_Wings	2
Flies	Flies
Hungry	Unknown
Activity	Unknown

Frame Name

Bird

Slots:

color	unknown
Eats	Worms
No_Wings	2
Flies	True
Hungry	Unknown
Activity	Unknown

Frame Representation

➤ Facet:

Provide you with additional control over property values, and Can instruct a property how to obtain its value or what to do if its value changes.

➤ IF-Needed Facet:

The facet is used in a case when no slot value is given.

```
IF Tweety:no_wings <2  
Then tweety:Flies = False  
IF Tweety:no_wings =2  
THEN Tweety:Flies = True
```

➤ IF-Changed Facet:

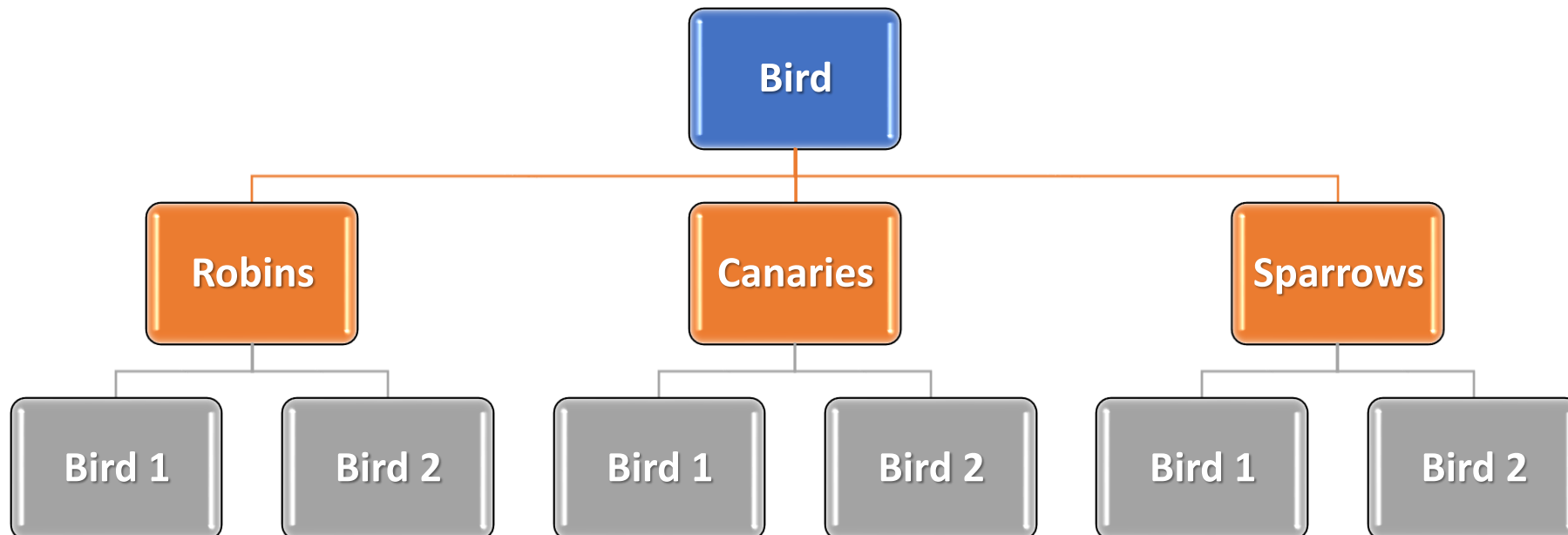
A method can also be attached to a that performs some action whenever its value changes.

```
IF Self: Hungry = True  
Then Self: Activity = Eating
```

Frame Representation

➤ Hierarchy of Frame:

Creating complex frame structures organizing by different levels of abstraction. Each level inherits from the previous level.



Semantic Network Representation

➤ Semantic Networks Representation:

Provides a graphical view of a problem's important objects, properties and relationships.

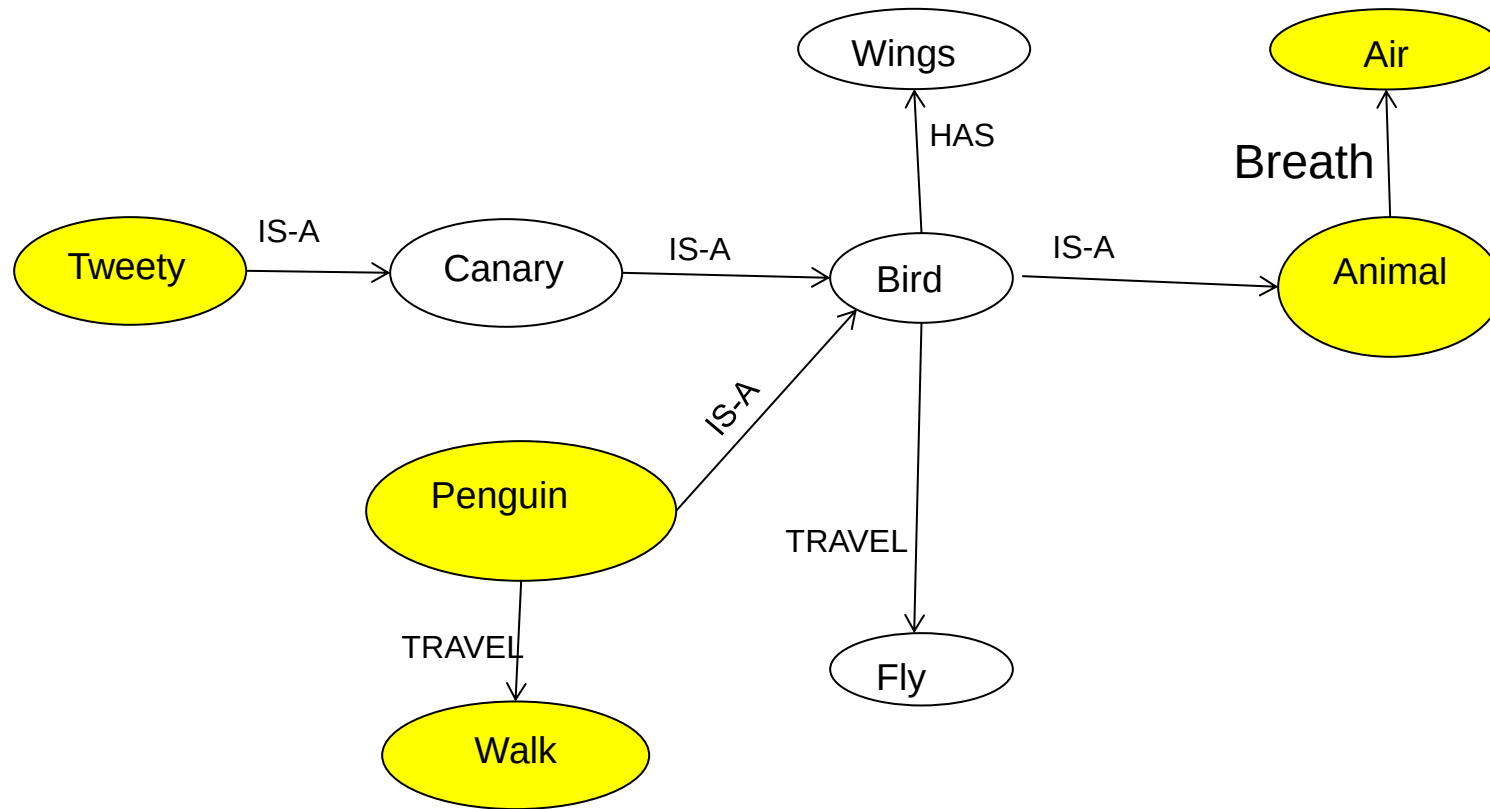
➤ It contains nodes and arcs that connect the nodes that represent the relationships between the nodes.

➤ The node can represent:

- Objects
- Object properties
- Property value

➤ Both the nodes and arcs have labels that clearly describe the objects represented and their natural relationships.

Semantic Network Representation



➤ **Ontology:**

It is a data model that represents knowledge as a set of concepts within a domain and the relationship between these concepts.

The main components of an ontology are:

Concepts: types of classes, objects, or kinds of things.

Relations: ways in which classes and individuals can be related to one another

Axioms: including rules in a logical form that together comprise the overall theory that the ontology describes in its domain of application.

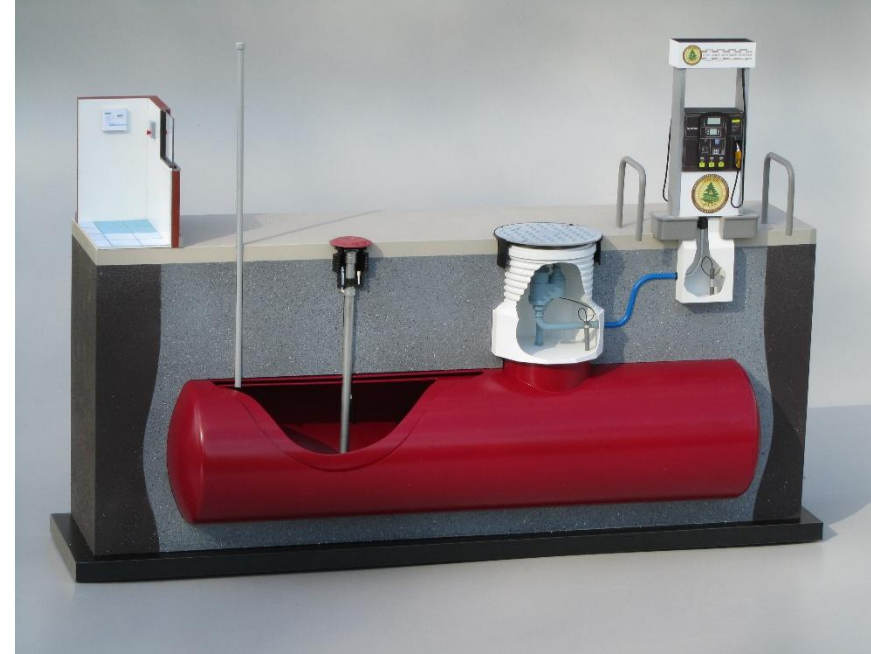
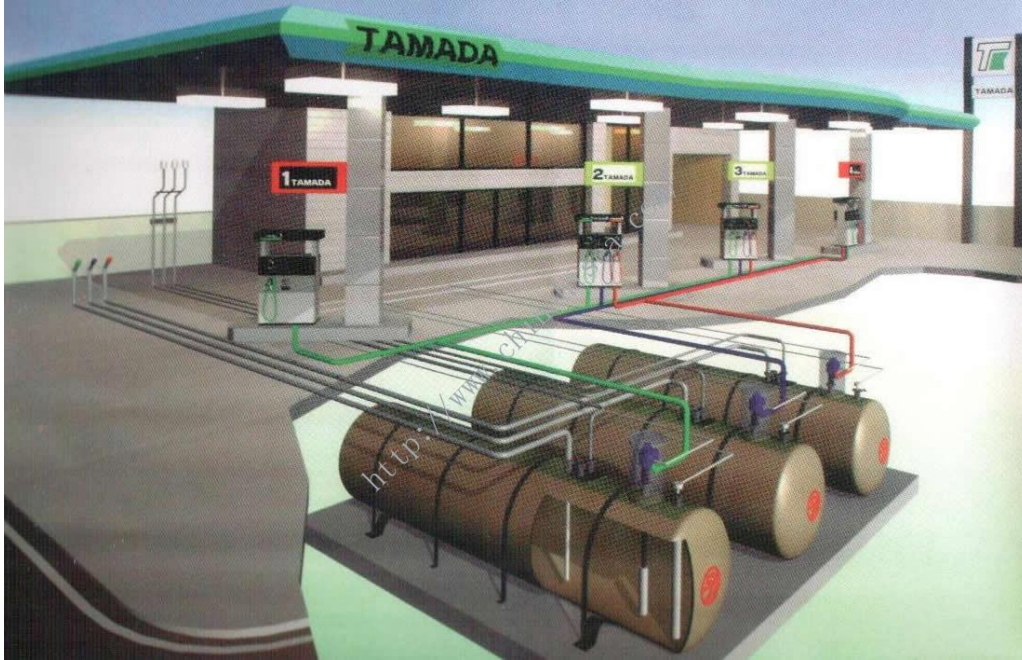
Ontologies are commonly encoded using ontology languages (OWL, XML,...).

Fuel station ontology schema



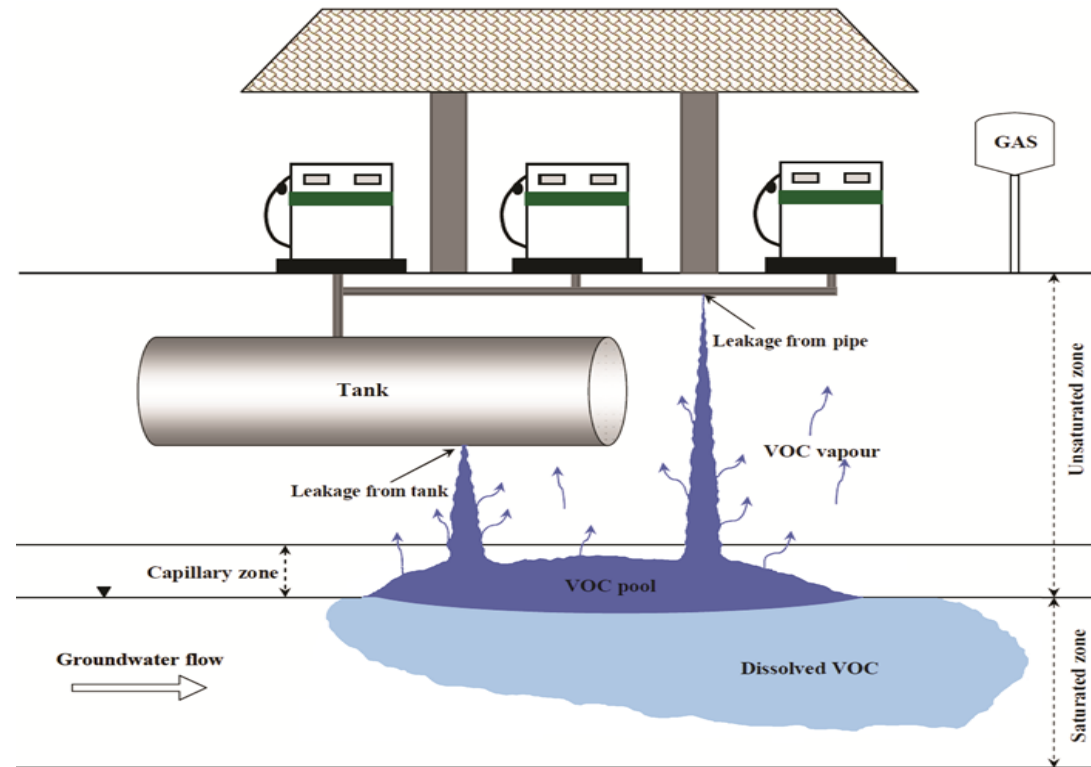
➤ Fuel Island:

It is an area in fuel station within where the fuel dispensers are fixed, cars, trucks and motorcycle are directed to fuel island for refuel.



➤ Reservoir:

- It is underground large fuel tank (reservoir) with fuel capacity range from 25000 to 40000 m³.
- It may be supply fuel to one or more fuel island.
- Each reservoir can hold only one fuel type.



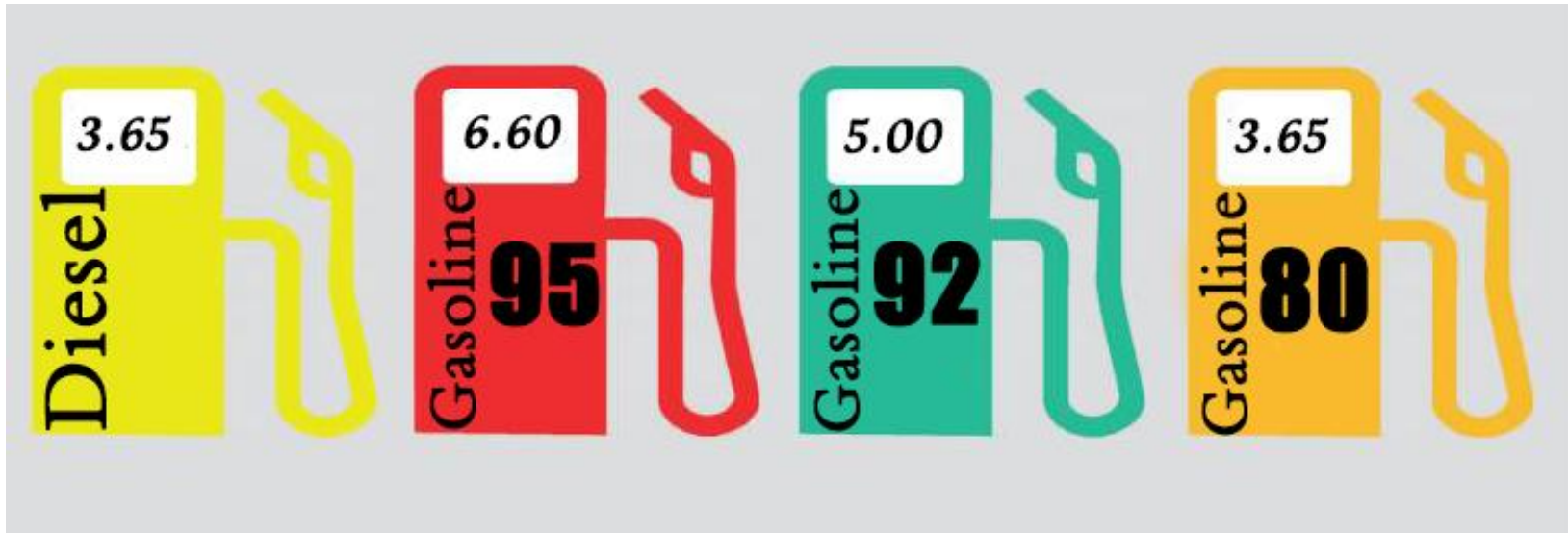
➤ Pumps:

They are applied to the underground fuel reservoir to pump fuel to the fuel nozzle. One pump may supply fuel to more than one fuel nozzle.



➤ **Calibration:**

Each fuel nozzle is subjected to calibration, this process occurred by station engineer by using standard calibration vessel (20L) then match the readings of the dispenser counter with that of the vessel.



- **Fuel type:**
Such as gasoline 80, 92, 95 and diesel.



Manager



Engineer



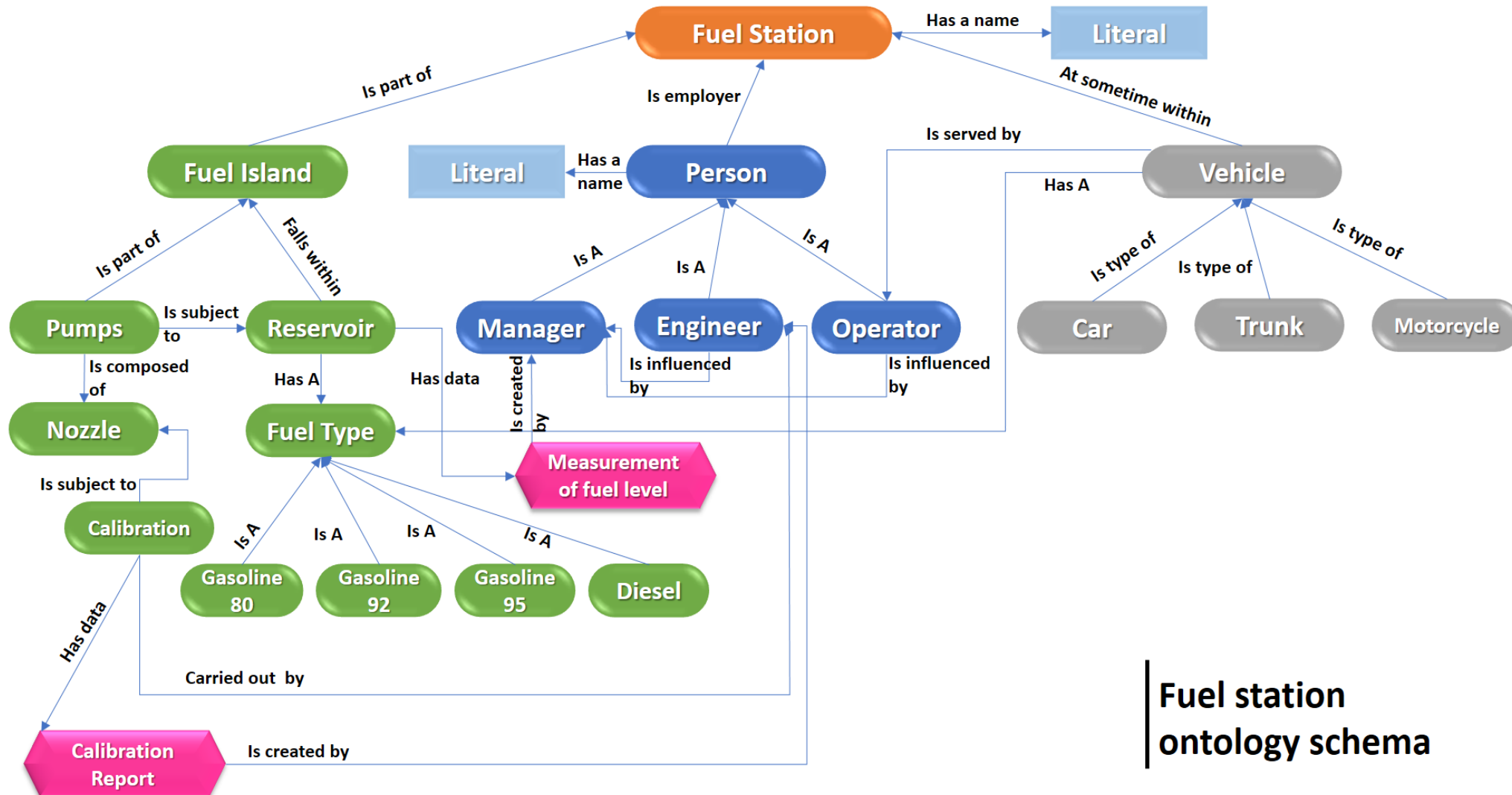
Operator

- **Fuel station employees:**
Each fuel station has a manager, engineer and operators.



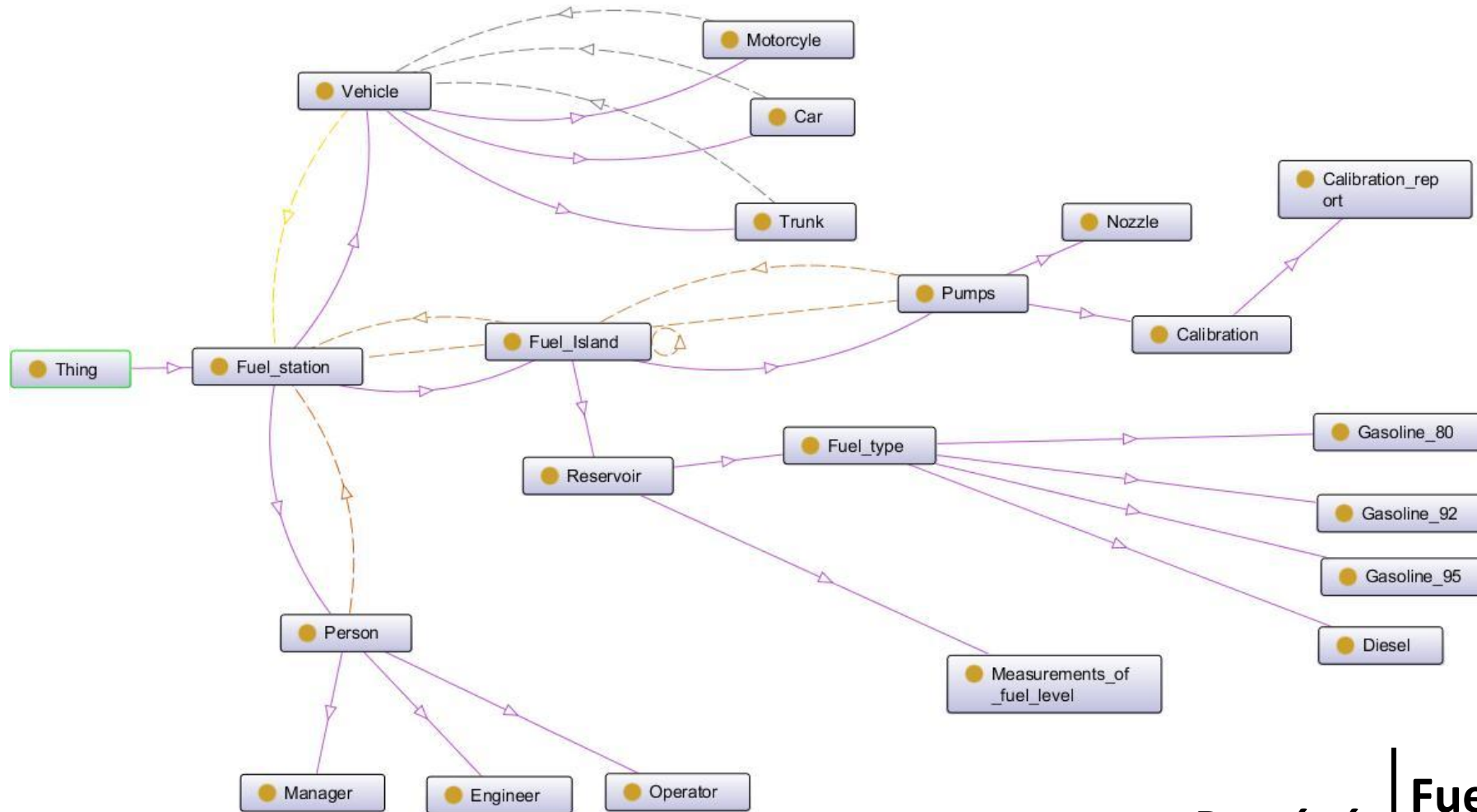
➤ **Vehicle:**

Such as cars, trunks and motorcycle, sometime they are within fuel station to request refuel.



Fuel station
ontology schema

Ontology



Protégé

Fuel station
ontology schema

Summary

- **Understanding the knowledge representation.**
- **Understanding the most common techniques for knowledge representation.**
- **Understanding Ontology**



Thank You