[AI Notes - UNIT - 5 - Artificial intelligence](https://www.studocu.com/in/document/vignans-institute-of-management-and-technology-for-women/computer-science-engineering/ai-notes-unit-5-artificial-intelligence/99792575?utm_campaign=shared-document&utm_source=studocu-document&utm_medium=social_sharing&utm_content=ai-notes-unit-5-artificial-intelligence)

UNIT - V

# Expert Systems:

**Expert** **system** **=** **knowledge** **+** **problem-solving** **methods** A knowledge base that

captures the domain−specific knowledge and an inference engine that consists of algorithms for manipulating the knowledge represented in the knowledge base to solve a problem presented tothe system.

Expert systems (ES) are one of the prominent research domains of AI. It is introduced by theresearchers at Stanford University, Computer Science Department.

What are Expert Systems?

The expert systems are the computer applications developed to solve complex problems in aparticular domain, at the level of extra−ordinary human intelligence and expertise.

Characteristics of Expert Systems

1. High performance
2. Understandable
3. Reliable
4. Highly responsive

## Capabilities of Expert Systems

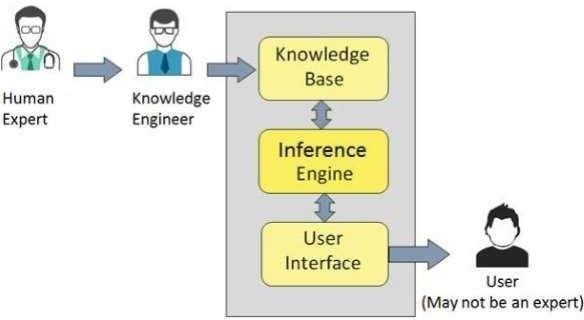
The expert systems are capable of −

* + Advising
  + Instructing and assisting human in decision making
  + Demonstrating
  + Deriving a solution
  + Diagnosing
  + Explaining
  + Interpreting input
  + Predicting results
  + Justifying the conclusion
  + Suggesting alternative options to a problem They are incapable of –
  + Substituting human decision makers
  + Possessing human capabilities
* Producing accurate output for inadequate knowledge base
* Refining their own knowledge.

## Components of Expert Systems

The components of ES include –

1. Knowledge Base
2. Inference Engine
3. User Interface



**Knowledge** **Base**

It contains domain−specific and high−quality knowledge. Knowledge is required to exhibit intelligence. The success of any ES majorly depends upon the collection of highly accurate and precise knowledge.

**What** **is** **Knowledge?**

The data is collection of facts. The information is organized as data and facts about the task domain. Data, information, and past experience combined together are termed as knowledge. Components of Knowledge Base

The knowledge base of an ES is a store of both, factual and heuristic knowledge.

* Factual Knowledge − It is the information widely accepted by the Knowledge Engineers and scholars in the task domain.
* Heuristic Knowledge − It is about practice, accurate judgement, one’s ability of evaluation, and

guessing.

**Knowledge** **representation**

It is the method used to organize and formalize the knowledge in the knowledge base. It is in the form of IF−THEN−ELSE rules.

**Knowledge** **Acquisition**

The success of any expert system majorly depends on the quality, completeness, and accuracy of the information stored in the knowledge base.

The knowledge base is formed by readings from various experts, scholars, and the Knowledge Engineers. The knowledge engineer is a person with the qualities of empathy, quick learning, and case analyzing skills.

He acquires information from subject expert by recording, interviewing, and observing him at work, etc. He then categorizes and organizes the information in a meaningful way, in the form of IF−THEN−ELSE rules, to be used by interference machine. The knowledge engineer also monitors the development of the ES.

**Inference** **Engine**

Use of efficient procedures and rules by the Inference Engine is essential in deducting a correct, flawless solution.

In case of knowledge−based ES, the Inference Engine acquires and manipulates the knowledge from the knowledge base to arrive at a particular solution.

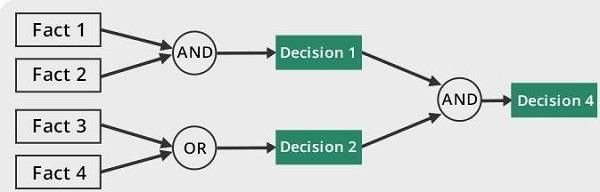
In case of rule based ES, it −

* Applies rules repeatedly to the facts, which are obtained from earlier rule application.
* Adds new knowledge into the knowledge base if required.
* Resolves rules conflict when multiple rules are applicable to a particular case. To recommend a solution, the Inference Engine uses the following strategies –
  1. Forward Chaining
  2. Backward Chaining

## Forward Chaining

It is a strategy of an expert system to answer the question, **“What can happen next?”**

Here, the Inference Engine follows the chain of conditions and derivations and finally deduces the outcome. It considers all the facts and rules, and sorts them before concluding to a solution. This strategy is followed for working on conclusion, result, or effect. For example, prediction of share market status as an effect of changes in interest rates.

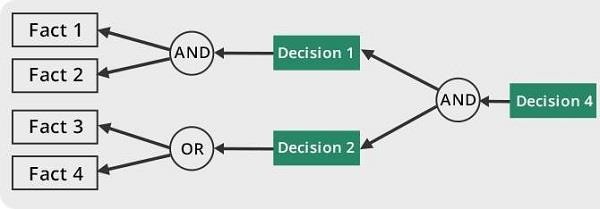


## Backward Chaining

With this strategy, an expert system finds out the answer to the question, “Why this happened?”

On the basis of what has already happened, the Inference Engine tries to find out which conditions could have happened in the past for this result. This strategy is followed for finding out cause or reason.

For example, diagnosis of blood cancer in humans.



# Representing and Using Domain Knowledge

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. **But** **how** **machines** **do** **all** **these** **things** **comes** **under** **knowledge** **representation** **and** **reasoning**. Hence we can describe Knowledge representation as following:

* Knowledge representation and reasoning (KR, KRR) is the part of Artificial intelligence which concerned with AI agents thinking and how thinking contributes to intelligent behavior of agents.
* It is responsible for representing information about the real world so that a computer can understand and can utilize this knowledge to solve the complex real world problems such as diagnosis a medical condition or communicating with humans in natural language.
* It is also a way which describes how we can represent knowledge in artificial intelligence. Knowledge representation is not just storing data into some database, but it also enables an intelligent machine to learn from that knowledge and experiences so that it can behave intelligently like a human.

## What to Represent:

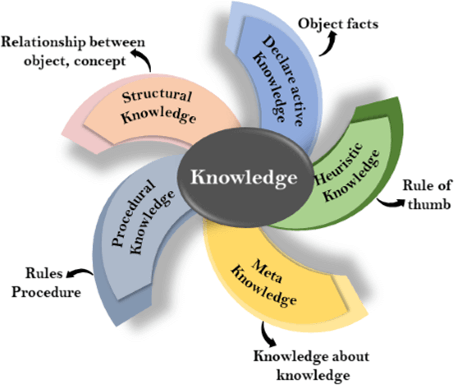
Following are the kind of knowledge which needs to be represented in AI systems:

* **Object:** All the facts about objects in our world domain. E.g., Guitars contains strings, trumpets are brass instruments.
* **Events:** Events are the actions which occur in our world.
* **Performance:** It describe behavior which involves knowledge about how to do things.
* **Meta-knowledge:** It is knowledge about what we know.
* **Facts:** Facts are the truths about the real world and what we represent.
* **Knowledge-Base:** The central component of the knowledge−based agents is the knowledge base. It is represented as KB. The Knowledgebase is a group of the Sentences (Here, sentences are used as a technical term and not identical with the English language).

**Knowledge:** Knowledge is awareness or familiarity gained by experiences of facts, data, and situations. Following are the types of knowledge in artificial intelligence:

## Types of knowledge

Following are the various types of knowledge:



## Declarative Knowledge:

* + Declarative knowledge is to know about something.
  + It includes concepts, facts, and objects.
  + It is also called descriptive knowledge and expressed in declarativesentences.
  + It is simpler than procedural language.

## Procedural Knowledge

* + It is also known as imperative knowledge.
  + Procedural knowledge is a type of knowledge which is responsible for knowing how to do something.
  + It can be directly applied to any task.
  + It includes rules, strategies, procedures, agendas, etc.
  + Procedural knowledge depends on the task on which it can be applied.

## Meta-knowledge:

* + Knowledge about the other types of knowledge is called Meta−knowledge.

## Heuristic knowledge:

* + Heuristic knowledge is representing knowledge of some experts in a field or subject.
  + Heuristic knowledge is rules of thumb based on previous experiences, awareness of approaches, and which are good to work but not guaranteed.

## Structural knowledge:

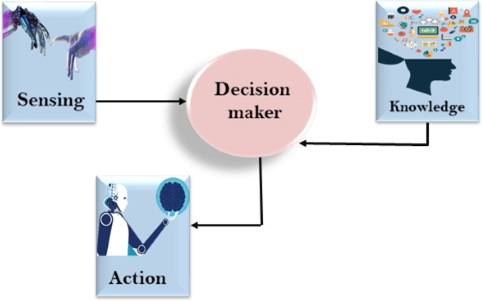
* + Structural knowledge is basic knowledge to problem−solving.
  + It describes relationships between various concepts such as kind of, part of, and grouping of something.
  + It describes the relationship that exists between concepts or objects.

# The relation between knowledge and intelligence:

Knowledge of real−worlds plays a vital role in intelligence and same for creating artificial intelligence. Knowledge plays an important role in demonstrating intelligent behavior in AI agents. An agent is only able to accurately act on some input when he has some knowledge or experience about that input.

Let's suppose if you met some person who is speaking in a language which you don't know, then how you will able to act on that. The same thing applies to the intelligent behavior of the agents.

As we can see in below diagram, there is one decision maker which act by sensing the environment and using knowledge. But if the knowledge part will not present then, it cannot display intelligent behavior.



# AI knowledge cycle:

An Artificial intelligence system has the following components for displaying intelligent behavior:

1. Perception
2. Learning
3. Knowledge Representation and Reasoning
4. Planning
5. Execution

# Approaches to knowledge representation:

There are mainly four approaches to knowledge representation, which are given below:

## Simple relational knowledge:

* + It is the simplest way of storing facts which uses the relational method, and each fact about a set of the object is set out systematically in columns.
  + This approach of knowledge representation is famous in database systems where the relationship between different entities is represented.
  + This approach has little opportunity for inference.

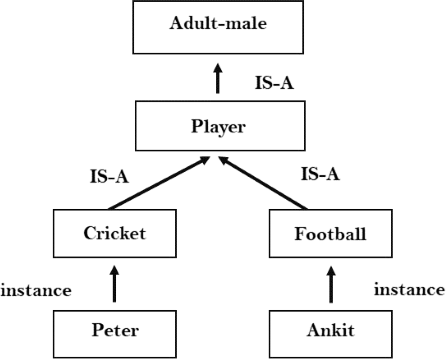
## Example: The following is the simple relational knowledge representation.

|  |  |  |
| --- | --- | --- |
| **Player** | **Weight** | **Age** |
| Player1 | 65 | 23 |
| Player2 | 58 | 18 |
| Player3 | 75 | 24 |

1. **Inheritable** **knowledge:**

* In the inheritable knowledge approach, all data must be stored into a hierarchy of classes.
* All classes should be arranged in a generalized form or a hierarchal manner.
* In this approach, we apply inheritance property.
* Elements inherit values from other members of a class.
* This approach contains inheritable knowledge which shows a relation between instance and class, and it is called instance relation.
* Every individual frame can represent the collection of attributes and its value.
* In this approach, objects and values are represented in Boxed nodes.
* We use Arrows which point from objects to their values.

## Example:

****

1. **Inferential** **knowledge:**

* Inferential knowledge approach represents knowledge in the form of formal logics.
* This approach can be used to derive more facts.
* It guaranteed correctness.

**Example:** Let's suppose there are two statements:

* 1. Marcus is a man
  2. All men are mortal Then it can represent as;

**man(Marcus)**

∀**x** **=** **man** **(x)** **>** **mortal** **(x)s**

## Procedural knowledge:

* Procedural knowledge approach uses small programs and codes which describes how to do specific things, and how to proceed.
* In this approach, one important rule is used which is **If-Then** **rule**.
* In this knowledge, we can use various coding languages such as **LISP** **language** and **Prolog** **language**.
* We can easily represent heuristic or domain−specific knowledge using this approach.
* But it is not necessary that we can represent all cases in this approach.

## Requirements for knowledge Representation system:

A good knowledge representation system must possess the following properties.

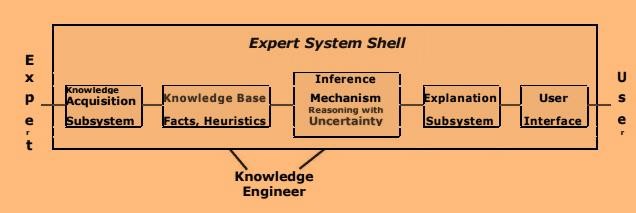
1. **Representational** **Accuracy:** KR system should have the ability to represent all kind of required knowledge.
2. **Inferential** **Adequacy:** KR system should have ability to manipulate the representational structures to produce new knowledge corresponding to existing structure.
3. **Inferential** **Efficiency:** The ability to direct the inferential knowledge mechanism into the most productive directions by storing appropriate guides.
4. **Acquisitional** **efficiency-** The ability to acquire the new knowledge easily using automatic methods.

# Expert System Shells

An Expert system shell is a software development environment. It contains the basic components of expert systems. A shell is associated with a prescribed method for building applications by configuring and instantiating these components.

## Shell components and description

The generic components of a shell : the knowledge acquisition, the knowledge Base, the reasoning, the explanation and the user interface are shown below. The knowledge base and reasoning engine are the core components.



All these components are described in the next slide.

## Knowledge Base

A store of factual and heuristic knowledge. Expert system tool provides one or more knowledge representation schemes for expressing knowledge about the application domain. Some tools use both Frames (objects) and IF−THEN rules. In PROLOG the knowledge is represented as logical statements.

## Reasoning Engine

Inference mechanisms for manipulating the symbolic information and knowledge in the knowledge base form a line of reasoning in solving a problem. The inference mechanism can range from simple modus ponens backward chaining of IF−THEN rules to Case−Based reasoning.

## Knowledge Acquisition subsystem

A subsystem to help experts in build knowledge bases. However, collecting knowledge, needed to solve problems and build the knowledge base, is the biggest bottleneck in building expert systems.

## Explanation subsystem

A subsystem that explains the system's actions. The explanation can range from how the final or intermediate solutions were arrived at justifying the need for additional data.

## User Interface

A means of communication with the user. The user interface is generally not a part of the expert system technology. It was not given much attention in the past. However, the user interface can make a critical difference in the perceived utility of an Expert system.

# Explanation

Most expert systems have explanation facilities that allow the user to ask questions − why and how it reached some conclusion.

The questions are answered by referring to the system goals, the rules being used, and existing problem solving. The rules typically reflect empirical, or "compiled" knowledge. They are codes of an expert's rules of thumb, not the expert's deeper understanding.

## Example :

Dialog with an expert system designed to give advice on car problems. System Does the car start ?

User No.

System Does the engine turn over ? User Yes.

System Do you smell gas ? User Yes

System Recommendation − Wait 5 minutes and try again. User Why ?

System I used the rule:

If not start, and engine\_turn\_over, and smell\_gas

Then recommendation is 'Wait 5 minutes and try again' User

Note : The rule gives the correct advice for a flooded car, and knows the questions to be ask to determine if the car is flooded, but it does not contain the knowledge of what a flooded car is and why waiting will help.

## Types of Explanation

There are four types of explanations commonly used in expert systems. Rule trace reports on the progress of a consultation;

* Explanation of how the system reached to the given conclusion;
* Explanation of why the system did not give any conclusion.
* Explanation of why the system is asking a question;

# Knowledge Acquisition

Knowledge acquisition includes the elicitation, collection, analysis, modeling and validation of knowledge.

## Issues in Knowledge Acquisition

The important issues in knowledge acquisition are:

* knowledge is in the head of experts
* Experts have vast amounts of knowledge
* Experts have a lot of tacit knowledge
* They do not know all that they know and use
* Tacit knowledge is hard (impossible) to describe
* Experts are very busy and valuable people
* One expert does not know everything
* Knowledge has a "shelf life"

## Techniques for Knowledge Acquisition

The techniques for acquiring, analyzing and modeling knowledge are : Protocol−generation techniques, Protocol analysis techniques, Hiera hy−generation techniques, Matrix−based techniques, Sorting techniques, Limited−information and constrained−processing tasks, Diagram−based techniques. Each of these are briefly stated in next few slides.

## Protocol-generation techniques

Include many types of interviews (unstructured, semi−structured and structured), reporting and observational techniques.

## Protocol analysis techniques

Used with transcripts of interviews or text−based information to identify basic knowledge objects within a protocol, such as goals, decisions, relationships and attributes. These act as a bridge between the use of protocol−based techniques and knowledge modeling techniques.

## Hiera hy-generation techniques

Involve creation, reviewing and modification of hieratical knowledge.

Hiera hy−generation techniques, such as laddering, are used to build taxonomies or other hiera hical structures such as goal trees and decision networks. The Ladders are of various forms like concept ladder, attribute ladder, composition ladders**.**

## Matrix-based techniques

Involve the construction and filling−in a 2−D matrix (grid, table), indicating such things, as may be, for example, between concepts and properties (attributes and values) or between problems and solutions or between tasks and resources, etc. The elements within the matrix can contain: symbols (ticks, crosses, question marks ) , colors , numbers , text.

## Sorting techniques

Used for capturing the way people compare and order concepts; it may reveal knowledge about classes, properties and priorities.

## Limited-information and constrained-processing tasks

Techniques that either limit the time and/or information available to the expert when performing tasks. For example, a twenty−questions technique provides an efficient way of accessing the key information in a domain in a prioritized order.

## Diagram-based techniques

Include generation and use of concept maps, state transition networks, event diagrams and process maps. These are particularly important in capturing the "what, how, when, who and why" of tasks and events.