**Velammal College of Engineering and Technology, Madurai**

**Department of Information Technology**

**2015-2016 Even Semester**

**Course Code-Title : CS6659 ARTIFICIAL INTELLIGENCE Degree/Year/Semester: B.TECH-IT/**III/VI

**Name of the Instructor: Ms. S.Nirmala**

**UNIT V**

**EXPERT SYSTEMS**

Expert systems - Architecture of expert systems, Roles of expert systems - Knowledge Acquisition –Meta knowledge, Heuristics. Typical expert systems - MYCIN, DART, XCON, Expert systems shells.

**Knowledge-based expert systems or simply expert systems**

1. An expert system is software that attempts to reproduce the performance of one or more human experts, most commonly in a specific problem domain. It represents the expertise knowledge as data or rules within the computer and it can be called upon when needed to solve problems
2. Expert system developed via specialized software tools called shells .Shells come equipped with an inference mechanism such as backward chaining, forward chaining and both. It may or may not have learning components and they are tested by being placed in the same real world problem solving situation

Examples:

* Diagnostic applications, servicing:
	+ People
	+ Machinery
* Play chess
* Make financial planning decisions
* Configure computers
* Monitor real time systems
* Underwrite insurance policies

**Architecture of Expert systems:**

****

**Roles of Expert Systems:**

* **Knowledge base -** a declarative representation of the expertise, often in IF THEN rules
* **Working storage -** the data which is specific to a problem being solved
* **Inference engine -** the code at the core of the system . Derives recommendations from the knowledge base and problem-specific data in working storage
* **User interface -** the code that controls the dialog between the user and the system
* **Domain expert –** currently experts solving the problems the system is intended to solve
* **Knowledge engineer -** encodes the expert's knowledge in a declarative form that can be used by the expert system
* **User -** will be consulting with the system to get advice which would have been provided by the expert
* **System engineer -** the individual who builds the user interface, designs the declarative format of the knowledge base, and implements the inference engine
* **Shell -** a piece of software which contains:
1. The user interface
2. A format for declarative knowledge in the knowledge base
3. An inference engine

Major advantage of a customized shell is the format of the knowledge base can be designed to facilitate the knowledge engineering process. Knowledge engineer and the system engineer might be the same person depending on the size of the project

**Knowledge engineering process:**

1. The coding of the expertise into the declarative rule format can be a difficult and tedious task
2. The semantic gap between the expert's representation of the knowledge and the representation in the knowledge base should be minimized

**Expert System Features**

* **Goal driven reasoning or backward chaining** - an inference technique which uses IF THEN rules to repetitively break a goal into smaller sub-goals which are easier to prove
* **Coping with uncertainty** - the ability of the system to reason with rules and data which are not precisely known
* **Data driven reasoning or forward chaining** - an inference technique which uses IF THEN rules to deduce a problem solution from initial data
* **Data representation** - the way in which the problem specific data in the system is stored and accessed
* **User interface** - that portion of the code which creates an easy to use system
* **Explanations** - the ability of the system to explain the reasoning process that it used to reach a recommendation.

**Goal-Driven Reasoning**

* To pick the best choice from many enumerated possibilities
* The knowledge is structured in rules
* The rule breaks the problem into sub-problems

**Difference between forward and backward chaining:**



**Example** - This rules identifies birds:

IF family is albatross and color is white

  THEN bird is Laysan albatross

 IF family is albatross and color is dark

  THEN bird is black footed albatross

The following rule is one that satisfies the family sub-goal:

IF order is tubenose and size large and wings long narrow

  THEN family is albatross

**Uncertainty**

* Final answer is not known with complete certainty
	+ The expert's rules might be vague
	+ The user might be unsure of answers to questions
* **Example** - medical diagnostic system: multiple possible diagnoses
* One of the simplest schemes: to associate a numeric value with each piece of information

**Data Driven Reasoning**

* Problem of enumerating all of the possible answers before hand
* Example - configuration problems
* Keeps track of the current state, looks for rules moving the state closer to a final solution

**Difference between data driven and goal driven systems**

* Data driven system: the system must be initially populated with data
* Goal driven system: gathers data as it needs it

**Example**: A system to layout living room furniture (unplaced pieces of furniture):

 IF unplaced tv and couch on wall(X) and wall(Y) opposite wall(X)

  THEN place tv on wall(Y).

**Explanations**

* Ability to explain themselves
* System knows which rules were used during the inference process
* Can be very dramatic: dark colored and an albatross => the bird was a black footed albatross
* At other times, relatively useless to the user
	+ Example - car diagnostic system: no rules

**Advantages:**

* Consistent answers for repetitive decisions, processes and tasks
* Holds and maintains significant levels of information
* Encourages organizations to clarify the logic of their decision-making
* Never "forgets" to ask a question, as a human might

**Disadvantages:**

* Lacks common sense
* Cannot make creative responses as human expert
* Domain experts not always able to explain their logic and reasoning
* Errors may occur in the knowledge base
* Cannot adapt to changing environments

**Knowledge Base**

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

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 expert system 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 IT-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.

**Interface Engine**

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

In case of knowledge-based expert systems, the Interface Engine acquires and manipulates the knowledge from the knowledge base to arrive at a particular solution.

In case of rule based Expert System, 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.

**User Interface**

User interface provides interaction between user of the expert system and the expert system itself. It is generally Natural Language Processing so as to be used by the user who is well-versed in the task domain. The user of the expert system need not be necessarily an expert in Artificial Intelligence. It explains how the expert system has arrived at a particular recommendation. The explanation may appear in the following forms −

* Natural language displayed on screen.
* Verbal narrations in natural language.
* Listing of rule numbers displayed on the screen.

The user interface makes it easy to trace the credibility of the deductions. Requirements of efficient expert system user interface

* It should help users to accomplish their goals in shortest possible way.
* It should be designed to work for user’s existing or desired work practices.
* Its technology should be adaptable to user’s requirements; not the other way round.
* It should make efficient use of user input.