Unit-V

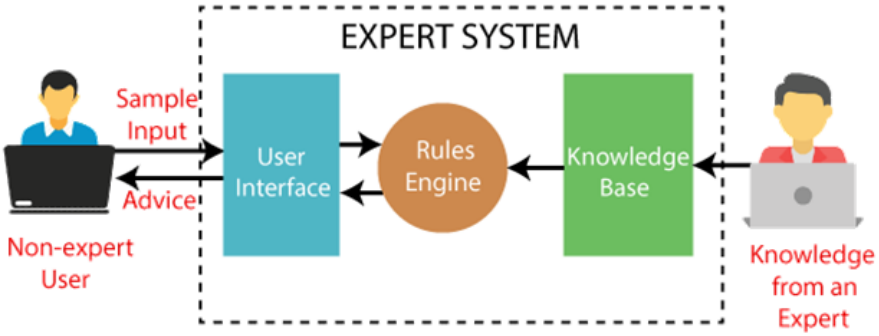
What is an Expert System?

An expert system is a computer program that is designed to solve complex problems and to provide decision-making ability like a human expert. It performs this by extracting knowledge from its knowledge base using the reasoning and inference rules according to the user queries.

The expert system is a part of AI, and the first ES was developed in the year 1970, which was the first successful approach of artificial intelligence. It solves the most complex issue as an expert by extracting the knowledge stored in its knowledge base. The system helps in decision making for complex problems using **both facts and heuristics like a human expert**. It is called so because it contains the expert knowledge of a specific domain and can solve any complex problem of that particular domain. These systems are designed for a specific domain, such as **medicine, science,** etc.

The performance of an expert system is based on the expert's knowledge stored in its knowledge base. The more knowledge stored in the KB, the more that system improves its performance. One of the common examples of an ES is a suggestion of spelling errors while typing in the Google search box.

Below is the block diagram that represents the working of an expert system:



Below are some popular examples of the Expert System:

- **DENDRAL:** It was an artificial intelligence project that was made as a chemical analysis expert system. It was used in organic chemistry to detect unknown organic molecules with the help of their mass spectra and knowledge base of chemistry.
- **MYCIN:** It was one of the earliest backward chaining expert systems that was designed to find the bacteria causing infections like bacteraemia and meningitis. It was also used for the recommendation of antibiotics and the diagnosis of blood clotting diseases.
- **PXDES:** It is an expert system that is used to determine the type and level of lung cancer. To determine the disease, it takes a picture from the upper body, which looks like the shadow. This shadow identifies the type and degree of harm.

- **CaDeT:** The CaDet expert system is a diagnostic support system that can detect cancer at early stages.

Characteristics (Role) of Expert System

- **High Performance:** The expert system provides high performance for solving any type of complex problem of a specific domain with high efficiency and accuracy.
- **Understandable:** It responds in a way that can be easily understandable by the user. It can take input in human language and provides the output in the same way.
- **Reliable:** It is much reliable for generating an efficient and accurate output.
- **Highly responsive:** ES provides the result for any complex query within a very short period of time.

Steps to Develop an Expert System:

Step1: Identification: Determining the characteristics of the problem.

Step2: Conceptualization: Finding the concept to produce the solution.

Step3: Formalization: Designing structures to organize the knowledge.

Step4: Implementation: Formulating rules which embody the knowledge.

Step5. Testing (Validation, Verification and Maintenance).

A knowledge engineer is an AI specialist, perhaps a computer scientist or programmer, who is skilled in the 'Art' of developing expert systems, unlike other engineering disciplines, there are no generally accepted criteria to determine exactly who is knowledge engineer; the field is much too new. You don't need a degree in "knowledge engineering" to call yourself a knowledge engineer; infact, nearly everyone who has ever contributed to the technical side of the expert system development process could be considered a knowledge engineer.

A domain expert is an individual who has significant expertise in the domain of the expert system being developed. It is not critical that the domain expert understand AI or expert systems; that is one of the functions of the knowledge engineer.

The knowledge engineer and the domain expert usually work very closely together for long periods of time throughout the several stages of the development process.

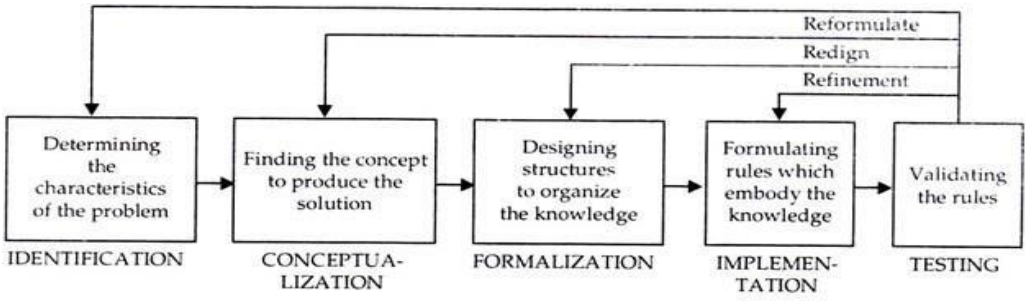


Fig. 12.8. *The five stages of expert system development.*

Stage # 1. Identification:

Before we can begin to develop an expert system, it is important to describe, with as much precision as possible, the problem which the system is intended to solve. It is not enough simply to feel that an expert system would be helpful in a certain situation; we must determine the exact nature of the problem and state the precise goals which indicate exactly how the expert system is expected to contribute to the solution.

To begin, the knowledge engineer, who may be unfamiliar with this particular domain, consults manuals and training guides to gain some familiarity with the subject. Then the domain expert describes several typical problem states. The knowledge engineer attempts to extract fundamental concepts from the similar cases in order to develop a more general idea of the purpose of the expert system.

After the domain expert describes several cases, the knowledge engineer develops a 'first-pass' problem description. Typically, the domain expert may feel that the description does not entirely represent the problem. The domain expert then suggests changes to the description and provides the knowledge engineer with additional examples to illustrate further the problem's fine points.

Next, the knowledge engineer revises the description, and the domain expert suggests further changes. This process is repeated until the domain expert is satisfied that the knowledge engineer understands the problems and until both are satisfied that the description adequately portrays the problem which the expert system is expected to solve.

This 'iterative' procedure (Fig. 12.9) is typical of the entire expert-system development process. The results are evaluated at each stage of the process and compared to the expectations. If the results do not meet the expectations, adjustments are made to that stage

of the process, and the new results are evaluated. The process continues until satisfactory results are achieved.

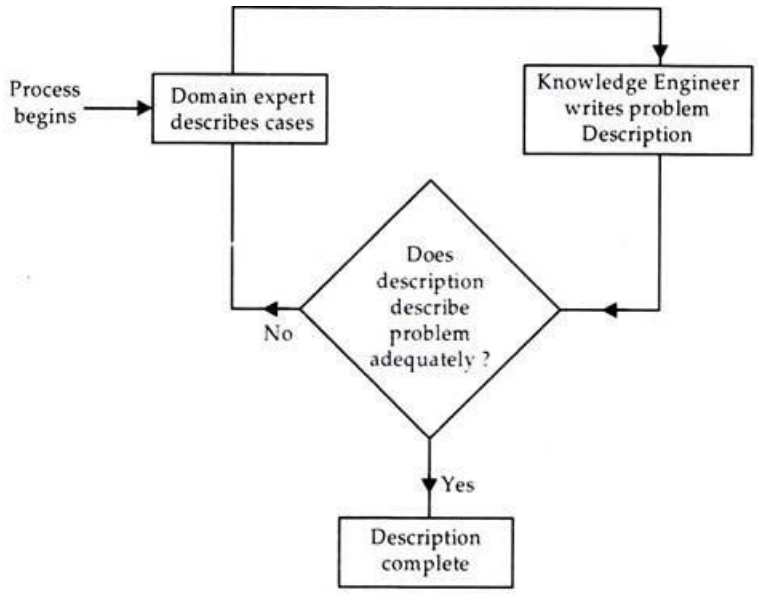


Fig. 12.9. *The iterative process of identifying the problem which the expert system is to solve.*

It is also important to identify our resources. Who is to participate in the development process? Does a single domain expert possess all the necessary expertise, or is the domain knowledge distributed over several people in an organisation? Can a single knowledge engineer develop the system in a timely fashion, or is it necessary to provide additional technical assistance?

Domain experts are not the only resources which must be identified. It is unusual for all domain knowledge to be embodied in human experts; therefore, more tangible sources of information, such as reference books and manuals, are usually identical and located.

Stage # 2. Conceptualisation:

Once it has been identified for the problem an expert system is to solve, the next stage involves analysing the problem further to ensure that its specifics, as well as generalities, are understood.

In the conceptualisation stage, the knowledge engineer frequently creates a diagram of the problem to depict graphically the relationships between the objects and processes in the problem domain. It is often helpful at this stage to divide the problem into a series of sub-

problems and to diagram both the relationships among the pieces of each sub-problem and the relationships among the various sub-problems.

As in the identification stage, the conceptualisation stage involves a circular procedure of iteration and reiteration between the knowledge engineer and the domain expert. When both agree that the key concepts-and the relationships among them-have been adequately conceptualised, this stage is complete.

Not only is each stage in the expert system development process circular, the relationships among the stages may be circular as well. Since each stage of the development process adds a level of detail to the previous stage, any stage may expose a weakness in a previous stage.

For example, a problem with the description generated in the identification stage may be discovered during conceptualisation. A key element of the description may have been omitted, or perhaps a goal was stated incorrectly. If this occurs, a brief return to the identification stage is required to increase the accuracy of the description (Fig. 12.10). A similar process can occur in any stage of development.

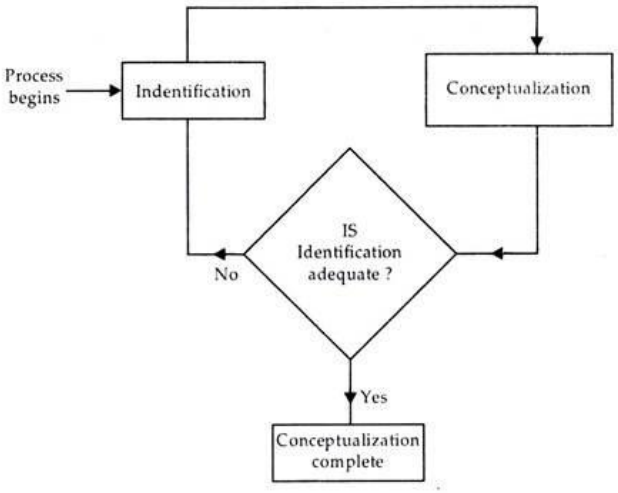


Fig. 12.10. *The iterative relationship between the indentification and conceptualisation stages of expert system development.*

Stage # 3. Formalisation (Designing):

In the preceding stages, no effort has been made to relate the domain problem to the artificial intelligence technology which may solve it. During the identification and formalization stages, the focus is entirely on understanding the problem. Now, during the

formalization stage, the problem is connected to its proposed solution, an expert system is supplied by analyzing the relationships depicted in the conceptualization stage. The knowledge engineer begins to select the techniques which are appropriate for developing this particular expert system.

During formalization, it is important that the knowledge engineer be familiar with the following:

1. The various techniques of knowledge representation and intelligent search techniques used in expert systems.

2. The expert system tools which can greatly expedite the development process.

3. Other expert systems which may solve similar problems and thus may be adaptable to problem at hand.

Often it is desirable to select a single development technique or tool which can be used throughout all segments of the expert system. However, the knowledge engineer may determine that no particular technique is appropriate for the entire expert system, making it necessary to use different techniques for different sub-problems. Once it has been determined which technique(s) will be used the knowledge engineer starts to develop a formal specification which can used to develop a prototype expert system.

In the case of a rule-based system, for example, the knowledge engineer develops a set of rules designed to represent the knowledge communicated by the domain expert. This is a critical part of the development process, requiring, great skill on the part-of the knowledge engineer. Many domain experts can explain what they do but not why; therefore, one of the knowledge engineer's primary responsibilities is to analyse example situations and filter in from those examples a set of rules which describe the domain expert's knowledge.

The formalisation process is often the most interactive stage of expert system development, as well as the most time consuming. The knowledge engineer must develop a set of rules and ask the domain expert if those rules adequately represent the expert's knowledge. The domain expert reviews the rules proposed by the knowledge engineer and

suggests changes, which are then incorporated into the knowledge base by the knowledge engineer.

As in the other development stages, this process also is iterative: the rule review is repeated and the rules are refined continually until the results are satisfactory. It is not unusual for the formalisation process of a complex expert system to last for several years. (Fig. 12.10)

Stage # 4. Implementation:

During the implementation stage the formalised concepts are programmed into the computer which has been chosen for system development, using the predetermined techniques and tools to implement a 'first-pass' (prototype) of the expert system.

Theoretically, if the methods of the previous stages have been followed with diligence and care, the implementation of the prototype should proceed smoothly. In practice, the development of an expert system may be as much an art as it is a science, because following all the rules does not guarantee that the system will work the first time it is implemented. In fact, experience suggests the opposite. Many scientists actually consider the prototype to be a 'throw-away' system, useful for evaluating progress but hardly a usable expert system.

If the prototype works at all, the knowledge engineer may be able to determine if the techniques chosen to implement the expert system were the appropriate ones. On the other hand, the knowledge engineer may discover that the chosen techniques simply cannot be implemented. It may not be possible, for example, to integrate the knowledge representation techniques selected for different sub-problems. At that point, the concepts may have to be re-formalised, or it even may be necessary to create new development tools to implement the system efficiently. The implementation stage is illustrated in Fig. 12.11.

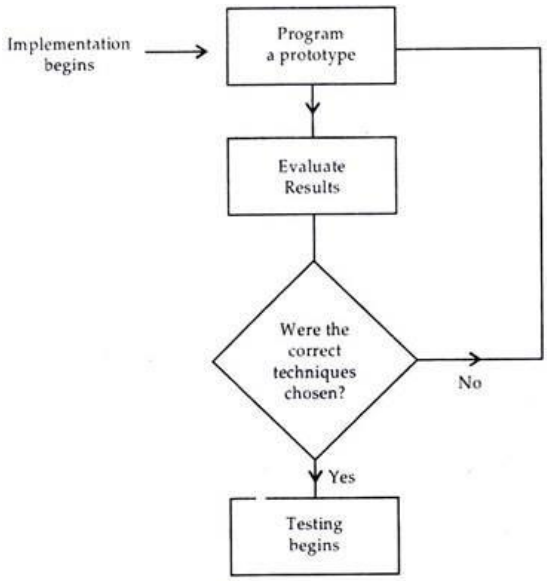


Fig. 12.11. *The implementation stage of expert system development.*

Once the prototype system has been refined sufficiently to allow it to be executed, the expert system is ready to be tested thoroughly to ensure that it expertise's correctly.

Stage # 5. Testing (Validation, Verification and Maintenance):

The chance of prototype expert system executing flawlessly the first time it is tested are so slim as to be virtually non-existent. A knowledge engineer does not expect the testing process to verify that the system has been constructed entirely correctly. Rather, testing provides an opportunity to identify the weaknesses in the structure and implementation of the system and to make the appropriate corrections.

Depending on the types of problems encountered, the testing procedure may indicate that the system was implemented incorrectly, or perhaps that the rules were implemented correctly but were poorly or incompletely formulated. Results from the tests are used as 'feedback' to return to a previous stage and adjust the performance of the system.

Once the system has proven to be capable of correctly solving straight-forward problems, the domain expert suggests complex problems which typically would require a great deal of human expertise. These more demanding tests should uncover more serious flaws and provide ample opportunity to 'fine tune' the system even further.

Ultimately, an expert system is judged to be entirely successful only when it operates at the level of a human expert. The testing process is not complete until it indicates that the

solutions suggested by the expert system are consistently as valid as those provided by a human domain expert.

How does an Expert System look like?

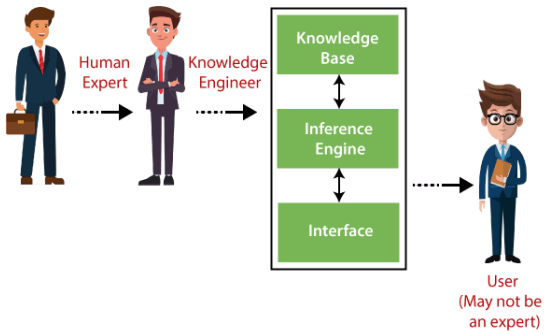
The ultimate test of an expert system is how well it performs, not in a development laboratory, but in 'real life' situations. A completed system not only must demonstrate consistently that it can deliver the required expertise, but it also must be easy to use so that the embedded expertise can be extracted easily. Therefore, developers have gone to the extent that expert systems are among the easiest to use of all sophisticated computer programs.

Expert systems typically are interactive; that is, we use a question-and-answer technique to communicate with them. Generally, we access an expert system by sitting at a computer terminal. We type the description of our problem on the computer keyboard, and the expert system displays questions or selections lists on the screen to prompt us to enter additional information. The 'give-and-take' between us and the computer continues until the system is able to reach a conclusion which is displayed on the screen.

Components of Expert System

An expert system mainly consists of three components:

- **User Interface**
- **Inference Engine**
- **Knowledge Base**



1. User Interface

With the help of a user interface, the expert system interacts with the user, takes queries as an input in a readable format, and passes it to the inference engine. After getting the response from the inference engine, it displays the output to the user. In other words, **it is an interface that helps a non-expert user to communicate with the expert system to find a solution**.

2. Inference Engine (Rules of Engine)

- The inference engine is known as the brain of the expert system as it is the main processing unit of the system. It applies inference rules to the knowledge base to derive a conclusion or deduce new information. It helps in deriving an error-free solution of queries asked by the user.
- With the help of an inference engine, the system extracts the knowledge from the knowledge base.
- There are two types of inference engine:
- **Deterministic Inference engine:** The conclusions drawn from this type of inference engine are assumed to be true. It is based on **facts** and **rules**.
- **Probabilistic Inference engine:** This type of inference engine contains uncertainty in conclusions, and based on the probability.

Inference engine uses the below modes to derive the solutions:

- **Forward Chaining:** It starts from the known facts and rules, and applies the inference rules to add their conclusion to the known facts.
- **Backward Chaining:** It is a backward reasoning method that starts from the goal and works backward to prove the known facts.

3. Knowledge Base

- The knowledgebase is a type of storage that stores knowledge acquired from the different experts of the particular domain. It is considered as big storage of knowledge. The more the knowledge base, the more precise will be the Expert System.
- It is similar to a database that contains information and rules of a particular domain or subject.
- One can also view the knowledge base as collections of objects and their attributes. Such as a Lion is an object and its attributes are it is a mammal, it is not a domestic animal, etc.

Components of Knowledge Base

- **Factual Knowledge:** The knowledge which is based on facts and accepted by knowledge engineers comes under factual knowledge.
- **Heuristic Knowledge:** This knowledge is based on practice, the ability to guess, evaluation, and experiences.

Knowledge Representation: It is used to formalize the knowledge stored in the knowledge base using the If-else rules.

Knowledge Acquisitions: It is the process of extracting, organizing, and structuring the domain knowledge, specifying the rules to acquire the knowledge from various experts, and store that knowledge into the knowledge base.

Development of Expert System

Here, we will explain the working of an expert system by taking an example of MYCIN ES. Below are some steps to build an MYCIN:

- Firstly, ES should be fed with expert knowledge. In the case of MYCIN, human experts specialized in the medical field of bacterial infection, provide information about the causes, symptoms, and other knowledge in that domain.
- The KB of the MYCIN is updated successfully. In order to test it, the doctor provides a new problem to it. The problem is to identify the presence of the bacteria by inputting the details of a patient, including the symptoms, current condition, and medical history.
- The ES will need a questionnaire to be filled by the patient to know the general information about the patient, such as gender, age, etc.
- Now the system has collected all the information, so it will find the solution for the problem by applying if-then rules using the inference engine and using the facts stored within the KB.
- In the end, it will provide a response to the patient by using the user interface.

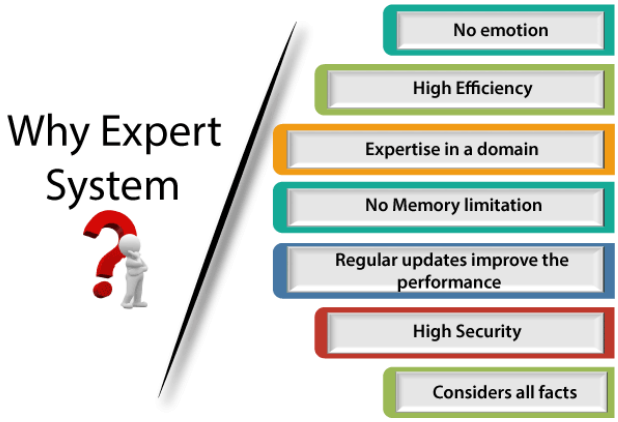
Participants in the development of Expert System

There are three primary participants in the building of Expert System:

1. **Expert:** The success of an ES much depends on the knowledge provided by human experts. These experts are those persons who are specialized in that specific domain.
2. **Knowledge Engineer:** Knowledge engineer is the person who gathers the knowledge from the domain experts and then codifies that knowledge to the system according to the formalism.

3. **End-User:** This is a particular person or a group of people who may not be experts, and working on the expert system needs the solution or advice for his queries, which are complex.

Why Expert System?



Before using any technology, we must have an idea about why to use that technology and hence the same for the ES. Although we have human experts in every field, then what is the need to develop a computer-based system. So below are the points that are describing the need of the ES:

1. **No memory Limitations:** It can store as much data as required and can memorize it at the time of its application. But for human experts, there are some limitations to memorize all things at every time.
2. **High Efficiency:** If the knowledge base is updated with the correct knowledge, then it provides a highly efficient output, which may not be possible for a human.
3. **Expertise in a domain:** There are lots of human experts in each domain, and they all have different skills, different experiences, and different skills, so it is not easy to get a final output for the query. But if we put the knowledge gained from human experts into the expert system, then it provides an efficient output by mixing all the facts and knowledge
4. **Not affected by emotions:** These systems are not affected by human emotions such as fatigue, anger, depression, anxiety, etc.. Hence the performance remains constant.
5. **High security:** These systems provide high security to resolve any query.
6. **Considers all the facts:** To respond to any query, it checks and considers all the available facts and provides the result accordingly. But it is possible that a human expert may not consider some facts due to any reason.

7. **Regular updates improve the performance:** If there is an issue in the result provided by the expert systems, we can improve the performance of the system by updating the knowledge base.

Capabilities of the Expert System

Below are some capabilities of an Expert System:

- **Advising:** It is capable of advising the human being for the query of any domain from the particular ES.
- **Provide decision-making capabilities:** It provides the capability of decision making in any domain, such as for making any financial decision, decisions in medical science, etc.
- **Demonstrate a device:** It is capable of demonstrating any new products such as its features, specifications, how to use that product, etc.
- **Problem-solving:** It has problem-solving capabilities.
- **Explaining a problem:** It is also capable of providing a detailed description of an input problem.
- **Interpreting the input:** It is capable of interpreting the input given by the user.
- **Predicting results:** It can be used for the prediction of a result.
- **Diagnosis:** An ES designed for the medical field is capable of diagnosing a disease without using multiple components as it already contains various inbuilt medical tools.

Advantages of Expert System

- These systems are highly reproducible.
- They can be used for risky places where the human presence is not safe.
- Error possibilities are less if the KB contains correct knowledge.
- The performance of these systems remains steady as it is not affected by emotions, tension, or fatigue.
- They provide a very high speed to respond to a particular query.

Limitations of Expert System

- The response of the expert system may get wrong if the knowledge base contains the wrong information.
- Like a human being, it cannot produce a creative output for different scenarios.
- Its maintenance and development costs are very high.

- Knowledge acquisition for designing is much difficult.
- For each domain, we require a specific ES, which is one of the big limitations.
- It cannot learn from itself and hence requires manual updates.

Applications of Expert System

- **In designing and manufacturing domain**

  It can be broadly used for designing and manufacturing physical devices such as camera lenses and automobiles.

- **In the knowledge domain**

  These systems are primarily used for publishing the relevant knowledge to the users. The two popular ES used for this domain is an advisor and a tax advisor.

- **In the finance domain**

  In the finance industries, it is used to detect any type of possible fraud, suspicious activity, and advise bankers that if they should provide loans for business or not.

- **In the diagnosis and troubleshooting of devices**

  In medical diagnosis, the ES system is used, and it was the first area where these systems were used.

- **Planning and Scheduling**

  The expert systems can also be used for planning and scheduling some particular tasks for achieving the goal of that task.