ARAB REPUBLIC OF EGYPT

THE INSTITUTE OF NATIONAL PLANNING

Memo No. (1571)

EXPERT SYSTEMS : AN OVERVIEW

By

Dr. Abdalla El Daoushy

Sept. 1993

CAIRO
SALAH SALEM St-NASR CITY

ABSTRACT :

At present, the most important application areas of Artificial Intelligence (AI) would be centered on knowledge engineering and ,more particularly, on Expert Systems (ESs).

Today, ESs are used in business, science, engineering, manufacturing, and many other fields.

This memo introduces just the fundamental principles of ESs and their advantages in real world applications. Knowledge representation, searching methods, inference mechanism are also included.

The memo. also mentions the development of ES shells and/or programming languages which minimize the effort needed for constructing a knowledge representation system and an inference engine for a particular applications.

At the end, it ends with the future of ESs in general.

CONTENTS

Introduction to Artificial Intelligence (AI):

The intelligence of a human being is the ability to understand, think, and reason on the available knowledge. It also includes the ability to update and upgrade this knowledge.

Generally, AI is that area of a computer science concerned with trying to make computers as smart as human beings. That is to make computers able to speak, listen, see, think, reason, understand, and do manual jobs.

But as "intelligence" is uncertain word, so is AI is not a well-defined field. AI is often means advanced software engineering, sophisticated software techniques for hard problems that cannot be solved in any easy way.

Another thing, AI often means the non-numeric methods of solving problems. Non-numeric methods are often "common sense" methods, not necessarily the best ones.

So, AI programs like people are usually not perfect, and even makes mistakes.

The AI domain is usually divided into several fields :

1, **Natural Languages Processing** : Natural languages refer to human being languages, such as english, arabic,etc. It will be nice if we can communicate with computers using our languages, and not the conventional computer languages. The AI field concerned with communicating with computers in our languages is called natural language processing.

2, **Voice Synthesis and Recognition** : The AI field concerned with communicating with computers directly by human voice is known as voice synthesis and recognition field. General speaking, voice recognition means making computers able to listen (understanding our speech).

3, **Computer Vision** : This field is devoted to process of emulating the human sight. It looks for techniques to analyze and interpret visual information in order to replace the human sight.

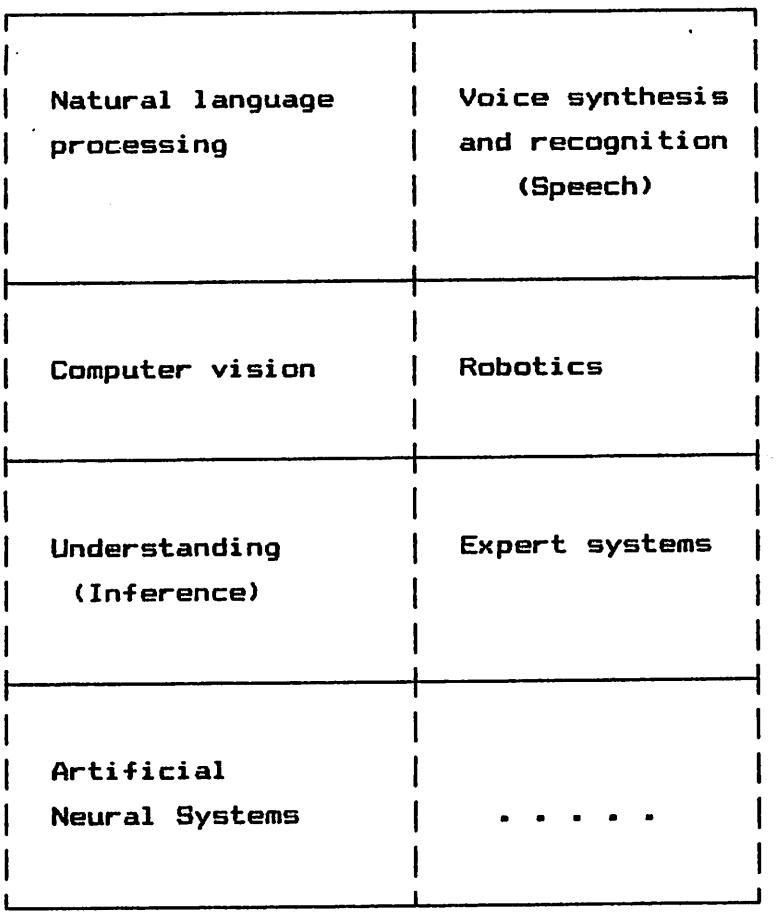
4, **Robotics** : The robotics is the field which concern with simulating both the human peripherals and the part of the human brain that controls these peripherals of the human body.

5, **Understanding (Inference)** : This field is mainly devoted to the researches leading the computer to remember complicated interrelated facts, and draw conclusions from them in a way like the emulation of the human being understanding and thinking process.

6, **Expert Systems (ESs)** : ES are AI programs that allow computers to simulate a human expert in a subjected knowledge area. For example, MYCIN is an ES which was designed to deal with problems in the treatment and diagnosis of infectious diseases.

7, **Artificial Neural Systems (ANS)** : ANS is a new development in computer programming arose in 1980's, based on how the human's brain processes information. ANS has had remarkable success in providing real-time response to complex pattern recognition problems. In one case, ANS running on an ordinary micro-computer obtained a very good solution to the "traveling salesperson problem" in 0.1 second compared to the optimum solution that required an hour of CPU time on a main-frame [6].

In summary, the following shows some areas of AI fields :



EXPERT SYSTEMS :

Introduction :

One area of AI research which is quite advanced is that of ESs. Indeed, ESs is one of the few areas of AI which has moved out of the research laboratory and moved into the real world and is beginning to realize its potential in industrial and commercial applications.

An ES is an AI computer program which makes the computers act as a human expert within one particular domain of knowledge.

An ES is an AI computer program which makes extensive use of a knowledge base to solve problems at a human expert level.

The knowledge of one or more experts in a specific area (domain) is packaged into a knowledge base that can be processed by an ES. Even if the domain is a complex one, ES tries to achieve the same result that the expert could achieve.

Ideally, the ES can also learn from its mistakes and gain experience from its successes and failures. The ES should also be able to explain the reasoning behind the way in which it has arrived at a particular conclusion.

The ES has ,generally, 2 main parts :

1, A knowledge base, and
2, An Inference engine.

The Knowledge base is that part which represents the knowledge of the expert in his specific domain. A knowledge base usually contain facts and rules.

The inference engine is that part which represents the way of thinking and searching through the knowledge base of an ES. The inference engine searches through the knowledge base using either of the forward chaining or the backward chaining. That is, searches from the initial conditions to the goal or from the goal to the initial conditions.

ADVANTAGES OF HAVING AN ES :

There are many benefits that can be gained from the introduction of ESs technology within a particular area. Some of the more obvious benefits include :

1. **Increased availability :** The use of ESs increases the availability of expertise. Recall that the computers can ,now, be available anywhere and anytime.

2. **Reduced cost :** ESs can be much more cost effective (i.e.,much cheaper) than hiring the services of a real human expert.

3. **Reduced danger :** ESs can be used in environments that might be dangerous for a human.

4. **Performance :** The expertise is permanent, unlike human experts; they are scarce, their service are expensive, they are usually very much in demand and are very busy, and they are mortal (cannot live for ever). ESs do not suffer from these drawbacks.

5. **Multiple Experts :** The expertise are gathered from several experts. Hence, the knowledge of multiple experts can be made available to work simultaneously and continuously on a problem anytime.

6. **High productivity :** ESs are usually faster and never get tired. This means that the productivity of ES is higher than human expert.

7. **Saving the user's money and time :** ESs can save user's money and time by avoiding costly mistakes and bad decisions that can be made by a non-experienced human professionals. Consulting ES is much cheaper and faster than consulting a human expert. This is a direct result of reducing cost per site and increasing productivity.

8. **Updating and qualifying :** The knowledge of human experts must be put into an explicit form for entering in the computer. Because the knowledge is then explicitly known instead of being implicit in the expert's mind, it can be examined for correctness, consistency and completeness. The knowledge may then have to be adjusted or reexamined which improves the quality of the knowledge.

9. **Explanation :** The ES can explicitly explain in detail the reasoning that led to a conclusion. A human expert may be too tired, unwilling or unable to do this all the time. This increases the confidence that the correct decision is made.

10. **Fast response :** Fast or real-time response may be necessary for some applications. Depending on the software and hardware used, an ES may respond faster and be more available than a human expert. Some emergency situations may require responses faster than a human and so a real-time ES is a good choice.

USE OF ESs :

ESs are gradually finding their way into many areas of modern life. Indeed, only certain types of applications are suited to implementation in ES form.

How do we determine whether or not a particular area is suitable for ESs treatment?. General speaking, most of problems which ,mainly, need practical experience in order to be solved. Anyhow, the following rules give an indication of the sort of criteria which must be satisfied in order for a particular application area to benefit from ES treatment :

- The problem under consideration should be able to be reduced into a series of rules rather than mathematical formulas or equations. In particular, ES treatment is not applicable if the problem involves a large number of calculations.

- The problem under study should be well understood so that well defined rules can be formulated to represent human expertise.

- The field under study should not comprise problems which take too short time (< 0.5 an hour) or too long time (> one week).

- There should be general agreements among recognized experts in the field (no use if all experts have different ideas or theories --- in such a case whose knowledge would be computerized).

- The number of rules necessary to describe the system should be sufficiently large to warrant the development of an ES.

- There should be one or more experts who are agreeable to their involvement in the project.

ESs APPLICATIONS :

ESs have applied to several fields of knowledge to solve specific problems. Some have been designed as research tools, while others fulfill important business, science, engineering and military functions.

- **Business ESs :** loan evaluation ESs, tax advising ESs, and business management ESs are examples of the business ESs. Banks have standard criteria for evaluating loan applications. In addition, Expert loan-evaluation bankers have their own experience. Both the standard criteria and the expert knowledge can be gathered in a knowledge base to be used in a loan evaluation ES.

Taxes are a vital part of the financial strategies of both individuals and companies. Tax advising ES which helps individuals and companies make good tax decisions are now amiable.

Business management ESs being developed. These ESs perform a variety of financial analysis and make recommendations to the management. There are also other business ESs which are used to evaluate business plans.

- **Science and Engineering ESs :** Engineers are using knowledge based Computer-Aided-Design (CAD) systems to create new products. Ford organization used ES to diagnose machine tool problems. In the computer's system configuration field, one of the most common ESs is XCON of digital equipment corporation. The configuration of a computer system means that when a customer places an order, all the right parts (software, hardware, and documentation) should be supplied. An ES which is designed to aid medical practitioners in the diagnosis of specific pain would have a knowledge base which is contained facts and rules about the likely causes of such pain, and the particular symptoms related to each of these causes. MYCIN of MIT is an example of such ESs.

- **Military ESs :** The military people see intelligent software as a way to improve performance and gain competitive advantages. Several military ESs for the air forces, army, and navy were developed.

. **Air force ESs :** Air crew ES helps pilots to do their jobs more efficiency. A program called "Pilot Associate" helps pilots plane in their missions. It also monitors the air craft subsystem for problems and informs the pilot. Some air forces are already using ESs to perform maintenance on combat air crafts. Most of these ESs run on PCs and are designed to diagnose problems in the air craft engine and avionics.

. **Army ESs :** In the army, some ESs which help in training commanders, upgrading the way a weapon works, maintaining and repairing of equipment are built. It should be noted that the use of ESs to improve the ability to maintain and repair equipment, should lead to reduce the maintenance costs and increasing the equipment preparedness.

. **Navy ESs :** The navy uses ESs in budget analysis. Navy budgets are large and complex, and requires numerous analysis to determine their content and make good decisions based on priorities.

Also, in general, there are ESs for use in agriculture, chemistry, oil exploration, low, meteorology, physics, process control, education, etc.

The following illustrates some available ESs in many real world applications :

1. Medical Applications :

MYCIN : Diagnosis and treatment of meningitis and certain other blood diseases.

EMYCIN : ES shell developed from MYCIN.

AL/X : ES shell based on MYCIN and PROSPECTOR.

CADUCES : Diagnosis internal medicine diseases.

VM : Monitors intensive case patients.

GUIDON : Instruction, bacterial infections.

INTERNIST : Diagnose internal medicine problems.

CASNET : Management of glaucoma treatment.

2. Chemistry Applications :

DENDRAL : Identify organic compounds from mass spectrometer data.

META-DENDRAL : Induces rules of molecular structure from mass spectrometer data.

CLONER : Design new biological molecules.

SECS : DEsign complex and organic molecules.

MOLGEN : Plan experiments on molecular genetics.

3. Geology Applications :

PROSPECTOR : Identification of minerals and selection of drilling sites.

LITHO : Interpret oil well log data.

4. Engineering Applications ;

SACON : Structural engineering.

STREAMER : Instruction, operation of steam power plant.

5. Electronic Applications :

SU/X : Signal interpretation.

SOPHIE : Instruction, circuit fault diagnosis.

EURISCO : Design, 3D micro electronics.

6. Computer Applications :

XCON : Configure DEC computer systems.

XCON/R1 : Configure VAX computer systems.

BDS : Diagnosis bad parts in switching networks.

7. Mathematical Applications :

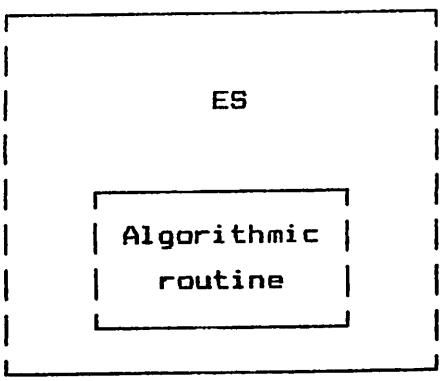
MACSYMA : Solves mathematical problems of algebra, trigonometry, and calculus.

TYPES OF ESs :

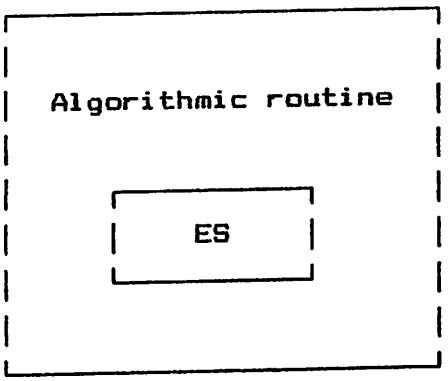
There are several types of ESs. These types include the stand-alone, hybrid, linked, dedicated, and real time ESs.

Stand-alone ESs : A stand-alone ES is the one that runs by itself and fully occupies its host computer. Most of ESs that have been encountered are of this type.

Hybrid ESs : ESs that are embedded in or have some algorithmic routines embedded in it are called hybrid ESs.

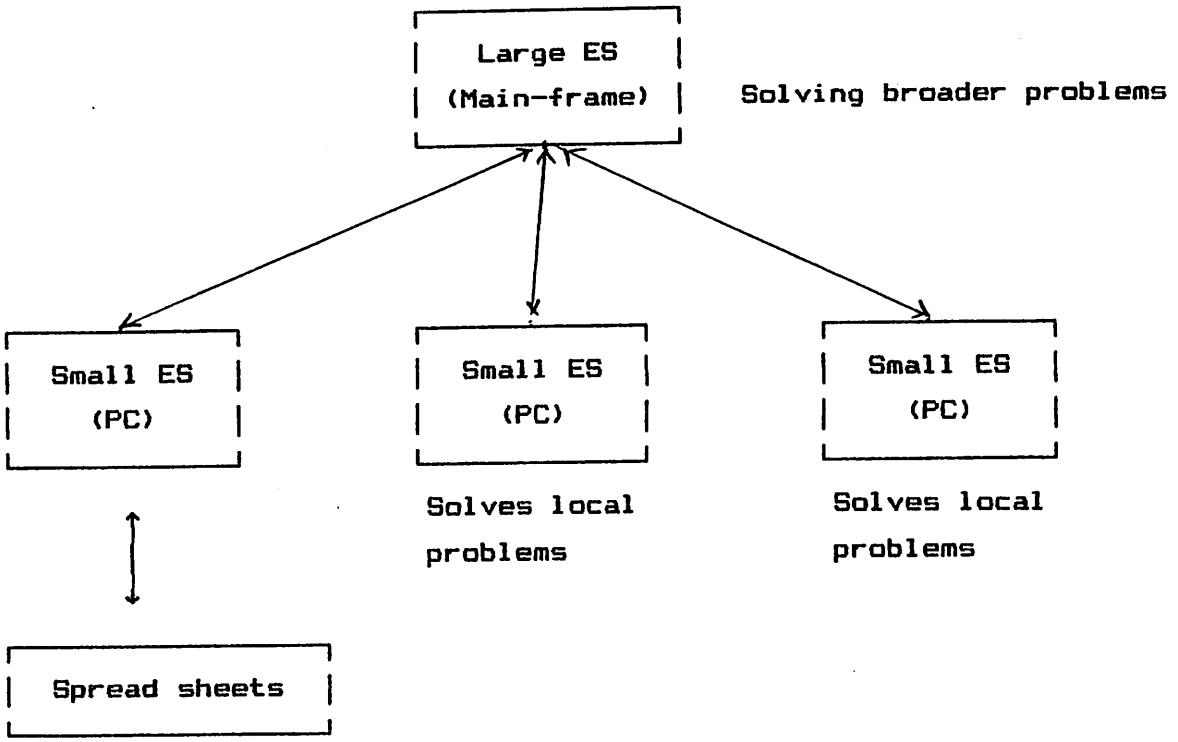


Algorithmic routine embedded in an ES



ES embedded in an Algorithmic routine

Linked system : Another form of mixed systems is the one that links conventional programs and ESs. Commonly, the ES requires input data from another source in order to solve the problem or make its decision. For example, many ESs are set up with links to external software packages such as spread sheets or dataBase management systems. It is also possible to link multiple ESs where these linked systems may exchange information with each other as shown :

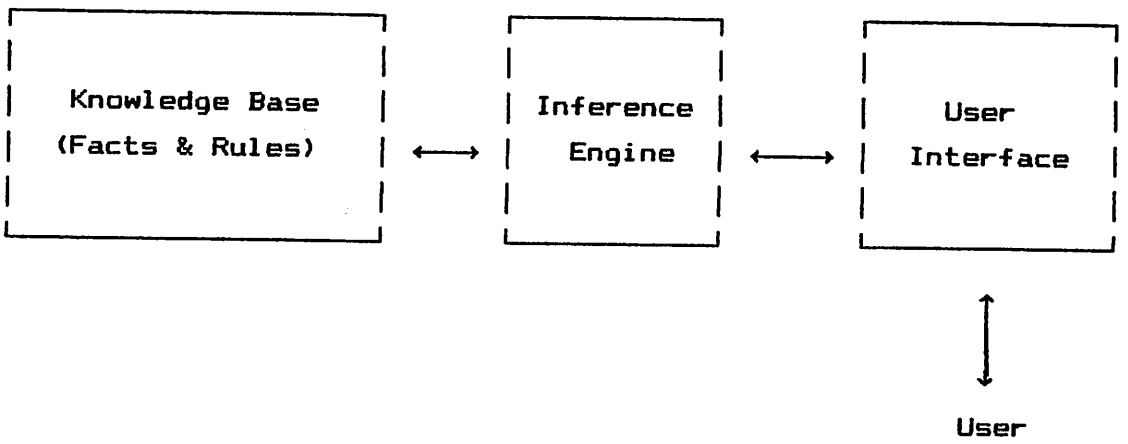


- **Dedicated ESs :** Another type of ESs is the dedicated ES such as the one burred in a weapon or a factory process-control equipment. In this case, the microprocessors burred in a weapon or an equipment are running ESs as well as standard algorithmic software.

- **Real-time ESs :** A real-time ES is designed to respond quickly to inputs and perform the necessary processing almost immediately.
An example of a real time ES is the one in a fighter or a bomber aircraft. In this case, the real-time ESs might monitor the radar, the infrared sensors, and another input sources to reach a conclusion and recommend a course of action. Recall, since a pilot needs to know "right now", the real time processing is required.

THE STRUCTURE OF AN ES :

An ES simply comprises 3 main components as follows ;



The Knowledge Base : It comprises a series of facts and rules about a specific problem domain from which the ES draws its expertise.

A fact is a clear, concise statement which expresses something which is true within the particular problem domain. For example, the following statements are all facts :

Kalifa is a man.
Rami lives in a house.
Shadi drives a car.
Mona has black hair

A rule is usually of the form :

IF statement 1 THEN statement 2.

where statement 1 and statement 2 are both expressions which may or may not be true.

The rule is simply stating that if statement 1 is true then this implies that statement 2 is true. For examples :

If a man marries twice, then he is a bigamist.
If a person has $1,000,0000, then he is a millionaire.

Such facts and rules must be expressed in terms of an appropriate programming language such as PROLOG.

The Inference Engine : In order to make use of the expertise which is embodied in the knowledgebase, the ES must also possess an element which can scan facts and rules, and provide answers to the queries given to it by the user. This element is known as the Inference engine. The inference engine has the ability to look through the knowledge base and apply the rules to the solution for a specific problem. It is therefore the driving force of the ES.

The User Interface : It is the means by which the user communicates with the ES. Ideally, this interface should be as english-like as possible so as to facilitate use by inexperienced users. That is, an ideal ES would allow the user to type (or speak) his questions to the system in english. The system would then recognize the meaning of the questions, and use its interface engine to apply the rules in the knowledgeBase to deduce an answer. This answer would then be communicated back to the user in simple english. Such an ideal system is ,however, not likely to be available in the immediate future due to the difficulties which arise in trying to program a computer to recognize and understand the meaning of even the most simple english sentences and phrases.

The ES can also comprise the following components (some of them are usual optional features on many systems) :

The Explanation Facility : which explains the reasoning of the system to a user.

The Working memory : It is a global dataBase of facts used by the rules.

The Agenda : which is a prioritized list of rules created by the inference engine, whose pattern are satisfied by facts in working memory.

The Knowledge Acquisition Facility : It is an automatic way for the user to enter knowledge in the system rather than by having the knowledge engineer explicitly code the knowledge.

METHODS OF KNOWLEDGE REPRESENTATION :

There are several methods of knowledge representation which can efficiently support the structuring and processing of knowledge. These methods include ,for example, production rules, semantic networks, schemata, frames, and predicate calculus.

PRODUCTION RULES :

This method represents knowledge in the form of IF/THEN type rules. They are called production rules. The following are examples of such rules:

IF it is very heavy
   and it is a mammal
   and it lives in water
THEN it is a dolphin.

IF the regulator output is zero
   and the power is ON
   and the regulator input is correct
THEN the regulator circuit is defective.

Systems that are set up on the bases of production rules are called production systems. Thus an ES whose knowledge base is represented in the form of production rules is a production system.

SEMANTIC NETWORKS :

Semantic network , or semantic net , or net , is a classic AI representation technique used for "propositional information". A semantic net is sometimes called a "propositional net".

A "proposition" is a statement that is either true or false such as :

all dogs are mammals,
a triangle has 3 sides.

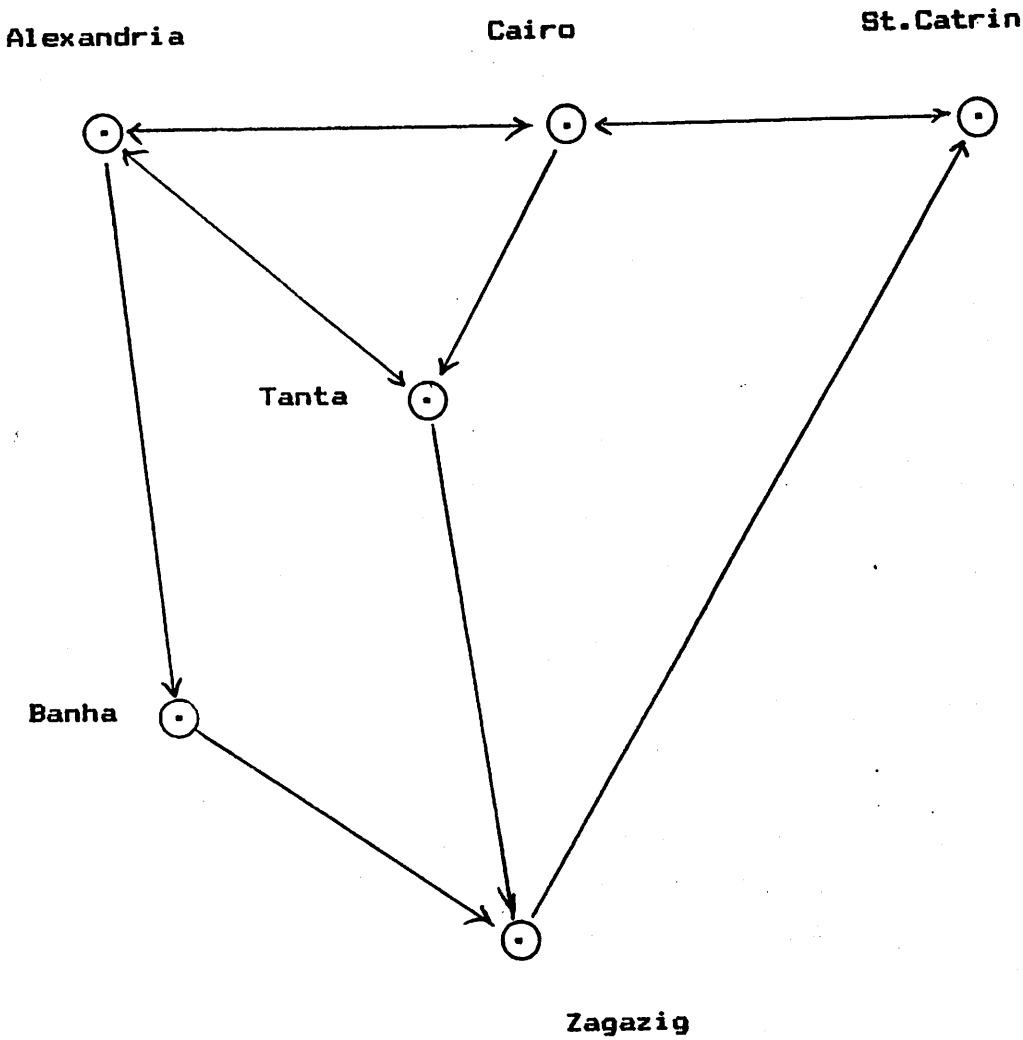
A proposition is always true or false and is called atomic because its truth-value cannot be further divided.

Semantic nets were first developed for AI as a way of representing human memory and language understanding. semantic nets are used to analyze the meaning of words in sentences. Since then, semantic nets have been applied to many problems involving knowledge representation.

The structure of a semantic net is shown graphically in terms of nodes and arcs connecting them.

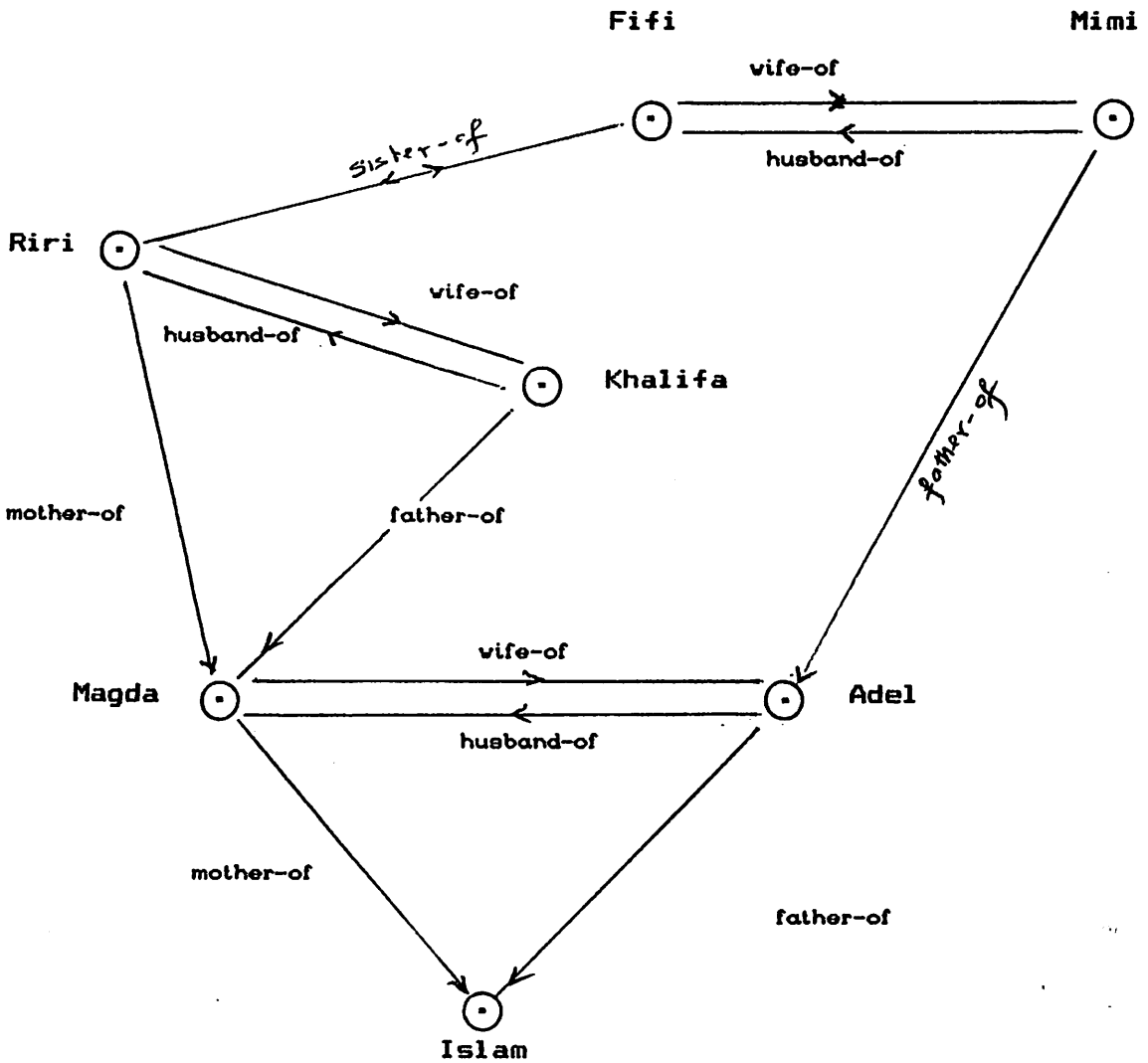
Nodes are often referred to as objects and arcs as links. The links of a semantic net are used to express relationships. Nodes are generally used to represent physical objects, concepts, or situations. A single link and its two nodes can be interpreted as chunk of knowledge.

The following figure shows a net in which the links indicate an airlines routes between cities:



(A general net --- A directed graph)

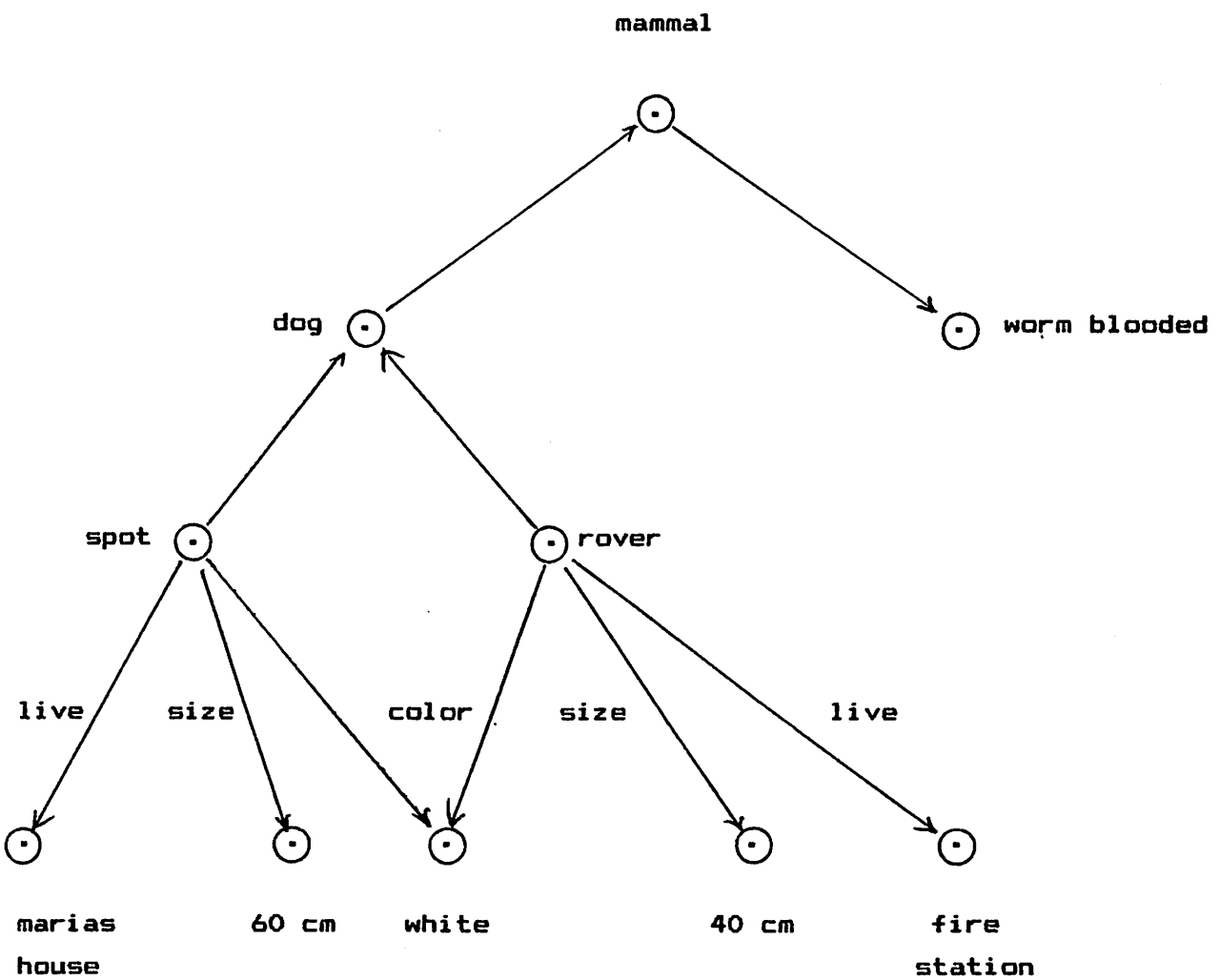
And in the following figure, the links show the relationship between members of a family :



Relationships are of of primary importance in semantic nets because they provide the basic structure for organizing knowledge. Without relationships, knowledge is simply a collection of unrelated facts.

With relationships, knowledge is a consistent structure about which other knowledge can be inferred. For example, in the figure above, it can be inferred that Riri and Khalifa are the grandparents of Islam even though there is no explicit line labeled "grandparent-of".

Another example of a portion of a knowledge base represented in the form of semantic network is as follows :



In this figure, the following relationships have been represented :

ISA(dog,mammal).	i.e. the Dog is a mammal The ISA is a relation which provides information that every dog is a mammal but not every mammal is a dog (from the arrow above).
ISA(spot,dog).	
ISA(rover,dog).	
Live(marias_house,spot).	i.e., Spot lives in marias_house.
Live(fire_station,rover).	The live relation provides information to where an animal lives.
Color(white,spot).	Spot color is white.
Color(white,rover).	
Temp(worm_blooded,mammal).	i.e. Mammals are worm-blooded. Temp is the relation that describes the body temperature.
Size(60_cm,spot).	i.e., spot is 60 cm high
Size(40_cm,rover).	

SCHEMATA :

A semantic net is an example of a **shallow (simple) knowledge structure.** The shallowness occurs because all the knowledge in the semantic net is contained in the links and nodes (chunks). The term knowledge structure is analogous to a data structure in that it represents an ordered collection of knowledge rather than just data.

A deep knowledge structure has causal knowledge that explains why something occurs. For example, it is possible to build a medical ES with shallow knowledge as follows :

IF a person has a fever THEN take an aspirin.

But these systems do not know the fundamental biochemistry of the body and why aspirin decreases fever. In other words, the expert system's knowledge is shallow because it is based on syntax and not semantics, where any two words could be substituted for X and Y in the following rule :

IF a person has a (X) THEN take a (Y).

where X and Y are not variables in this rule, but represent any two words.

Doctors ,for example, have **causal knowledge** because they have taken many courses and have experience from treating ill people. If a treatment is not working right, doctors can **reason** about it to find an alternative. In other words, an expert knows when to break the rules.

Many types of real world knowledge cannot be represented by the simple structure of a semantic net. More complex structures are needed to better represent complex knowledge structures.

In AI, the term schema is used to describe a more complex knowledge structure than the semantic net.

FRAMES :

One type of schemata that has been used in many AI applications is the frame. Another type of schemata is the script, which is essentially, a time-ordered sequence of frames.

As a method for understanding vision, natural language, and other areas, frames provide a convenient structure for representing objects that are typical to a given situation such as stereotypes. In particular, frames are useful for simulating common sense knowledge which is a very difficult area for computers to master.

While semantic nets are basically a 2 dimensional representation of knowledge, frames add a 3rd dimension by allowing nodes have structures. These structures can be simple values or other frames.

PREDICATE CALCULUS :

The predicate calculus is a formal notation used to describe objects and their logic relationships. The following example shows some natural language statements and their corresponding predicate calculus statements :

(1) Husband_and_Wife(john,mary)	John and Mary are husband and wife.
(2) Lives_in(mary,new_york)	Mary lives in New York.

In these examples, Mary, New York, and John are arguments. Husband_and_wife, and lives_in are the predicates.

Predicate calculus ,therefore, is a formal language with its own syntax and grammar which can be used to represent knowledge. Inference mechanisms can evaluate statements represented by predicate calculus and draw conclusions.

The programming language PROLOG is extremely well suited for expressing the language represented in the predicated calculus form.

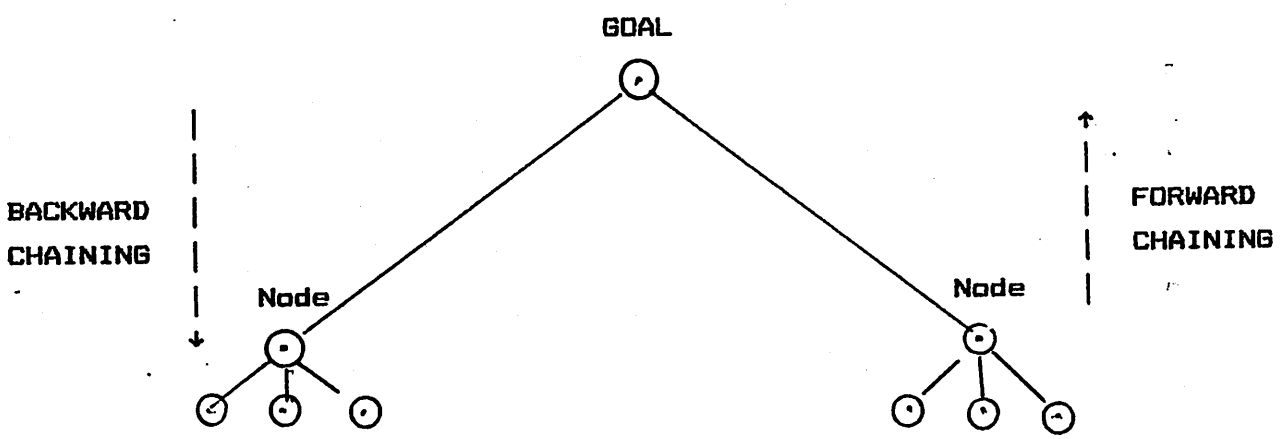
METHODS OF INFERENCE (REASONING) :

This topic is particularly important in ESs because inference is the common technique by which ESs solve problems. There are 2 common ways of inference. Inference with FORWARD CHAINING, and inference with BACKWARD CHAINING.

FORWARD CHAINING is sometime called data-driven inference, and BACKWARD CHAINING is sometime called goal-driven inference.

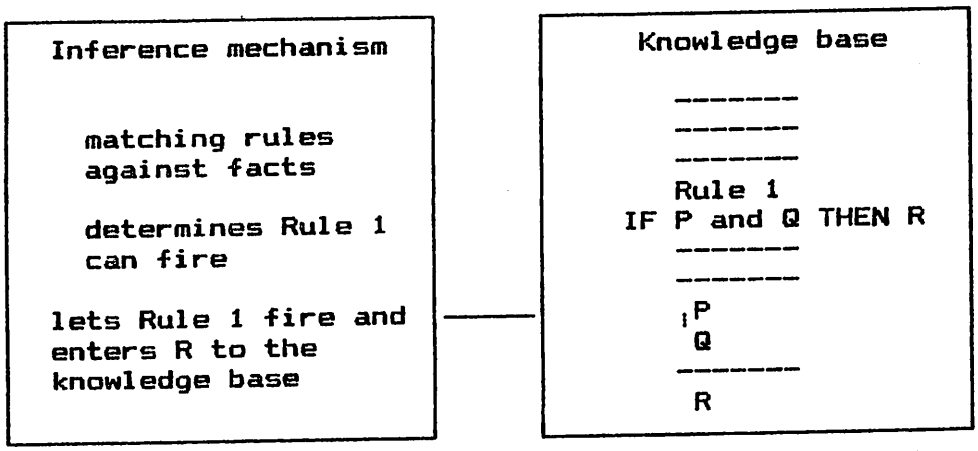
FORWARD CHAINING :

The process in which knowledge is searched or traversed from some given facts to its goal is called a FORWARD CHAINING process. The knowledge base is searched for rules that match the known facts and the action part of these rules . . . This process continues until either the goal is reached, or until no more rules can fire.



BACKWARD CHAINING :

Backward chaining starts from a goal (i.e., conclusion). All rules that contain this conclusion are then checked to determine whether the condition part of the rules have been satisfied. In other words, the condition parts are considered as subgoals and the knowledge is searched again for rules that match the subgoals. The process is repeated until all generated levels of subgoals are satisfied or no more rules can fire.



(The Inference mechanism through the knowledge base)

The function of the inference mechanism include the following :

1, To select the rule to be processed,
2, To determine how and where the rules be processed,
3, To determine which actions are to be taken, and
4, To control the dialog with users.

INFERENCE MECHANISM :

The inference mechanism represents the logical unit by means of which conclusions are drawn from the knowledge base according to a defined problem-solving processes of human expert.

A conclusion is reached by applying a rule of existing facts. The following explains this process:

Consider the following knowledge base which consists of 2 facts and a rule:

Fact : P
Fact : Q
Rule : IF P AND Q THEN R.

Applying the rule means: from the condition "P AND Q" draw the conclusion "R". That is, if the fact P is true in the knowledge base, and the fact Q is true in the knowledge base, then add the new fact "R" is true to the knowledge base.

A fact exists in an ES if it is contained in the knowledge base. The facts specified in the IF part of the rule are "premises" and the fact contained in the THEN part is called "conclusion".

When a rule is applied to any fact(s), we say it fires. The firing of a rule results in the entry of a new (inferred) fact into the knowledge base.

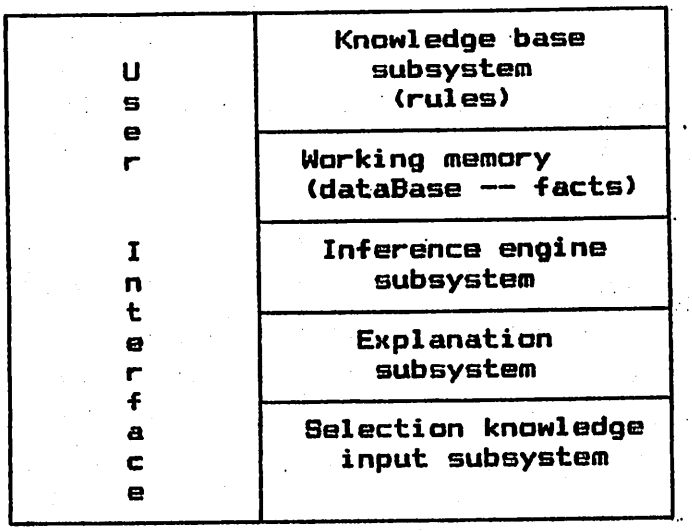
HOW ES WORKS :

We have seen that an ES can consists of 4 main parts : a knowledge base, an inference mechanism, working memory (data base), and user interface.

Most rule-based ESs use BACKWARD CHAINING in their inference process. The inference engine will search through the knowledge base for rules that match the known goal. When a rule is satisfied, a new fact is drawn and stored in a place in memory ,called the working memory which is a global dataBase of facts used by the rules, as a new subgoal. Thus the dataBase is the working memory area where the current status of the system is stored. The searching process continues using the knowledge in the knowledge base and the knowledge in the dataBase until all subgoals in the database turn to be facts. That is, the goal is reached, or until no more rules can fire.

The user can ask the ES how/why the conclusion was taken?. The part of the ES that is responsible about this is called the **explanation subsystem**. This subsystem tells the user the line of reasoning followed by the inference engine in reaching its final conclusion.

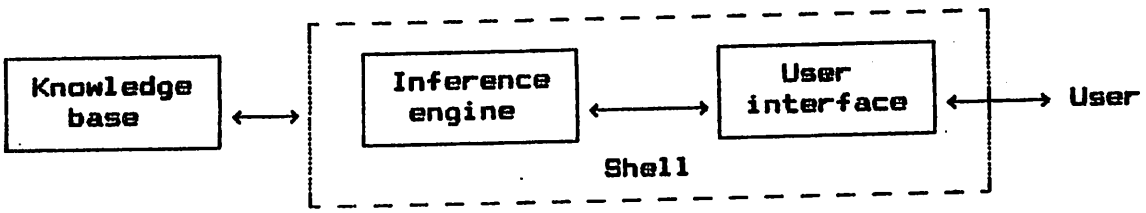
It should be noted that there is another ESs part called the **knowledge input subsystem**. The function of this part provides a convenient means for entering new rules or making modifications to the rule based system.



(The components of an Expert System)

ESs SHELLS :

The most useful tool for creating an ES is a SHELL (ES Generator) which consists of all the standard parts of an ES except that it has empty knowledge base.



There are two main types of the shells : The INDUCTION SHELL and the RULE-BASED SHELL.

The INDUCTION SHELL :

This shell allows us to enter knowledge as examples in a matrix form. The shell then infers rules that are used by the inference engine to reach the conclusion.

The RULE-BASED SHELL :

This type allows us to enter knowledge as rules and facts. The rule-based shells are more common.

Many shells are available now for PC computers and can also be obtained for mini- and main-frame computers.

ESs Limitations :

(1) The lake of causal knowledge : A practical limitation of many ESs today is the lake of causal knowledge. That is, the ESs do not really have an understanding of the underling causes and effects in a system. It is much easier to program ES with shallow knowledge based on empirical and heuristic knowledge than deep knowledge based on the basic structure, function, and behavior of objects. For example, it is much easier to program an ES to prescribe an aspirin for a person's headache than program all the underlying biochemically, physiological, anatomical and neurological knowledge about a human body. The programming of a causal model of the human body would be an enormous task and even if successful, the response time of the system would probably be extremely slow because of all the information that the system would have to process.

(2) Heuristic knowledge : One type of shallow knowledge is heuristic knowledge (the term comes from Greek and means to discover). Heuristics are not guaranteed to succeed in the same way that an algorithm is a guaranteed solution. Instead, heuristic knowledge are empirical knowledge gained from experience which may aid in the solution but are not guaranteed to work. However, in many fields such as medicine and engineering, heuristics play an essential role in some types of problem solving. Even if an exact solution is known, it may be impractical to use because of cost or time constraints. Heuristics can provide shortcuts that can reduce time and cost.

(3) Another problem with ESs today is that their expertise is limited to the knowledge domain that the systems know about. Typical ESs cannot generalize their knowledge by using analogy to reason about new situations that the way people can. Although rule induction helps, only limited types of knowledge can be put into an ES this way. The customary way of building an ES by having the knowledge engineer repeat the cycle of interviewing the expert, constructing a prototype, testing, interviewing, and so on is a very time consuming and labor intensive task. In fact, this problem of transferring human knowledge into an ES is so major that it is called the knowledge acquisition bottleneck.

In spite of their present limitations, ESs have been very successful in dealing with real world problems that conventional programming methodologies have been unable to solve, especially those dealing with uncertain or incomplete information. The important point is to be aware of the advantages and limitations of this new technology so that it can be appropriately utilized.

TOOLS OF ESs :

A development tool is a program or a collection of programs that facilitates the creation of other software. There are several kinds of tools ----- general tools for generic software development, AI tools, and ESs tools.

A spectrum of such tools is shown in the following figure :

Conventional languages		AI languages		
ASSEMBLY	HIGH-LEVEL	LISP	PROLOG	ES SHELLS

(Development Tools)

The easiest way to develop an ES is to use an ES shell; all what we have to do is to write the knowledge base.

If a convenient ES shell is not available, then an AI language is a 2nd choice. The reason is that AI languages (such as PROLOG) have their own built-in inference mechanisms.

If an AI language is not available, then structured programming languages (such as C or PASCAL) is a 3rd choice.

Finally, an ASSEMBLY language (not recommended) is the last choice, but it should be noted that this will be a difficult process.

THE FUTURE :

It is very difficult to speculate about the future possibilities that AI ınd ESs may offer to the computing industry and to the public in general.

One thing is ,however, certain: indeed, it is quite possible that the ield of AI is one of the most promising advances in technology that the ıorld has ever seen.

The computing or information technology industry is (in comparison with ıuch established industries as engineering) relatively immature (not yet ully developed). That is, there is no long history behind the computing ndustry. The methods, techniques, and concepts which are being applied in omputing are themselves not yet fully tried and tested,. Yet the industry s growing at an amazingly rapid rate ---- much more rapidly than the rate t which any earlier industrial revolutions have developed. It is for this eason that theory and research are perhaps rather far ahead of practical se.

This situation ,however, will not continue for ever. As previously tated, the world of commerce and industry is beginning to wake up to the otential that AI has to offer.

Some of the more obvious opportunities which are currently available are iscussed briefly as follows :

The use of robots is growing into many industrial plants. This pplication of AI technology will ,without doubt, spread into all areas of ngineering and the "hard" construction industry in addition to areas such s laboratory automation.

Work on computer vision, speech recognition, and natural language understanding will ,when it become true, produce computers which possess most of the senses of a human being. The extent to which these senses will be able to be developed ,however, remains to be seen. The next stage would be to transfer these senses within a robot, to thus produce a machine which can seen us, hear us, understand us, and talk to us !. All of this may seem like something out of a science fiction film; however, stranger things have happen !.

The future of ESs themselves is perhaps even more assured than that of the remainder of AI fields. This is because ESs are already being put to practical use in many fields of human endeavor. The breadth of usage of ESs having started to expand so that such systems are used to simulate human expertise in all areas of life, both at home and at work.

Indeed, it is perhaps in the home where we will see some of the greatest and more obvious applications of ESs. ESs could be used for gaining advice on such subjects as legal problems. These systems could be available at the touch of a bottom at a home terminal connected via a telephone line to national or international networks of such facilities.

In the future, it could be possible to find a new inference mechanism (instead of the current inference mechanism of searching) based on neural networks behavior.

REFERENCES

1, Abdalla ElDaoushy, 1991
An Introduction to Artificial Intelligence through TURBO PROLOG
Institute of national planning, Cairo, EGYPT

2, Christopher JOhn Hogger, 1984
Introduction to Logic Programming
Academic press, Inc.

3, Claudia Marcus, 1986
PROLOG Programming, Applications for dataBase Systems,
Expert Systems, and Natural Language Systems
Addison-Wesley Publishing Company, Inc.

4, Harold S. Stons, 1985
Artificial Intelligence Research in JAPAN
Computer, IEEE Computer Society, Sept, 1985

5, Ivan Bratko, 1990
PROLOG, Programming by Artificial Intelligence, 2nd edition
Addison-Wesley Publishing Company.

6, Joseph Giarratano and Gary Riley, 1989
Expert Systems, Principles and Programming
PWS-KENT Publishing Company, Boston

7, Mohammed A. Ghonimy, 1985 (in arabic)
The future of Computers and Knowledge Systems
Ain-Shams University, Cairo, EGYPT

8, Mohammed A. Ghonimy, 1991
Tutorial 1, Expert Systems techniques, first National Expert Systems & Development Workshop, feb 26-28, 1991
Expert Systems for improved Crop Management Project, UNDP/FAO --- EGY/88/024

9, Neil G. Rowe, 1988
Artificial Intelligence Through PROLOG
Prentic-Hall International, Inc.

10, Peter Smith, 1988
Expert System Development in PROLOG and TURBO PROLOG
Sigma Press, UK &
Halsted Press (a division of John Wiley and Sons)

11, Stuart C. Shapiro and Others [Editors], 1987
Encyclopedia of Artificial Intelligence Volume 1
John-Wiley and Sons