Not entered (78)

DESIGN AND IMPLEMENTATION OF INDUSTRIAL EXPERT SYSTEMS

Dissertation submitted to the Jawaharlal Nehru University
in partial fulfilment of the requirements for
the award of the Degree of

MASTER OF TECHNOLOGY

in

Computer Science and Technology

by

SATYA PRASAD SAHU

School of Computer and Systems Sciences
Jawaharlal Nehru University
New Delhi-110067
January 1989

TO MY PARENTS

CERTIFICATE

This work, embodied in the dissertation titled, "**Design and Implementation of Industrial Expert Systems**" has been carried out by Mr.Satya Prasad Sahu, student in the School of Computer and System Science, Jawaharlal Nehru University, New Delhi.

This work is original and has not been submitted for any degree or diploma in any other university or institute.

Satya Prasad Sahu

Dr. S. Balasundaram
Associate Professor
School of Computer and
System Science
Jawaharlal Nehru University

Prof. Karmeshu
Dean, School of Computer and
System Sciences
Jawaharlal Nehru University

ACKNOWLEDGEMENTS

I am deeply thankful to Dr. S. Balasundaram for the guidance he has provided during the project . My fellow students in this school Mr. R. Srinivas,Mr. N.K. Jain have been very helpful. Mr. Rajnish Kohli of Hinditron Ltd. has been very kind in providing all possible help. I also thank Prof. Karmeshu for his encouragement. My thanks are also due to Mr. anjay Kumar Jain and Mr. Krishna Prasad for their assistance in typing work.

Satyaprasad Sahu

S.P. SAHU.

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 INDUSTRIAL EXPERT SYSTEMS

Expert systems are destined to occupy the centre stage of managerial decision making. They embody the rules and procdures that govern various operations in an industrial enterprise. By augmenting the traditional buisiness computing facilities that already exist, with the powerful A.I. tool of expert systems , an organisation can have a competetive edge.

The establishment of knowledge based systems is inevitable considering its all-encompassing potential and increasing acceptance in the western countries. The design and establishment of such systems is cost effective as there are a variety of tools available today at the disposal of a knowledge engineer. One can therefore look forward to a huge market for Industrial Expert Systems.

There are sevaral general benefits of an expert system.

i) There will be a systematic documentation of expertise.

ii) Consequently, there will be widespreasd availabiity of expertise

iii) There will be a drive towards standardising rules and procedures. This will result in greater productivty.

iv) External consultation will be minimized resulting in a lot of saving of time and money.

v) There will be a greater awareness about the corporate environment, both internal and external. Since there are so many advantages of of adopting expert systems, there will be a widespread use in key areas of business and industry. The list of application is very big. Historically, expert systems were first developed medical diagnosis and related fields (eg. MYCIN). Later such systems were also used in fields such as geology, chemistry and anthropology.

Theoritically, there exists an opportunity for implementing an expert system in any of the situations listed in the following page. However they can be impractical, uneconomic and unwieldy for several applications.

The claim that such systems can only be used for large applications is also not true. On the contrary, small areas are where the maximum opportunities lie. They can be developed rapidly and relatively inexpensively on easily available tools.

This project aims to highligt precisely this. By developing expert systems for small and medium sized problems and implementing them in the field, one can justify its need and underscore its utility in the industry.The various classes of application are as following:

* Diagnosing the cause of a situation (e.g., auditing, troubleshooting, debugging).
* Prescribing a course of action (e.g., planning, designing, repairing).
* Predicting what will happen (e.g., planning, forecasting speculating).
* Understanding what is happening (e.g.,interpreting, teaching, monitoring).
* Governing what is happening . (e.g., implementing, controlling, managing)

* Evaluating diagnoses (e.g,, prescriptions and predictions).

1.2 INDUSTRIAL SYSTEMS

Every industrial enterprise needs numerous systems that are stable, in order to run efficiently. They can be broadly classified into technological and manegerial systems. A majority of these systems are simple since they are governed by relatively simple and stable rules and procedures. But usually there are many systems which are governed by a complex set of rules and proceduces. The variability involved in these systems are tremendous. To handle such systems, the industry employs experts.

The traditional data processing methods have been successful in computerising the simple systems such as payroll, inventory control supply and distribution etc. The other manegerial systems such as sales forcasting, productivity diagnosis, Job costing advise falls within the domain of experts of who are few in number and expensive to hire. Coming to technological systems, part-programs for NC machines are procedural and do not fall within the purview of A.I.. There are certain robots for assembly which need rule bases consisting of complex rules. It is for such systems that we need to harness AI technology. Where ever there is an expert who takes decisions logically and rationally, he/she can be theoritcally replaced by an expert system.

Another important characterstic of Industrial systems is that they are inherently numerical. A majority of the variables involved are numeric.

Although Industrial systems are generally stable they are subject to frequent modifications owing to the dynamics of the environment.

The involvement of computer technology in industrial systems started with data processing. This was a labour saving device which stored and manipulated data bases. Commercial database systems like CODASYL, IMS. etc. were available. They were generally linked with documentation systems. The record keeping systems governed by software called the management information systems (MIS). There was further advancement in the software technology involved wherein, the record keeping system was further linked with a decision making system. This was called the decision support system (DSS). It followed simple procedural models imposed over the database. Some of the decision support systems used standard Operation Research models for optimal decision making. In fact,a vaste majority of procedural models in Operations Research are dealt with in the industry.

1.3 MODELS OF INDUSTRIAL EXPERT SYSTEM

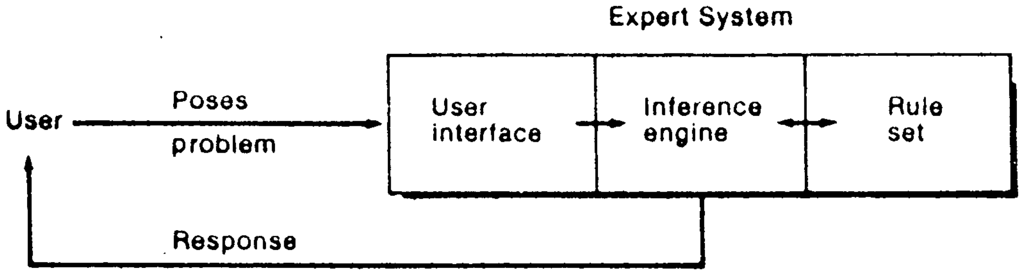
The traditional expert systems have a simple structure consisting of a simple user interface which channels I/O with the user. Based on the data and instructions given by the user, the inference engine works over a fixed knowledge base. The system which is generally called a production system gives the output of the reasoning. See Fig. 1.1

The Industrial expert system needs different configuration. As it was stated earlier the industrial system are essintially numerical. The inference engine would therefore work on a knowledge system consisting of the following.

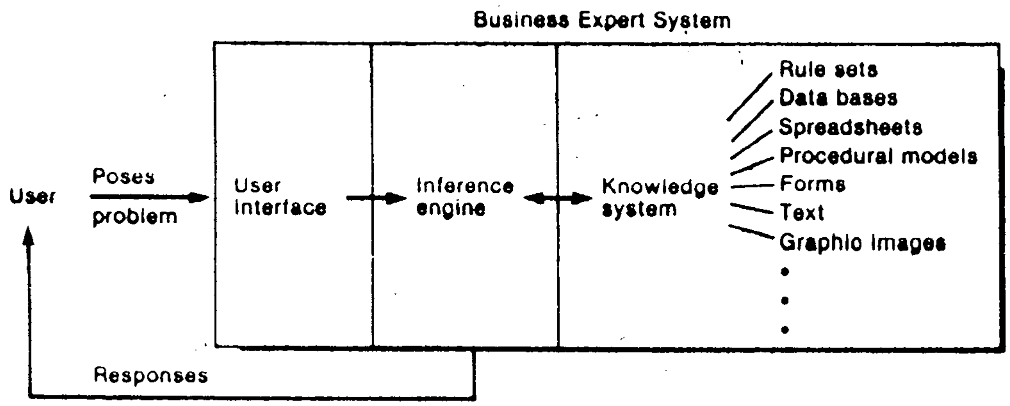
Data-bases

Procedural models

Structure of a conventional expert system



Structure of a business expert system



Rule bases

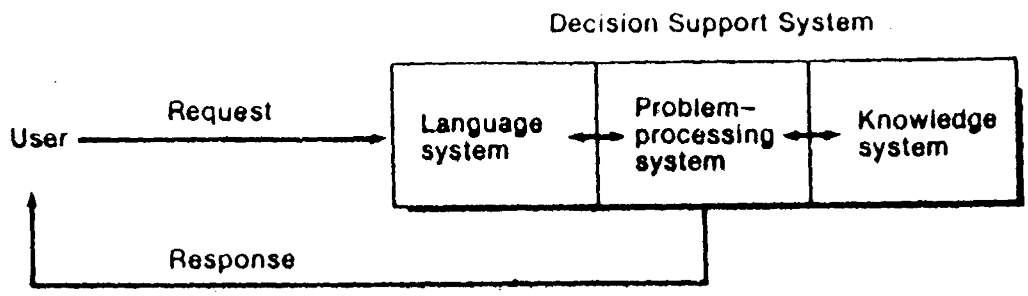
Text

Spreadsheet

Refer to Fig. The Industrial Expert System.

Industrial Expert Systems are supposed to be used primarily to make decisions in the same way as an expert would do. Thus, it is essentially a decision support system (DSS). The configuration of a decision support system is given in the figure 1.3. In a decision support system there will be an elaborate problem processing system, knowledge system and a language system. As suggested by Holsapple, C.W. et al. [7] the DSS has multiple problem processing facility which consists of, database managements spreadsheet analysis, statistical analysis, etc. All this is part from an inference engine which also an essential feature. See fig. 1.4. The DSS therfore embodis a problem processor supporting many knowledge management abilities and many knowledge representation methods. With the development of natural language processors there can be alternative user interface methods. They include menu guidance, direct command structure, natural language conversation and customized interfaces. See Figure 1.5

DSS structure



Representative expert systems

Expert system	*Problem area*
DENDRAL [13]	Determines the chemical structures of unidentified molecules.
MACSYMA [14]	Solves differential and integral calculus problems in applied mathematics.
MYCIN [20]	Diagnoses and prescribes treatments for bacterial blood infections.
INTERNIST, CADUCEUS [19]	Diagnoses internal medical ailments.
CASNET [22]	Diagnoses glaucoma and recommends therapies.
PUFF [12]	Diagnoses lung dysfunctions.
SACON [2]	Advises how to analyze mechanical structures.
PROSPECTOR [6]	Determines the major types of ore deposits present in a geological site.
CRYSALIS [7]	Determines the protein structures of unidentified molecules from electron density maps.
R1, XCON [15]	Determines an appropriate computer system configuration for a customer's needs.

fig 1.4 **Problem processor supporting many knowledge management abilities and many knowledge representation methods**

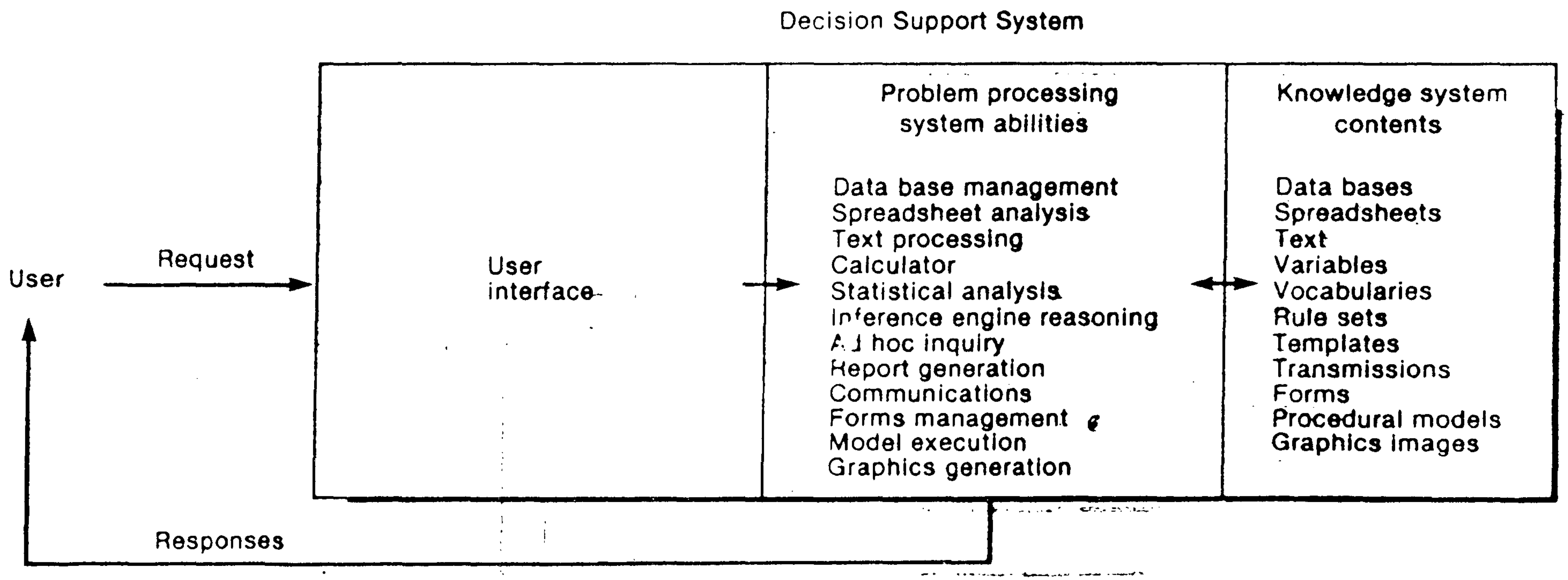
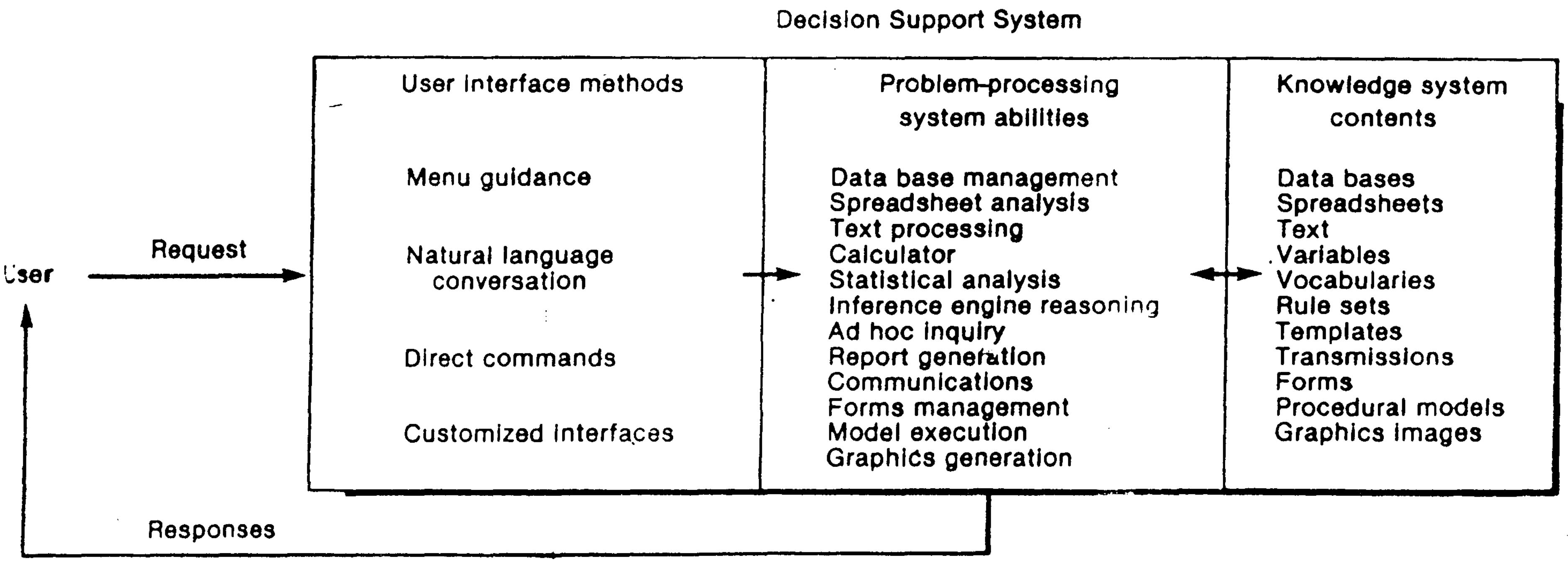


fig 1.5 Alternative interface methods



Thus, one can visualize the most general model of an Industrial Expert System. Most of the entities and concepts of this model, when viewed in isolation, are easy to implement. But to implement the entire model as a whole, we need to develop rigourous methods of software intergration. By software integration, it is meant, the functionality of one piece of software can be coordinated from another piece of software. Software integration is something quite different from software compatibility. If one program can read file or output of another program then the two programs are said to be compatible.

1.4 SOFTWARE INTEGRATION

The theory of software integration has identified three distinct approaches [11], [12] :

1. To begin with a group of compatible stand alone programs and add some software for coordinating their activities for integrating them.

Such integrating software is called an operating environment. Examples of generic operating environments includes DesQ, Gem, TopView, and Windows. The other operating environment work over specific collection of programs.

Data transfer between two programs by creating intermediate data files each time a transfer is needed. For small amounts of data, the 'cut' and 'paste' techniques can be used, Routines facilitating 'cut' and 'paste' have been implemented in this project.

2. In the nested (sometimes called inclusive) style of integration nesting is done on one or more secondary components within a dominant component.

   (eg. In LOTUS 1-2-3 the dominant component is the spreadsheet processor, the secondary components are rudimantary graph generator and pseudo-file management capability). The disadvantage with this style of integration is that, the functional capabilities of the secondary component is restricted by the dominant component.

3. The third major style of integration is the synergistic integration technique. Here, the integration is done in such a way that no component restricts the

functionalities of the other components. Thus, the total effect of the system would be more than the sum of the effect of the individual components (hence the word 'Synengistic') [13].

This third approach is the most suitable approach to make the Generalised Problem Processor a reality.

CHAPTER 2

PROSPER: DESIGN AND IMPLEMENTATION

2.1.0 INTRODUCTION

PROSPER is the name of the software which consists of two parts.

1. Quality Assurance Expert System
2. Productivity Evaluation and Diagnosis

These two pieces of software are designed to be applicable primarily for engineering industry. Also with minor modifications they can be used for other industrial enterprises.

Expert systems are not in use currently in Indian Industry, except for a few rule based applications in microchip manufacturing and robot technology. The idea of expert systems, as experts in industrial management is treated with disbelief. Far from being popular, they are being treated as highly futuristic and imaginary.

On the contrary there have been successful applications of expert systems in Europe and USA. There, this technology is graduating from a university/laboratory stage to a commercial stage.

2.2.0 SURVEY

With this background, we have started with a list of possible opportunities for expert systems in industry. Initially the list was very long. It was narrowed down to the following :

1. Expert system for process planning; Methodizing and process sheet preparation.
2. Expert system for costing of manufactured parts (it was later decided that procedural models would suffice for this task).
3. Expert system for production scheduling. There were precedents set in Italy where such an expert system had been designed and successfully implemented in a flexible manufacturing system environment.
4. An expert system for quality control auditing for quality assurance system.

5. An expert system for productivity evaluation and diagnosis.

6. An expert system for developing an investment plan for tax avoidance.

7. An expert system for avoiding stock loss and wastage in process industry.

This list was prepared after consultation with men in the industry. Those who were consulted include men from junior management to those from general management. The following criteria are used:

1. What are the decisions to be taken and who takes them ?

2. The existence of a rule base or heuristics.

3. It importance to the growth of the industry.

4. The interrelationship of rules and the traditional data processing.

Considering the availibility of expert ese and time we have decided to take up projects 4 and 5. The methodology for the design and implementation of one of the systems is inspired from Prof. Robert Kellers book "Expert

System Technology"[5], where he has advocated strutured system development. The other was designed and developed using methods given by A.B. Whinston, C.W. Holsapple[7].

2.3.0 QUALITY ASSURANCE EXPERT SYSTEM

This is a knowledge base system to assess the quality control system of a manufacturing concern. With this software, we can evaluate and diagnose the quality control and quality assurance of any firm producing engineering goods. The methodology for developing this system has been adopted from Prof. Robert Kellers book on expert system technology and some of them are originally prepared.

2.3.1 The Current Physical System

The decision makers' designation somebody i the middle management level who takes the major decision of accepting or rejecting the supplies. The current physical system includes the the quality assurance questionaires, the tenders and the quotations, the vendor's previous business record, Junior Engineers who go the vendor.

2.3.2 The Current Functional System

The current functional system is summarised in the data refer to appendix (A). (Also for a description of the procedure refer to appendix A1).

2.3.3 The New Physical System

It consists of the following hardware, software and other entities:

1. Stores and Purchase Office
2. PROSPER (Quality Assurance System)
3. Previous records of business of the propective vendor.
4. IBM ccompatibel PC AT/XT
   * 1 Mbytes Main Memory
   * 132 Column Printer
   * Monochrome or preferably colour display
   * 80 column stationaery

2.3.4 The New Functional System

The new functional system is summarized in the data flow diagram shown in the following page.

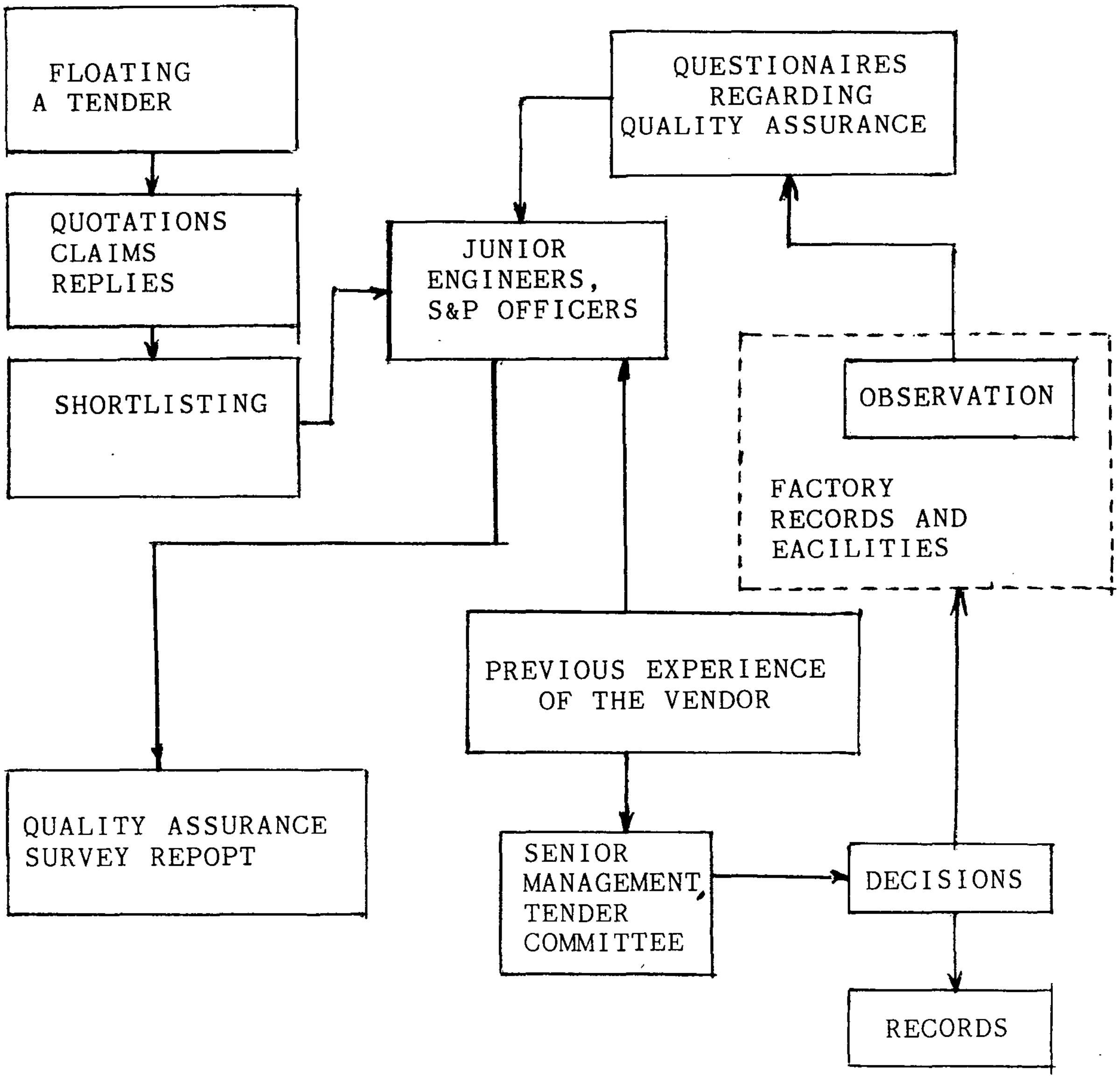


Fig: 2.1 CURRENT FUNCTIONAL SYSTEM

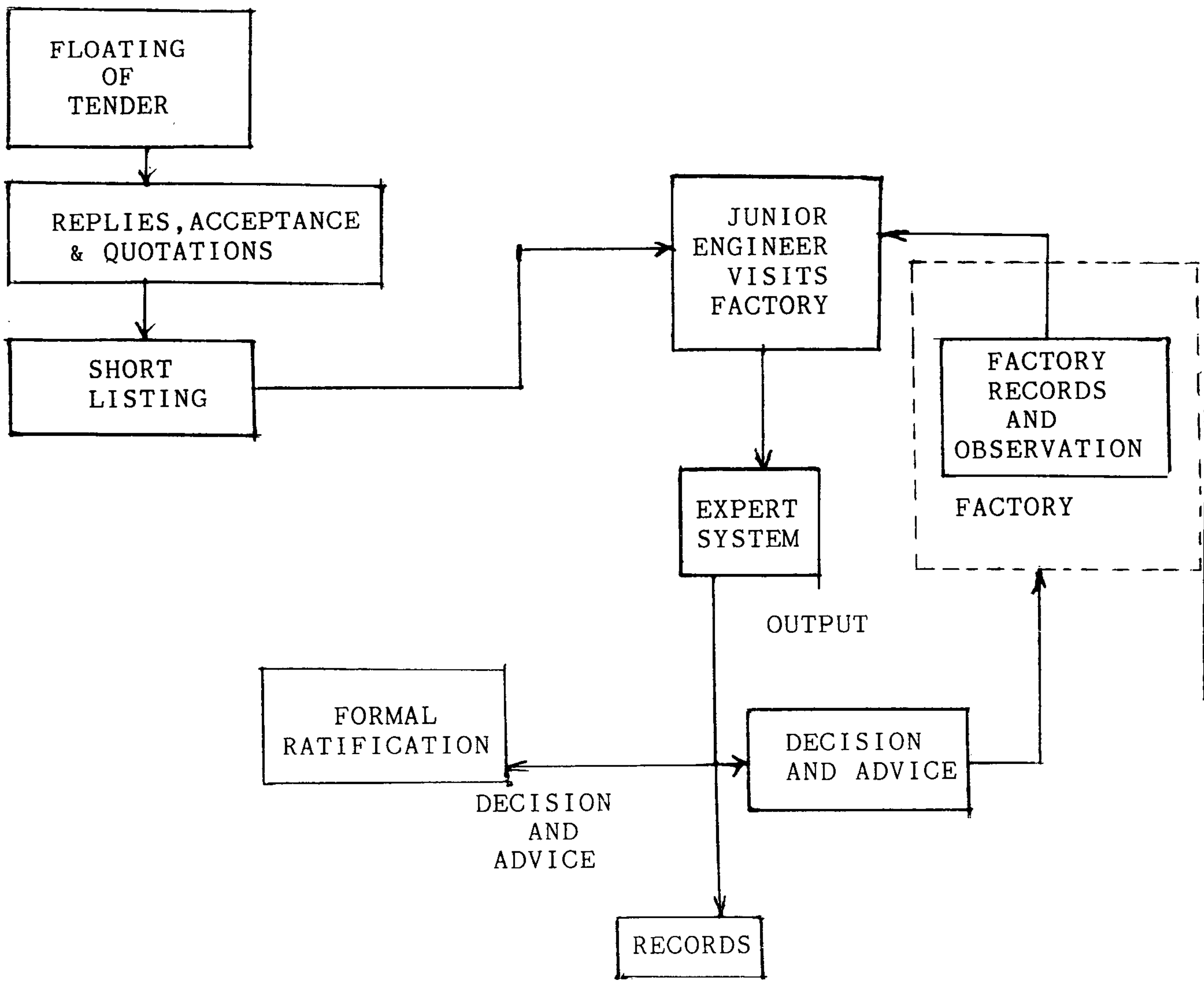


Fig: 2.2 NEW FUNCTIONAL SYSTEM

The main changes made are stated below :

1. The senior management or the tender committee which used to be the major decision making body, is now out of picture. It has been replaced by the expert system.

2. There will not be any survey report. There will be an advice generated by the expert system along with the major decision. The advice will be binding on the prospective supplier.

For a detailed description of the decision variables and the decision structure, please refer to the Appendix A.

This ends the structured analysis activity. We can now start with the structured design.

2.4.0 STRUCTURED DESIGN

A rough configuaration of the system has a clear separation between the data processing components and the A.I. component. The system, having furnished with the data processing components, starts off with the knowledge base (A.I. component).

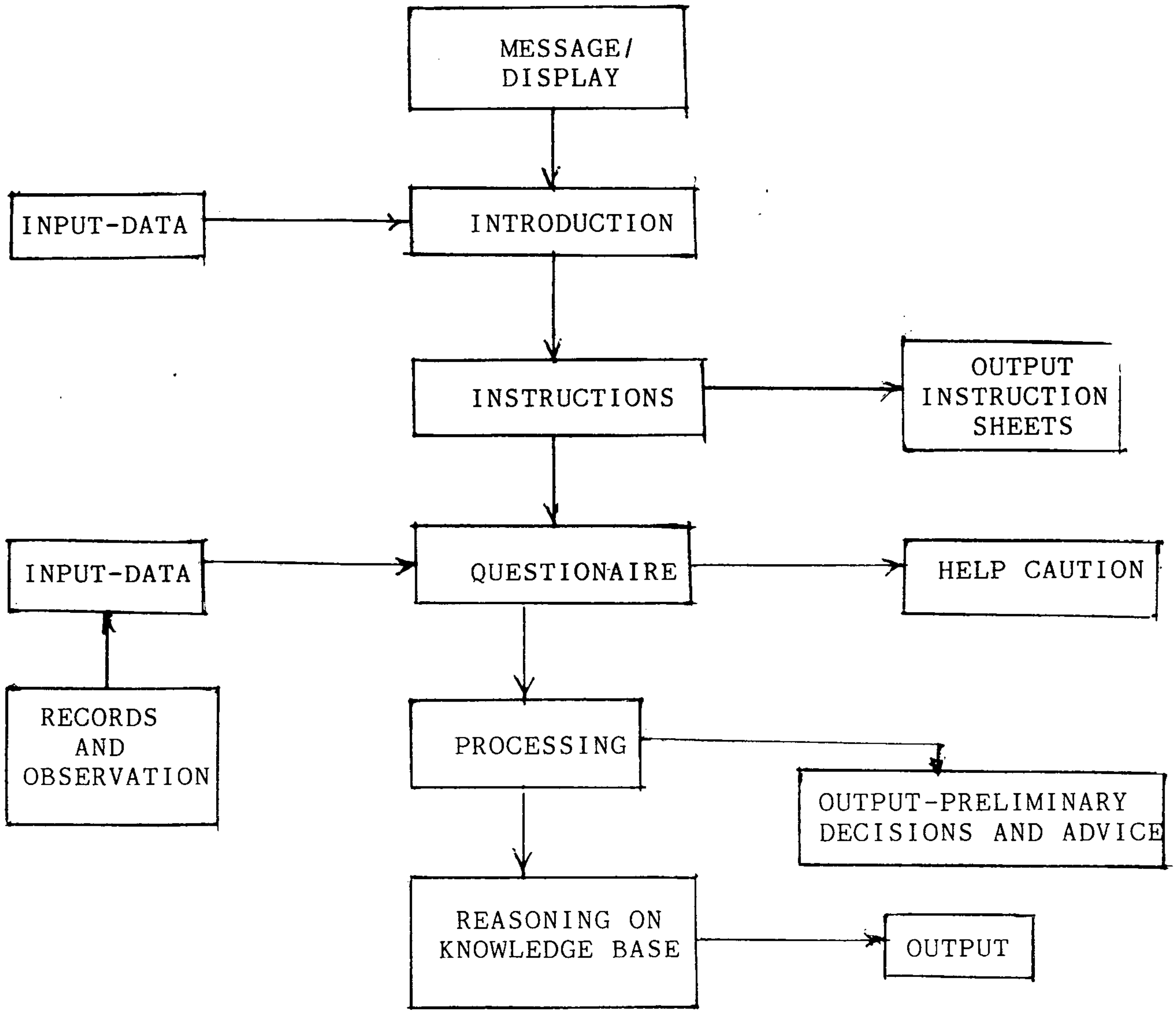


Figure 2.3 SCHEMATIC DIAGRAM OF
I/O AND PROCESSING IN PROSPER

2.4.1 Desing Specification of The Quality Assuance System

The Data Processing components : The Data Processing components include

1. The initial I/O of introduction
2. The output of instructions
3. Display of hints
4. I/O involving the questionaire which incudes five sets of questions.
5. All the processing with the inputs done during the questionaire.

The auxilliary facilities provided during the above processing period are

1. display of time and date
2. Help on an interactive basis.
3. Facility to escape to the DOS.

   See figure 2.4

The AI component : - The AI component does the reasoning with the data aquired and processed interactively. The data is imposed on a rule - base.

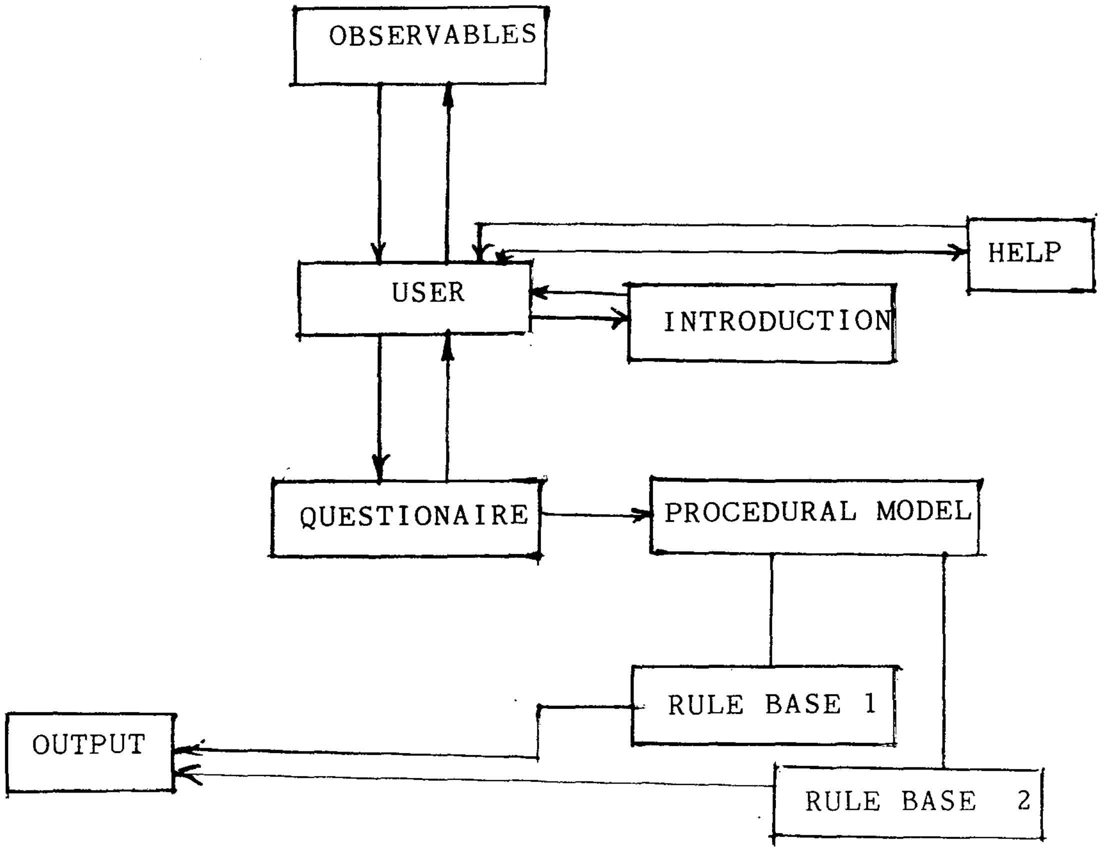


Fig. 2.9 : DATA FLOW DIAGRAM FLOW PROSPER

The rule base is the main part of the production system. The production system does not need any input from the user. It generates the output on the screen. the output should alternatively be able to be printed on the printer. (The output is the result of the various rules being fired).

2.4.3 Software Features :

The software features used in PROSPER are as under.

1. PROSPER is a batch file consisting of compiled files of PASCAL, PROLOG and DOS routines. The execution of PROSPER, therefore does not need. PROLOG or PASCAL compilers. (TURBO PROLOG and TURBO PASCAL have been used to develop PROSPER).

2. The software has the following characteristics

   (a) It is structured and modular in character.

   (b) It is user-friendly and interactive

   (c) The possibility of the run entering into infinite loops is very minimal. To this end several measures have been taken while writing program modules .

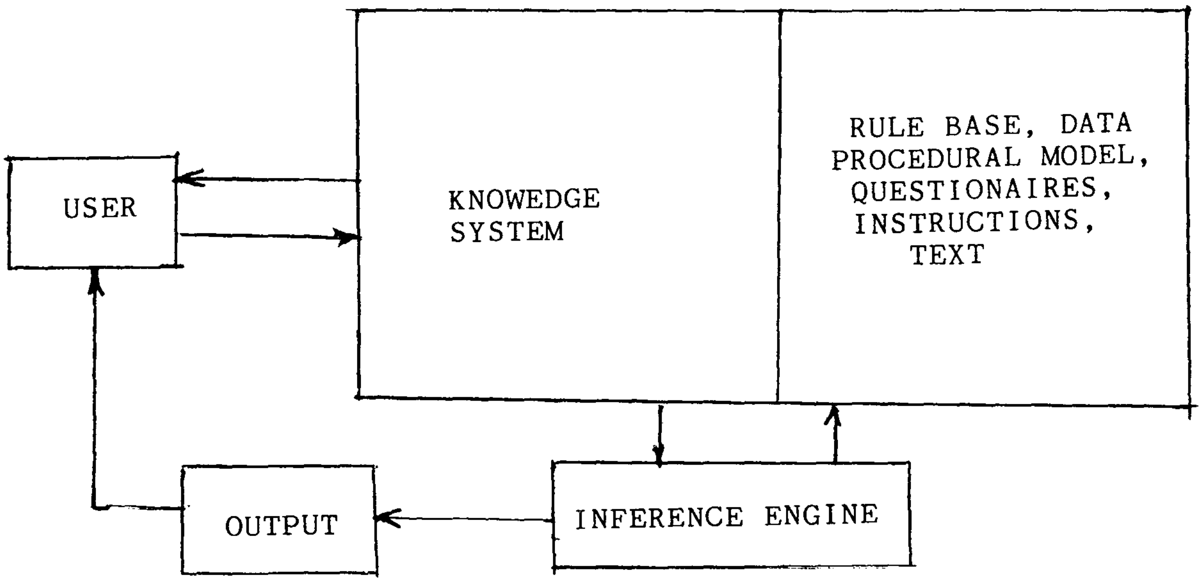


Fig. 2.5 : SCHEMATIC DIAGRAM : A.I.COMPONENT

3. Certain advanced features of TURBO PASCAL have been used in developing this programs. They include screen handling techniques, windowing, memory organization, use of DOS interrups etc. These are discussed in detail later in the chapter. In fact part of the reasoning has been done using TURBO PASCAL in a elementary rule based reasoning.

4. The crux of the reasoning have been developed using TURBO PROLOG. The TURBO PROLOG's inference engine has been exploited carefully and successfully. the decision structure and the rule base minimized the necessity of using backtracking. As a rule, and this has been suggested by Boroland corporation itself, cut and fail predicates have been avoided as far as possible. The compiled program has ben made externally executable. The procedure has been explained later in this chapter.

5. Colour graphics has been used to make alternative displays windows.

2.4.4 Limitations

Some of the limitations of the software are:

1. The knowledge base cannot be interactively modifiable. Although the required data required to work the

knowledge base is fed interactively, the essential character of the knowledge base remains to be same.

2. The output of the system are essentially textual. It has been suggested by many that the output should have been supported by numberic data as well.

3. The data transfer of the output of the TURBO PASCAL program to TURBO PROLOG PROGRAM is done using a DOS harware interrupt. This is normally considered as the desiuable by many professional programmers.

4. This being only the prototype version stress has been laid on methodology of design and development, techniques, and usage but not completeness. There are several simple aspects of the software that are going to be implemented after suitable soft suggestions are received after implementation. As of now, this process would post-date the period of the project.

The features and problems of the knowledge base and its limitations are discussed in the Appendix . The problem of implementation discussed later in this chapter .

In order to modify the knowledge base one has to go to the source code and deal directly with the knowledge base there.

2.5 PRODUCTIVITY EVALUATION AND DIAGNOSIS

2.5.0 Introduction :

This is a knowledge-based system. Its purpose is to evaluate the productivity of a given production unit and subsequently provide diagnostic information. This is the prototype of the system which, when imporved upon, can possibily find commercial applications.

The methodology used in developing this system has been inspired from a series of papers by Boncczek, R.H., CW. Holsappe, K. Tam, et al [7] (see references).

In the developmental cycle the most important activity is the knowledge system specification. The following is a descreption of the same.

2. 5.1 Knowledge system specification :

Fule set specification starts with problem definition, In problem definitions, the input and output to the system are clearly specified. Here it is in the form of a data base which will be updated by the user. The database maintained is therefore, a dynamic database. The data will run on a procedural model and subsequently it will run the rule base to generate diagnostic advice.

The next step is the identification of variables invovled in the reasoning structure. They are classified systematically providing clear inofrmation with regard to its source, type, range, whether certainty factor is involved or not etc. in a prescribed form given in appendix A2. the bubble diagram describing the interrelationship between the variables is drawn, as shown in the following page. It is important to note that the numerous variables shown are actually the key control variables. The actual list of variables would have become unwieldy to handle.

Having obtained the variable specification we start defining the rules and the order in which they will be fired. A typical rule when coded in PROLOG looks like this.

```
/* Rule * /
notool :-
     /* input from .DAT files */
```

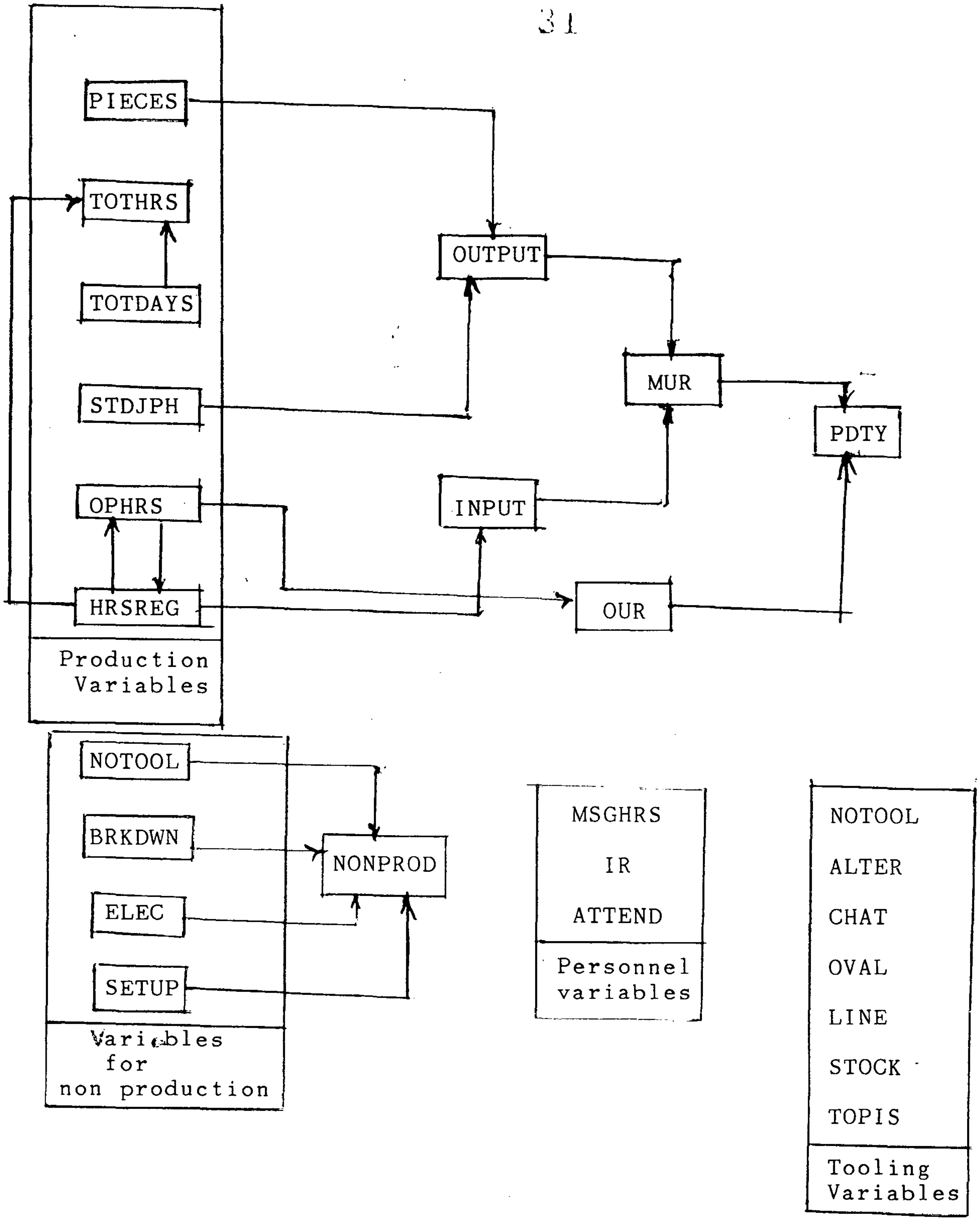


Fig. 2.6 : VARIABLE STRUCTURE AND INTER RELATION SHIP IN P.E.D.

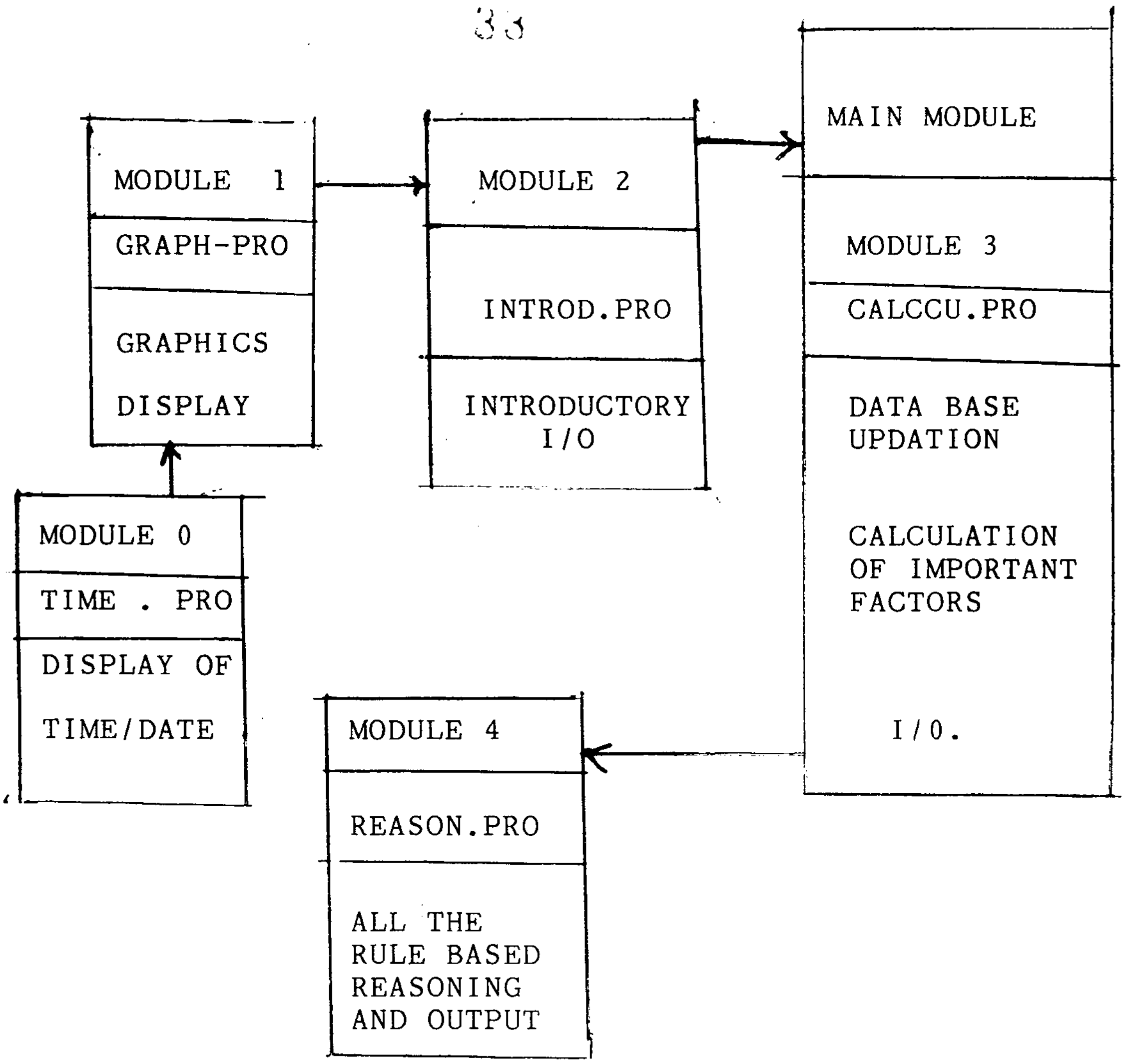


Fig. 2.7 : The functional modules

```
notool/tothrs G.T.  0.20,
Toolgeo,
Alter = 'false,'
Write ("STRING") !.
```

The comment "/*input from .DAT files */" is to indicate that the values of NOTOOL, TOTHRS and ALTER are to be read from the input device (disk in this case). The list of rules used in the Productivity Evaluation and Diagnosis is given in the Appendix.

The structure of program for Productivity evaluation and diagnosis is given in the following page. The methods of modular programming has been followed. The modularization is basically functional. Each module has a pre-defined functional role.

2.6 FEATURES OF TURBO PROLOG

Apart form the regular features of TURBO PROLOG and TURBO PASCAL certain special and advanced features of these languages were used. Some of these features are found to be essential for developing industrial expert systems.

The special features of TURBO PROLOG used were as follows.

1. Running programs using internal goals and compiled for external execution.

Internal goals are parts of the PROLOG Programs. They obviate the necessity of providing goals for execution of the programs by the user. In modular programming only one of the modules decided as the main module can contain the internal goal. The goal section of the program follows the predicates or global predicates section.

Internal goals are necessary for external execution. Compiling for external executiion can be done in TURBO PROLOG as follows :

The program or the project file should have only one goal. Each of the modules should be compiled as project file (or the program, if there is only one module). Using 'options' in the main menu the 'exe. autolink' should be chosen. The program should be run so that the object code is generated and an XXX.EXE file of the same name is created. It should be kept in mind that this code is about eight times the souce code in terms of memory. The external compilation is a two step process. Both steps are processed automatically with no user intervention necessary. The first step is the actual compile operation which creates an XXX.OBJ file. An internal linker then combines with this file with INIT.OBJ (which is privided by Boroland Co. itself) and portions of PROLOG.LIB to create the final XXX.EXE file. We need to have the linker file PLINK.BAT, a batch file with a similar structure as that of DOS' linker program LINK.EXE.

2. Linking TURBO PROLOG programs with routines of other languages.

As mentioned earlier, most industrial expert systems have data processing requirements within their knowledge system. Therefore, to be able to use routines of other languages is really a boon to the developers. However, routines cannot be called from any where within the TURBO PROLOG program modules. The routines should be modules in themselves and should be object programs. The procedures for doing this is as follows.

The 'option' menu should be used and it must then be specified that we need to compile a .OBJ file. This stops the compile process before the linking begins. We can then use an external linker succh as LINK.EXE to link the .OBJ files from other languages.

However, not all is so rosy as it may seems. The procedural programs do not fit into object oriented programs like PROLOG. This limits the utility of linking other languages with PROLOG. For example, a variable instantiated in one clause, doesnot retain its value for the next clause. This leads us to be the problem of software integration.

Tecchniques of integration: By integration it is meant, the transfer of data as well commands using well defined calling/sequences and modes. from one module to another module of a different class of language.

For example, the output of the PASCAL program could be a set of flages controlling the execution of the PROLOG program. The output can also be data with an appropriate structures. Which could bind the free variables as has been done in Quality Assurance Expert System. The choice of structured data is very limited. At the most we can have records in the .DAT files that instantiate with the compound object of the PROLOG module. In this way we can bind any number of variables, although most clauses require data only for a few variables. An example is given below.

```
RECORD = Array [1..4] of String;
OUTF = file type;
ASSIGN (OUTF, DATA.DAT);
RECORD : = [A,B,C,D];
WRITE (OUTF, RECORD);
```

The abvoe pascal program segments writes the value of the RECORD which of type array on to a variable on to a variable OUTF of file type. The dos file name is DATA.DAT where as OUTF is the symbolic file name.

To have access to the file DATA.DAT in the TURBO PROLOG module the following segment should be used.

```
domains
file = OUTPUT
RECORD = P (W,X,Y,Z)
Predicates
go
clauses
go :-
```

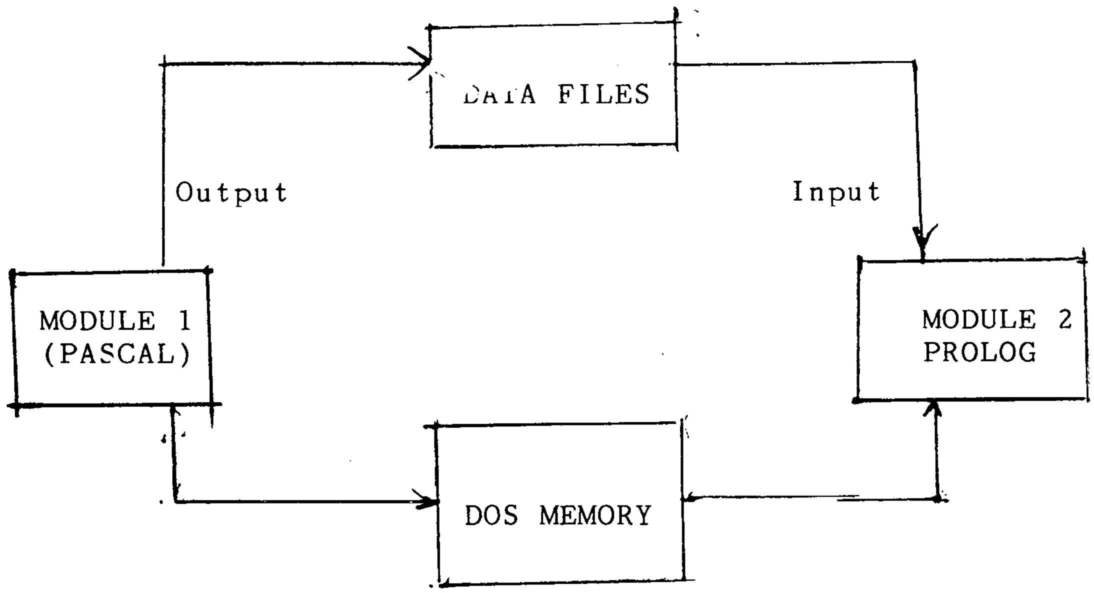


Fig. 2.8 : TECHNIQUES OF INTEGRATION

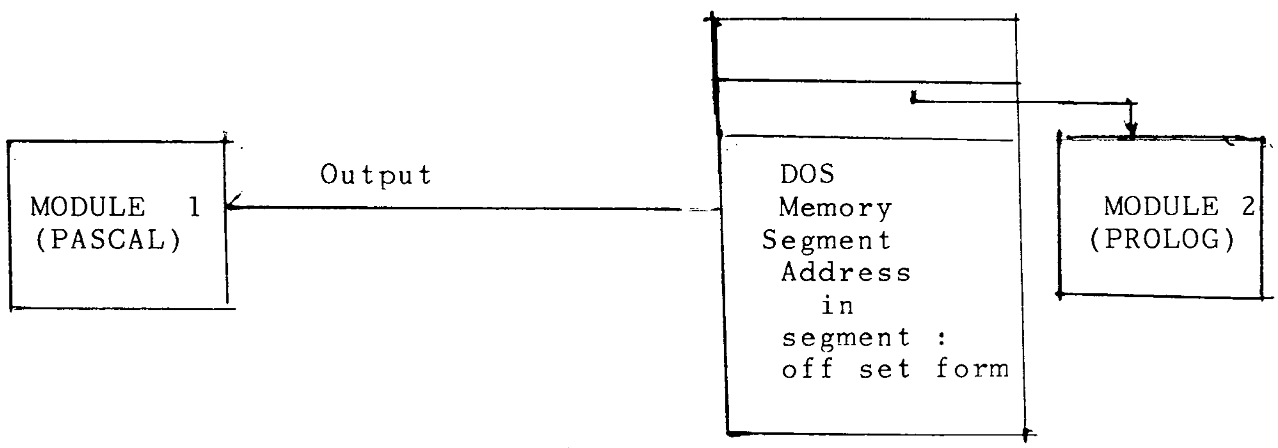


Fig. 2.9 : USING DOS MEMORY

```
openread (OUTPUT, "DATA.DAT"),
readdivce (OUTPUT),
read (P),
Closefile (OUTPUT).
```

As a result of this, the compound object variables W,X,Y,Z will get values of A,B,C AND D strings.

The other techniques is the data transfer using DOS memory. In this case the dodule can be called only once. The value of a variable on the DOS memory at an absolute address as figure in the following page. This address is derived out of the address in segment offset form. The TURBO PROLOG statments.

```
Mem Word (Segment, offset, word),
Mem Byte (Segment, offset, byte),
```

When excuted, shall store or read the word or byte (which can act as a flag) at the address specified in the given format.

3. Implementing Help

There are two ways in which one can implement help facility is TURBO PROLOG programs

1. Using editmsg a standard predicate we can create a help file that will permit the domains expert to deal with the expert system. The domain expert needs to press F1 for help when the program is being eecuted. The editmsg predicate has eight arguments that permit us to call the editor, write a string to the editor window. Edit that string, insert a left or right string in the text header, move the cursor and display a message at the bottom of the editor window. The form fo editmsg. predicate is

editmsg (Inputstring, Outputstring, Leftheader, Righteader, Message, Postion, Helpfilename, code).

2. In this method we use a file_str predicate: which is of the form :

File_str (Filename, Outputstring).

This predicate reads up to 64k of characters from Filename to the string Outputstring until the end of the file is recived this tring can be displayed using the display predicate.

for example

help :-

```
file_str ("Production.help" Help),
display (Help).
```

CHAPTER 3

DESIGN OF EXPERT SYSTEMS

3.1 METHODOLOGY

3.1.1 The need for a methodology :

The prevailing view that developing A.I. systems is the prerogative of the pioneering A.I. scientists has resulted in companies spending on the subject without appropriate strategies and planning.

Like any other project, A.I. projects also need the same consideration such as specifications, budget and timing.

Structural system development techniques offers a more appropriate approach to system development. These techniques are now practised widely in data processing shops and lend themselves readily to specifications and implementation of A.I. systems.

Every A.I. system project consists of pieces that can be done using traditional D.P. techniques which can be called the D.P. components and pieces that require A.I.

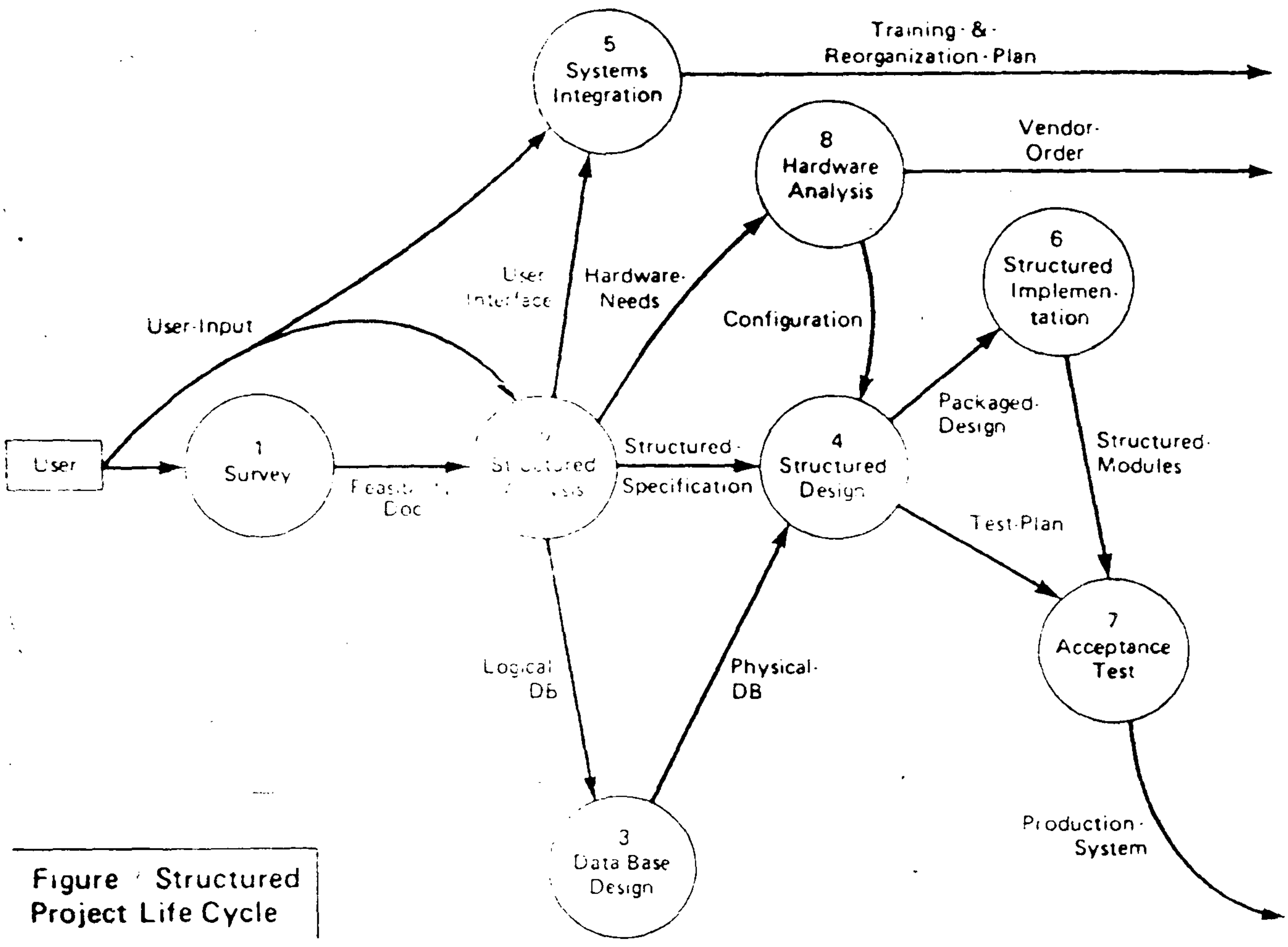
in some form (A.I. components). The structured methodology helps us separate A.I. and D.P. components of project, allowing each component to be developed so that the two can be integrated.

3.2.0 THE ELEMENTS OF STRUCTURAL DESIGN

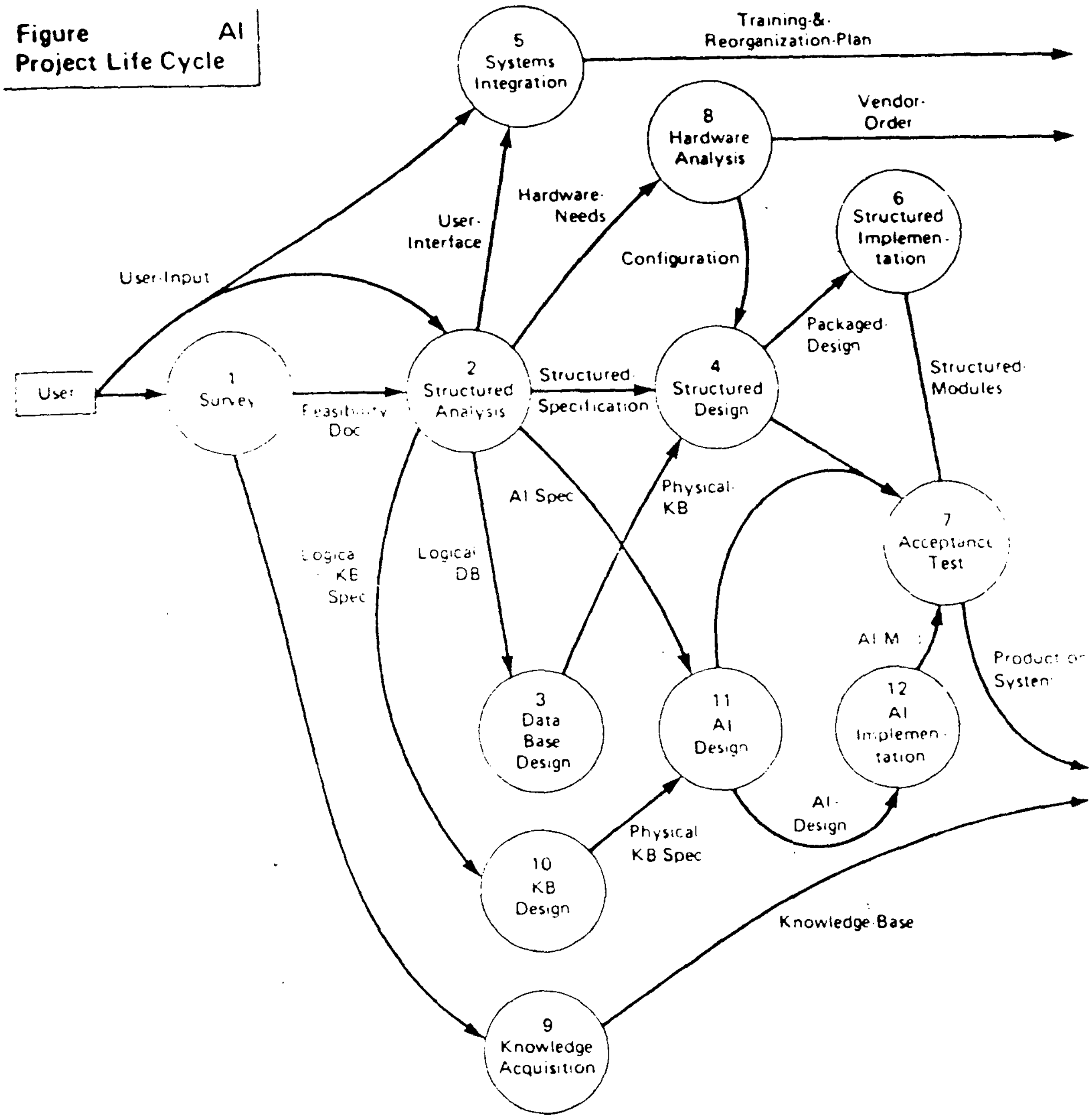
Prof Robert Keller has suggested an approach to the structural development of projects that is similar to the traditional data processing projects. These are four major activities convered under (1) Survey (2) Structural Analysis (3) Structurel Designs (4) Structure Implementation. Although they are interrelated they can be funded separately and can be understood clearly in terms of business.

The structural analysis parts result in a network diagram with narative specifications that represent the well defind parts of a users needs.

The structural design phase produces hierarchical charts which generally show well defined functional program modules called by well defined control programms with well



DFD Figure Structured Project Life Cycle



DFD Figure AI Project Life Cycle

i) A narrative overview of the relevance of K.B. systems with highlevel data flow diagrams showing K.B. systems with their interfaces to the rest of the company's environment.

ii) A payoff/cost analysis of the applications of AI to the domains.

iii) A risk analysis of the proposed development.

3.2.2 Prototyping

Having done the domain selections, a prototype of the application must be designed.

The prototype must roughly define the scope and output. Evolving the knowledge base to an expert state begins with the survey and continues through out the project and beyond.

The process of building a knowledge base begins and continues by watching the expert performing his/her job. Listing the decisions factors and preparing a rough rule base (often called pre-prototyping) is the next step.

During both survey and analysis the principle development planners are the knowledge engineers and the users (also called domain experts). It the user of MYCIN is a qualified medical practioner he can be called a domain expert.

3.2.3 Objectives

One way to begin is by characterising the organisational setting within which the expert system is used. There should be a clear statement of the expert systems purpose within that setting.

Q. What kinds of problem will it solve?

Q. Who will use it?

Q. When will they use it ?

Q. What kind of knowledge will the expert user be able to furnish to the expert system during consultation?

Q. How fast a typical consultation be if it is to be of any value to the user?

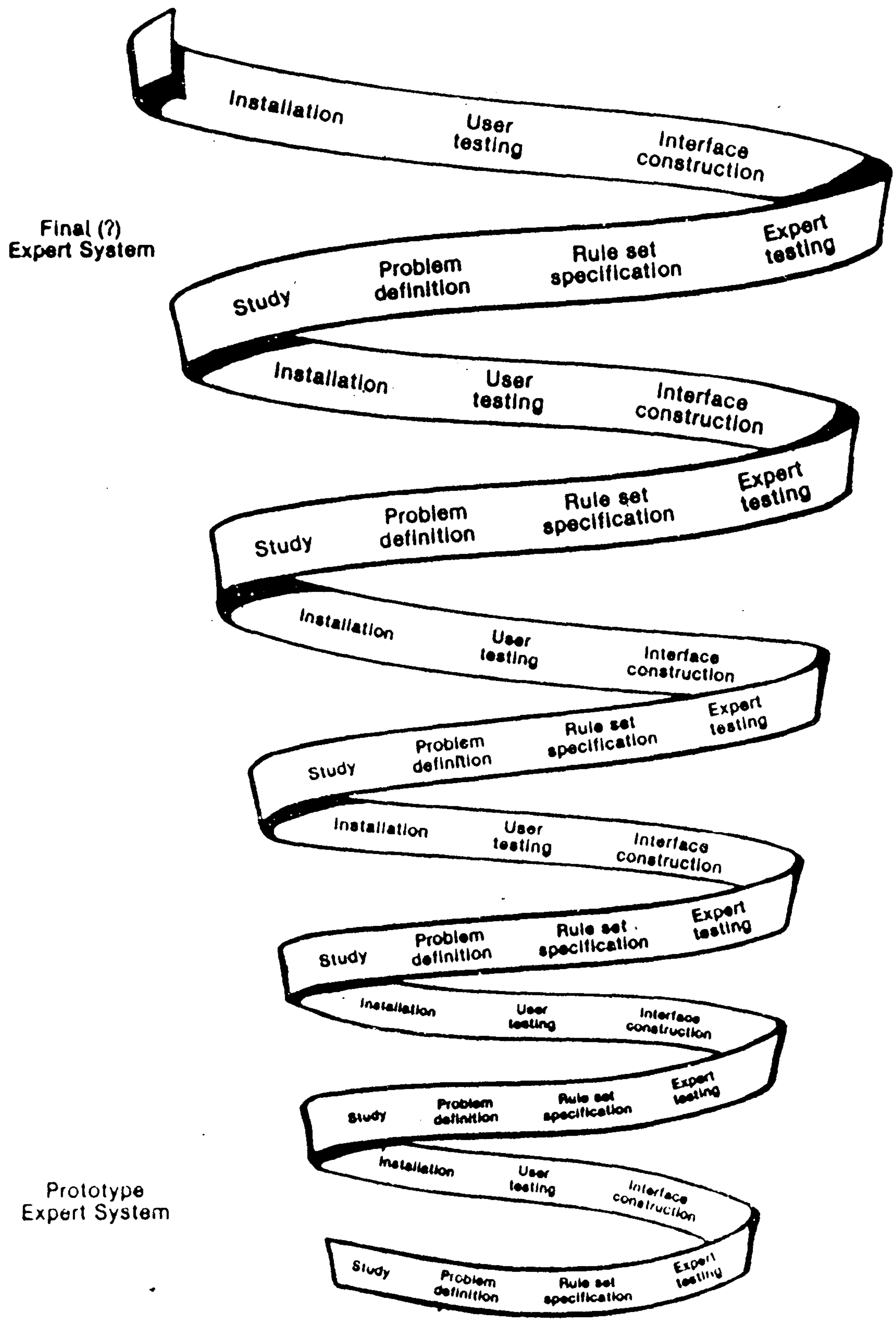
Q. What style of user interface is appropriate?

Q. How good must the expert system advise be?

Q. Why is the expert system being constructed?

Q. What is its expected benefits?

The development spiral



Q. In view of the benefits, how much is it reseanable to spend for the expect system's constrution and ongoing implementation?

Q. When must it be ready for operation.

All these question will then be the basis for planning and later, revaluation.

3.2.4 Planning

- It identifies resourses that are needed to construct
- Prescribe the sequeence of action
- Establishes budgets.

Larger problems typically require more effort in planning.

3.3.0 STRUCTURED ANALYSIS

One an application has been selected it is time to define completely the user requirements of the finished system. The stated purpose of an expert system project is to convert a human expert system into a computer expert system. To specify this conversion the techniques structured systems analysis are very effective.

defined calling sequences. Fig.3.2 Data Flow Diagrams for A.I.

There are some major difference between the traditional structional life cycle and an A.I. life cycle. The way the A.I. components are conducted, the people involved and the hardware is different from the traditional D.P. components.

The following are the major activities of the development process.

3.2.1 Survey

The main purpose of survey is to decide about the feasibility of the given project. Contents of the general goods and cost estimates.

Before going in for a knowledge based project one must decide on where and how to fit it in the current environment. The survey should identify the potential application domains and for each of the domains a report should be pre-pared, containing the following:

The primary outputs of an A.I. structured system analysis are (1) structured specification of the system and (2) a prototype knowledge base.

3.4.0 STRUCTURED SPECIFICATION

1. The structured specification is a document which represents the final stage of an analysied process in which we examine how the human expert divides the complex task to minitasks and decisions finally the heuristics involved in the atomic level.

The specification of the current system involves the following steps.

Step 1

Complete unambiguaous, and non-redundant documentation of what is being done now by the human experts. This is the documentation of the current physical system.

Step 2

In this step the physical system is distilled to get explicit decision and planning functions the expert is doing and the specific information used to perform those functions. This will result in the current functional system.

This is an important step since the real goal in developing the expert system is to presence the functional essence of what the human expert does.

When we are reasonably sure of the current functional system we can move on to specify the new system. This also involves a two stage process.

Step 1

To add any new functionality or data which may be needed in the new system. We will not consider any technologies constraints. The result of this stage is the new functional system.

Step 2

To specify the physical details of the new system and create documentation of the new physical system. At this stage the technological considerations prop-up. Whether the system will operate interactively or in batch, whether the processing is centralized or distributed. We now have a structured specification of the knowledge based system.

3.5.0. PROTOTYPING

The ability to create approximations of solution-prototypes is particularly important in an expert system project both because knowledge base creation is a highly interactive process and becauase the only proof we have for correctness of knowledge base is in its ability to make valid expert decisions.

There are now many domain independent expert system shells i.e. generalized inference engines-which are excellent for atleast begining the construction of a complex knowledge base. Such systems have the ability to reason and make inferences intellegently. They may or maynot be used in the final version of the system but they can be very useful in the begining .

Appendix: Commercially avaiable expert system shells.

A useful approach to rules set specifications would be to start with

1. Indentification of variables
2. Variable specification
3. Discovering rules based on inter relationship between variables
4. Goal specifications

The main outcome ofthe structured systems analysis is also similar.

Variable specification is a systematic classification of the given variable. The specifications must contain the following :

1. Variable type
   Numeric/string/logical
2. Size in case of string variable
3. Whether certainty factor involved.
4. Maximum number of values it can have in an inference.
5 Source-User/Database/Model/during consultation/Statistical/Assigned
6. Initial Status (before reasoning begins) Unknown/Known.
7. Depends on the following variables........

A bubble diagram describing the inter relationship between the variables must be drawn. The diagram should necessarily converge on the goal variable.

Every inter relationship should be defined as a rule for the knowledge base.

Having developed the rule base it should be subjected to expert testing. The software GURU gives the HOW and WHY commands. Using these commands the developer and tester can trace the line of reasoning so that if there is any disagreement between the experts opinion and output then it can be traced back.

The expert must know which rule has been fired and which rule has not been fired. He must also know what values of variable helped the rule get fired. The "trace" Window of PROLOG. is a useful faculity in his regard.

3.6.0 APPLICAION SELECTION

The enthusiasm in A.I. and for expert system technoloy is no focussed for applications in the business & industry. But looking for applications of expert system technology is the wrong approach. The right attitude is to identify problems that need solutions and then if it makes sense, to apply expert system technology in solving those problems.

In fact, virtually any traditional, business application can probably be improved by the selective introductions of heuristic processing.

These factors we consider for selecting knowledge based products are same as those we would evaluate in choosing any system project.

3.6.1 Worth

There should be some kind of a pay off, which is frequently messured in terms of profitability. With knowledge based systems the profitability involved are different from the traditional systems.

In general we are looking for projects in which a high pay off will result from increases in the speed of and the reliabiliy with which the tasks are performed, or from the ability to perform the tasks correctly. K.B.

applications have higher pay offs than traditional applications. One reason for this is that knowledge problems that were addressed by highly paid professionals.

3.6.2 Risk

Risk means the likelyhood of not being able to complete the project to a point when the results in terms of payoff, justify the cost to date. Risk is more the result of several other factors than a factor in itself.

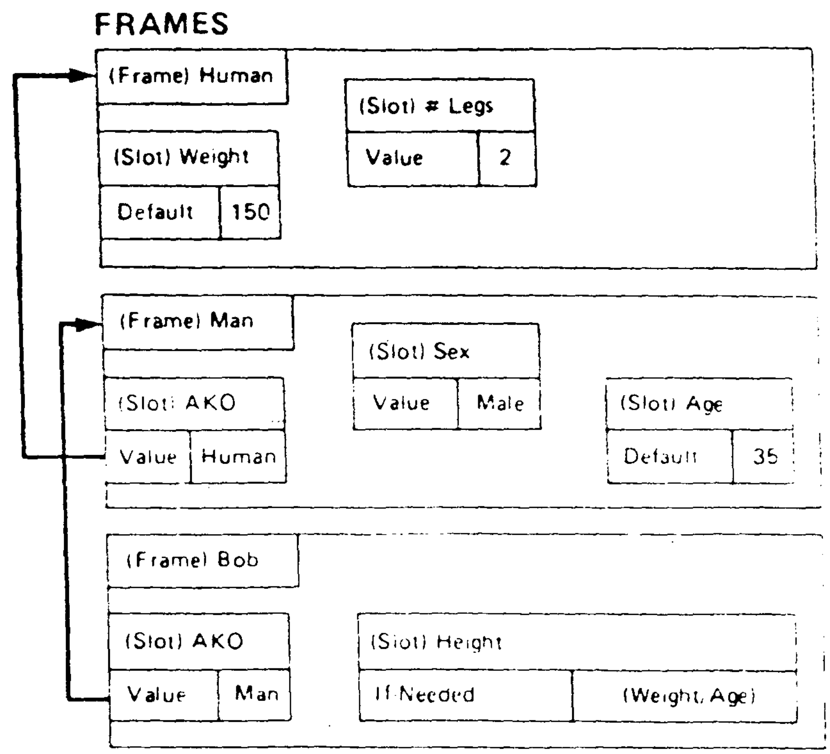
The list of potential industrial applications for expert systems is very long. The suitability of any of these areas depends partly on factors having to do with the nature of the knowledge involved, its reliability, completeness, ambiguity and stability.

In the ideal situation we would like to have an application in which knowledge in restricted to a very harrow domain and decisions are entirely determined by messurable factors that don't change over time. The real life expert faces a different situation which involves considerable risk. The risk factor should be a minimum while choosing an application.

3.7.0 FRAME BASED SYSTEM

Some of the most popular techniques of knowledge representation are production rules, AND-OR Trees, and Knowledge vectors. We wish to extend those simple paradigms to include a richer assortment of knowledge and then, using,PROLOG, develop a set of elementary access routines for a frame knowledge based management system.

FRAME :- A frame is a data-structure that can be used to represent any real or imaginary stereo-typical entity. Frames can be put together in sequence to represent a changing situation (like a movei)or linked hierarchically or in network fashion. Each frame is roughly equivalent to what we think of as a record in a traditional data base. A system of frames is roughly equivalent to a database in which many types of logical records are linked by perhaps several pointers. The difference between a frame and a data-base record is in the kind of information we find and how it is stored and used. In a data base record, each field has a fixed format with a defined type of information whereas in a frame it may not be so. A frame base has all the capability of a relational data-base. Like data base systems, a frame system should also possess access methods



A Hierarchical Example

and routines for modification, addition, and deletion. Considering the data structure of frame, List Procesing can be used conveniently for representing frames. In order to implement a frame system structure described above, we need to represent four things in each frame.

Name of the frame, slots, type of information, and value. Since each frame how a simple identifier name and possible many slots, one way to represent each slot is with a list. (The first element of the list well be the name of the frame, and each element following the name will represent a slot description).

Describing a frame structure notation is quite easy. This characteristic of using similar patterns at many levels of list structure is an important guideline to keep in mind while designing a list structure. Frame Structure representation of knowledge is a very flexiible and powerful medium. It can be suitably exploited for developing industrial expert systems.

3.8.0 LANGUAGES AND TOOLS

Traditionally, LISP has been used as a language for knowledge base applications., Several GPS models have been tried through LISP. PROLOG is more attractive for AI applications for several reasons. The advantages of PROLOG over LISP cannot be discussed as they are out of context. More than languages, a breed of software known as expert system development tools have emerged. These tools provide quick and easy ways of developing prototypes and are domain independent. One of the earliest such tools was EMYCIN.

EMYCIN has a MYCIN style knowledge system. Considerably more general tools for developing expert system are OPS 5 and ROSIE interpreters. These are problem processors that use forward reasoning to solve proble3ms. Jo build an expert system with such tools the developer creates a progiam consisting of statements desoribing a flow of reasoning to be followed when solving a problem though this involves considerably less effort then programing an expert syste,m from stratch in LISP and PROLOG languages. OPS 5 interpreter. In OPS 5, each rule has a lefthand side composed of a set of condition elements and a rigjhthand side that specifies action. A condition element is a pattern containing both constants and variables (denoted by names within angle brackets [^{2} and ^{3}]).

Expert system development tools

Tool	*Type*	*Vendor*	*Host machine requirements*	*Cost*
KEE	Stand-alone	IntelliCorp	Symbolics (and other) LISP machines	$60,000
KES	Stand-alone	Software A&E, Inc.	Symbolics (and other) LISP machines, IBM® XT™, VAX 11/780	$4,000–$23,500
M.1	Stand-alone	Teknowledge	IBM Personal Computer, XT™, AT™ (PC-DOS™)	$10,000
ART	Stand-alone	Inference Corp.	Symbolics (and other) LISP machines, VAX 11/780 (equipped with VAX-LISP)	$50,000
Personal Consultant	Stand-alone	Texas Instruments Incorporated	TI Professional Computer (768K RAM)	$3,000
TIMM	Stand-alone	General Research Corp.	IBM XT, VAX 11/780	$9,500–$39,500
RuleMaster	Stand-alone	Radian Corp.	UNIX™ operating system	$5,000–$15,000
Guru	Integral	MDBS, Inc.	IBM Personal Computer, XT, AT (PC-DOS) 16-bit machines with MS-DOS®, VAX 11/780	$2,995–$30,000

Note: Host machine requirements and prices subject to change.

The OPS 5 interpreter cyclically pertorms a recognition-action loop, during which the condition elements are matched to the working memory elements that have the same name to working memory elements that have the same attribute value. When all the condition elements in a rule are satisfied, the production is instantiated and enters the conflict set.

The OPS 5 interpreter selects for execution one instantiation of this set with a user-specifiable strategy. There are several tools that have become commercially available. They include systems such as KEE, KES, M.1, ART, Personal consultant, See table 1.3 bes TIMM and Rule Master.

With the introduction of GURU, a new kind of software tool that integrates expert system technology with main stream data processing methods. It provides facilities for transfering data between the expert system and external software packages. GURU actually incorporates full-scale database management spread-sheet structured programming language and other capabilities. This unified environment is singularly suitable for industrial applications.

Many computer hardware manufacturers and marketing a new class of computer called an A.I. work station or a LISP machine. This category of computer is unique in that its internal logic and memory structures are desiogned to efficiently execute programs written in LISP. Althougjh LISP itself is a higher level language, for those machines, LISP is equivalent to assembly language on the more traditional computers. LISP machines were developed in response to LISP Status for 30 years as the defacto standard programing language for A.I. research work.

3.9.0 PROLOG AND EXPERT SYSTEM

PROLOG is the first seriously popular, powerful language that embodies the approach of logic programming. PROLOG was designed to allow the programmer to specify a problem through logic allowing explicit expressions of facts, assumptions and goals rether than specific instructions to the computer on how the problem solution should be reached. PROLOG is an object oriented language. A majority of the expert systems currently being developed is in PROLOG.

CLOCKSIN AND MELLISH PROLOG or Common PROLOG is a very powerful version of PROLOG. But the PC based Turbo PROLOG is also handy for developing expert systems. It can accommodate a host of knowledge representation methods made of different data structures. Turbo PROLOG allows for descibing global and local ddomains and global and local predicates. This schema is useful in stuctured (modular) programing. It provides facility for maintaining dynamic databases internal goals and compiling directives.

PROLOG is highly suited for rapid prototyping. To this end, before we fully appreciate the uniqueness of PROLOG, it is important to look at the requirements imposed by the physical systems and models.

The D.F.D.'S of the new physical models which appear in the structured specifications contain a large number of mini systems each of which has a mini specification written for it. These mini specifications however tell us only what the function does to convert its inputs into outputs not how to write procedural program which will actually do the transformation.

Clearly what is needed is a programming language which fairly directly implements equivalents of the structured english specification used during the structured analysis and this is what PROLOG does. The mini expert systems designed so far on Turbo PROLOG are known to be able to handle complex knowledge structures. It has a general purpose global inference engine. Turbo prolog has provided for windowing and frame based knowledge representation.

Requesting BIOS Services Turbo PROLOG'S BIOS predicates is used for accessing BIOS services. It helps call interrupt services with great speed. Normally, one would have tried to interface assembly language (INT function) with turbo PROLOG. In doeing this, one has to take great care in saving flags and registers towards protecting the main program environment while calling the interrupt.

The Turbo predicate BIOS does all that. All one has to do is to specify the interrupt services one wants to access, required input parameters, required output parameters and variables in which it can return the result of the interrupt call.

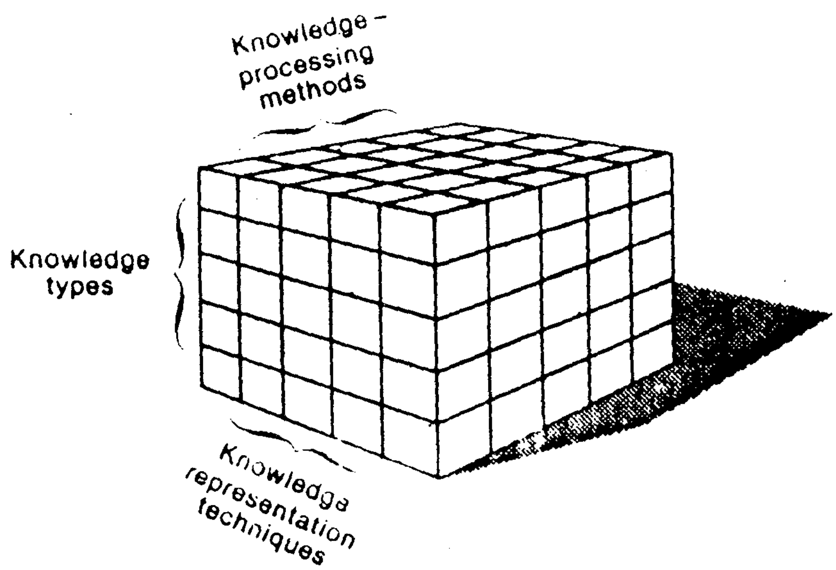
3.10.0 KNOWLEDGE-BASED INFORMATION CENTRE

We are rapidly entering into an era that will be dominated by knowledge based organizations. Each such organization will be populated by a society of knowledge workers who are interconnected by a computer based infrastructure. Each knowledge worker will be an expert in carrying out certain kinds of knowledge management activities. The workers knowledge processing efforts will be supported by AI work station within the industry. Figure. Activities of a knowledge worker .

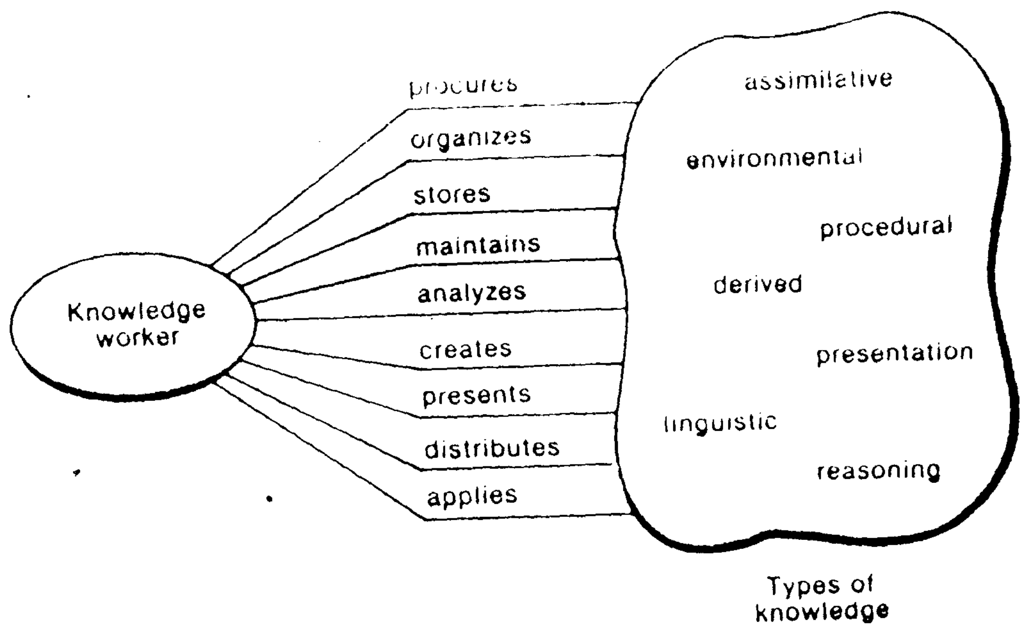
A knowledge based information center is a central facility within a company that supports a wide variety of tools and application for artificial intelligence. Typically, today's information centre whether centralized or distributed consists of a handful of not very user friendly poorly integrated data based systems for traditional applications.

The knowledge based information system should support expert system product and home-growth expert and house a broad base of end user. It should support a variety

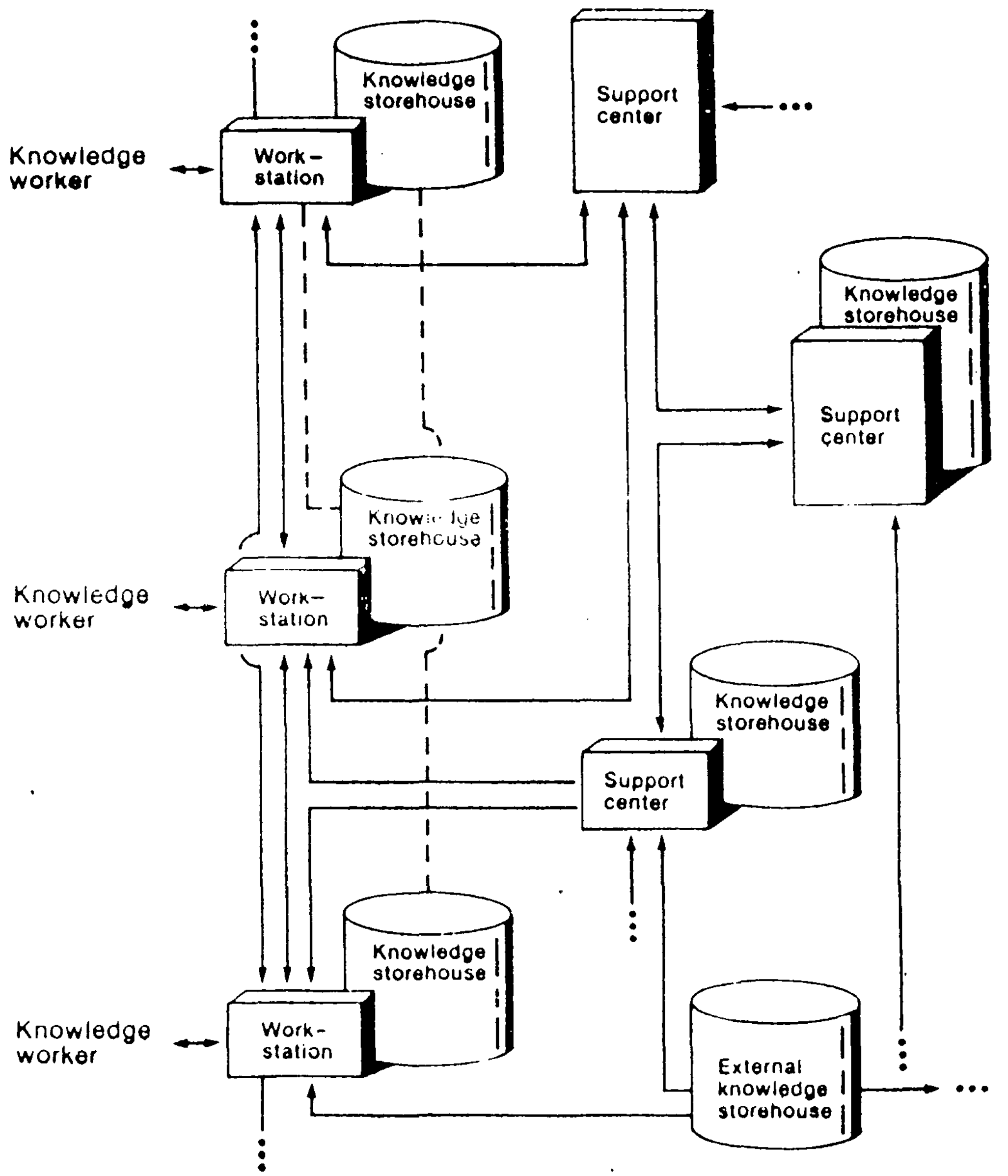
Knowledge management possibilities



Activities of a knowledge worker



Sample infrastructure for a knowledge-based organization



of tools that are oriented toward being expert system shells. It should support atleast one sophisticated software and handware knowledge engineering environment.

Every K.N.B.I.C faculity should be optionally available through a natural language interface.

CHAPTER 4

CONCLUSIONS

The past few years have witnessed rapid advancement in the field of Artificial Intelligence and expert system technology. As a result, expert system technologies have become available for application in several fields. An important area of application is the industry, which employs a variety of experts who take critical decisions involving complicated reasoning. Theoritically, every variety of expertise can be captured into an expert system.

Industrial systems have certain peculiar characteristics and requirements. The Industrial experts systems, therefore, need development tools and design and implementation methodologies that suit these requirements. For example, numeric data-processing and procedural modelling are inherent characteristics of most industrial systems. The development tools for industrial expert systems must provide for such processing.

Although, expert systems technology is still in its infancy, several standards have been laid down. The methodological problems of design and implementation have been studied in great detail and many tools have been developed and used. The popular variety of tools called the expert system shells are extremely useful for rapid prototyping. Many of them, however, lack the generality needed for Industrial Expert Systems.

The choice of PROLOG for rapid prototyping is quite attractive. A PC environment is suitable for this purpose mainly due to its availability. TURBO PROLOG was therefore choosen to develop prototypes for two domains (application areas), the selection of which, was result of rigourous considerations. The domains taken up for this project are :

1. QUALITY ASSURANCE EXPERT SYSTEM.

2. PRODUCTIVITY EVALUATION AND DIAGNOSIS.

These two domains present immense challenge due to the necessity of data-processing that goes side-by-side with their respective knowledge bases. It was necessary to use routines of other languages from within or outside TURBO PROLOG modules. To this end, several techniques were explored for implementation. The techniques range from the usage of ROM - BIOS and DOS interrupts to special features of TURBO PASCAL, TURBO PROLOG etc. TURBO PROLOG however, has certain glaring weaknesses which cripple the efficiency of the programs.

Industrial Expert Systems consist of knowledge systems rather than merely rule bases. Knowledge Systems include rule-bases, data - bases, procedural models and various types of I/O facilities. The above mentioned domains were gathered and compiled painstakingly after a series of direct interactions, with the experts in the industries.

The methodologies used to develop the two systems are different from each other. To develop the Quality Assurance Expert System, structured analysis, structured specification, structure design and implementation methods were used. This was inspired by Prof. Robert Kellers' papers and books on the subject. Productivity Evaluation and Diagnosis was developed using methods given in papers by C.W. Holsapple, A. G. Whinston etc. Both methods converge on several points and issues.

The key to success in Industrial Expert System lies in proper software integration between the AI and the data processing components of the given knowledge system.

The solution lies in a new kind of development tool, based on the DSS concept of a generalised problem processor.

In the process of developing the prototypes of the two expert systems we have encountered several difficulties and problems. Some of them are stated below.

1. The unavailability of commercial expert system shells and operating environments .

2. The unavailiability of expertise in several fields.

3. TURBO PROLOG doesn't lend itself well to numerical calculation. This difficulty was overcome. The result was a sort of library containing programs in TURBO PROLOG for arithmetic calculations.

4. The predicates in the programs were too many to handle. This problem was solved using modular programming.

5. For field level implementation, accurate field data was not available. Therefore, approximations had to be made during sessions with the expert systems.

6. Certain interfacing techniques between PROLOG and other languages were developed.

APPENDIX

APPENDIX A

A.1 QUALITY ASSURANCE SYSTEM

Large manufacturing concerns rely heavily on small manufacturers for the supply of components. It is convenient for them mainly because, the necessity of additional investment is obviated. Thus, the purchasing concern awards the contract to a suitable supplier, provided, terms of the big concern are met by the suppliers. Quality is one of the important components of the contract. Prior to awarding the contract, the concern inspects the facilities of the supplier. It carries out a survey called the Quality Assurance system survey. As the name suggests, the quality assurance system of the supplier is examined in detail by the survey.

After the survey, the report containing the approval or disapproval (including diagnostic suggestions) is given to the contractor/supplier.

OBJECTIVES : The software thus aims at evaluating the quality assurance facilities. This precedes the diagnosis and suggest for improvement. Evaluation involves :

1. The examination of individual aspect of the facilities.

2. Grading on these aspects is to be done interactively. The guidance for grading is provided by the software.

3. An assessment of these grades is doen aggregation of the individual groups. A, B, C etc.

Suggestion/advice are general and specific.

General suggestions are those suggestion which are derived from grade aggregation.

Specific suggestions are those which are derived from the ascertained facts.

A.2 RULE BASE

Please note that this rule base is only an approximate one.Its contents have been highly simplified and are not to be taken seriously.

FOR QUALITY ASSURANCE

Rule 1.

If $A \leq 60$ then

Then Quality control Organization

Leaves a lot to be desired.

Rule 2.

If $A_1 \leq 6$

AND A [illegible] 70

THEN you do not have manpower which is qualified an QC matters. There are too few inspectors taking care of too much of production.

Rule 3

IF PROCR $\leq$ 14,

AND QCMETH $\leq$ 70

AND RECORD $\leq$ 35,

THEN, you neither have proper mathodology for testing and quality control nor do you have the procedures for the same properly recorded. you are advice to review the procedures

Rule 4

IF EQPT $<$ 55

LABAN $<$ 15

THEN, your are short on equipment you have to purchase new equipment for quality control. At the same time you have to update the testing procedures, and documentation of the desting done on QC Equipment.

Rule 5

IF POSEG $<$ 40

AND FLAY $<$ 50

THEN Subseqent to your bit of rethinking on layout you have to take care of positive segregation.

Rule 6

If EQPT $<$ 55

THEN you do not have enough equipment. It is suggested that your puchases more equipment (preferably with specification).

Rule 7

IF EQPT $<$ 55

AND TESTIEQ $<$ 20

THEN your are supposed to list your equipment related to quality control with specified periodicity.

The table represents the subject-wise dustribution of the variables. For example the organisational ability of the firm is reflected in the values of the variables A_1, A_7, B_{12} and D_1

VARIABLE	DEPENDENT VARIABLES
Organisation	A_1 A_7 B_{12} D_1
Written Procedures, & Manuals	A_2 E_2
Instruction sheets	C_1 C_2 D_3 E_2
Equipment	A_3 B_5 C_4 C_7 C_{15} D_2 D_4 E_4
Facilities layout Lighting etc.	A_4 B8 C_{12} C_{16} C_{17} E_1 E_{14}
Communkcation of inspection reports	A_5 B_{10} E_{13}
Investigation system	A_6 C_{13} E_{10}
Records & documentation	B_2 B_4 B_7 C_5 C_6 C_8 C_{16} E_5 E_8

VARIABLE	DEPENDENT VARIABLES
Laboratory analysis	B_4 D_6
Testing of Equipment	B_6 D_5 E_5
Drawing & specification	C_3 D_4 E_3
Positive segregation	B_{11} B_{13} C_{11} C_{12} E_7 E_{12}
QC Methodology	C_9 C_8 C_{18} C_{14} D_7 D_9 E_1 E_6 E_9 E_{11}

APPENDIX B

B.1 PRODUCTIVITY EVALUATION & DIAGNOSIS

A firm owns several machine tools which are used in production. The management describes the job to the operators incuding the time within which it must be done.

The company invests (1) On the plant (2) In the machine tools (3) In the workmen (4) On the raw materials etc.

All these assets must be productively used. In order to know whether they have been productively used or not we must have certain standards with which to measure productivity.

For this we need to understand the process of production. It starts with the sales forecast. The production targets are fixed. A schedule for production is prepared. Acording to this schedule the jobs are dispatched. The process of production (supplied by methods dept). the material (job) (is supplied by materials dept) and the standard time is supplied by IED.

A work order containing the details of the processes to be performed (the relevant parts reproduced form the process sheets), is issued to the operator. The performs the operation under the supervisor. The supervisor has an account of the machine-hours looked, the operators machines and their relevant data. The inspectors responsible for quality control keep an account of the rejection rates. All these details form an input from productivity calculations.

These calculations are based on formulae known to objectively determine the producctivity of the machine tool. A machine tool must be utilized productively in order to make operations profitable.

Experts assign minimum values to different ratios, below which all operations are unprofitable..

PRODUCTIVITY ANALYSIS

Productivity is a ration of resources generated to the resources consumed in producc;ting them, both being measured in same units. It is usually expressed as an index in which the current position is compared with the position during a base period.

Productivity is a useful indicator of the economic efficiency of an industrial unit. The simplest type of productivity is labour productivity which is output per man hour. But such a measurement is not of much diagnostic value. An index which takes into consideration all the key factors influencing productivity is the most suitable index. The procedural model chosen for this expert system is described as below : -

The indeces to be considered in this model are :

1. Machine utilization ratio
2. Operator utilization
3. Machine hour rating

A detailed analysis with model of the reasons for low productivity leads to the diagnosis (i.e. determining the cause). Subsequently correct actions are taken for improvement. Machinery can underperform due to a varity of reasons. This is where the expert comes into play.

For example if any of the four are not optimumly performed, there could be a drop in productivity :

- M/C Maintaince
- fixture
- tooling
- setting time

Worker absentism, alcholism or any other lapse on the operator's part could also undermine productivity.

The expert analyses all the aspects and prescribes corrective action.

SOURCES OF INFORMATION

The machines and operations are considered as units for the purpose of analysis and diagnosis. Every machine is booked for operation for a certain period of time. A M/C time table log book is maintained for recording the activities of the machine.

The other sources of information are (A) Scarp record (B) Quality control report (these reports give the m/c wise rejection rates). (C) The absentism report (pay-wise) from the personnel dept/ppc dept. (D) Tool stock card/Tool register.

RULE BASE

Rule 1.

IF. PRODUCTIVITY $<$ 0.70.

THEN KOOK IS TRUE

Rule 2.

IF ; PDTY $>$ 0.70.

THEN ; OKAY = TRUE

Reason : Since productivity is low we must start with the diagnosis to determine the reason for the slump in productivity. OKAY will be fired.

Rule 3

IF ; HRSREG $<$ 8 X 0.75 X TOTAL DAYS

THEN : NONPLAN = TRUE

NONPLAN : - Message : The hours booked for production is adequate. Directives must be sent to the PPC department. An explanation should be asked for.

Reason : The houss registered should be 8 per shift. If total hours registered for production are less than three-fourths of the available hous then it should be considered as in-adequate planning.

Rule 4 :

IF : TOT HRS/DAYS < 1.0

AND NONPLAN FALSE

THEN : SHIFT = TRUE.

SHIFT : - Message : The shift incharge has not put the men on duty at the right time. He should be asked for explanation.

Reason : If the slump in production is not due to the lack of planning (NONPLAN = FALSE) but the hours registered per day are still poor, then the person incharge of the shift is responsible.

Rule 5 :

IF : SHIFT = FALSE

AND NO PLAN = FALSE

AND MSGHR/TOTDAYS < 2

THEN : IDLE = TRUE

IDLE : - The hours for which the workers booked were idle is on the higher side. If warning had already been given last month then mobilize the labour relation cells to deal with it.

Reason : If neither the PPC department nor the shift incharge are at fault and if the hous booked which were idle for al workers in the shift, is greater than 2 hrs per worker per shift then it is a problem to be dealt with initially by the shift incharge and later by the personnel and labour welfare cell.

Rule 6 :

IF IR = TRUE

THEN : IPRO = TRUE

IPRO : - Message : It is necessary to tackle the IR problem present in the area concerned very urgently. It should be higher on the priority of the management.

Reason : - Obviously an IR problem should receive higher priority attention.

Rule 7 :

IF : NONPRO/TOTHRS $<$ 0.15

THEN : TRIP/TRUE

Reason : The non-productive hours other than absenteesm and lack of planning are due to several reasons summed up by the predicate Trip (Non-tool, breakdown, set up, etc).

Rule 8 :

IF : ELEC/TOTDAYS $\leq$ 0.5

THEN : ELEF = TRUE

ELEF : - Message : - The slump in productivity could be due to frequenty electricity failure. concerned officials of the electricity board should be informed .

Reason : Lack of production due to elecctricity failure should be indicated.

Rule 9 :

IF SET UP/TOTHRS $\leq$ 0.05

THEN SET = TRUE

SET : - Mesage : Apparently, the set up time is high. This could be due to either the inexperience of the operator or to the lack of proper fixture. It is advised that the shift incharge should enquire into the problem and decide on further course of action.

Reason : The high set-up time clearly indicates in the premise and it can best be set up right by the shift incharge.

TOOLING PROBLEM

If NOTOOL 3% or rejection rate y % tool problem is sever.

1. If chatter after tool-grinding or time on the job or job becomes oval then tool grinding is not alright. Inform the tool crib manager. Remind the operator about the tool specifications.

2. If chatter on jonb etc. and tool grinding are alright then the problem is holding (chucks) vibration problem or spindle alignment problem or setting problem. Inform the maintainance department. Direct the attention of the shop floor manager to the erring operator.

3. If tools are not available in main the storage and alternate tools also not available then place an urgent telex to the regular supplier. (MATTER MOST URGENT EXPEDITE, etc.) Send the purchase officer to the tool crib and find out the position and direct him to take further necessary actions. Ask for a report of current stock position data.

4. If tool not availabel in main stor are and regrinding cannot be done within 15 minutes then direct the manager to release alternate tool. The manager must make a note of the same in the appropriate remark column of the tool register.

Commercial natural language systems

Natural language system	*Vendor*	*Host machine*	*Type*	*Data sources*	*Graphics supported*
Clout	Microrim	Micro	Stand-alone	R:base	None
Guru®	MDBS, Inc.	Mini, micro	Integral	Guru data bases	Bar, pie, line, etc.
INTELLECT	Artificial Intelligence Corporation	Mainframe	Stand-alone	IDMS, ADABAS, Vsam, FOCUS	Bar
K-Chat	MDBS, Inc.	Mini, micro	Integral	KnowledgeMan data bases	Bar, pie, line, etc.
RAMIS II English	Mathematica Products Group, Inc.	Mainframe	Stand-alone	RAMIS II	None
THEMIS	Frey Associates	Mini	Stand-alone	Datatrieve, DBMS-32	None

SOURCE LISTING AND SESSIONS

```
        /* module Ø  */

domains

predicates

  gool

clauses

   gool:-

    makewindow(1,7,7,"Timer",8,1Ø,12,6Ø),

    time(Ø,Ø,Ø,Ø),system("dir a:"),

    time(H,M,S,Hundredths),

    write(H," hours  "),

    write(M," minutes  "),

    write(S," seconds  "),

   write(Hundredths," hundredths of a second"),nl,nl.
```

```
/* OPENING MODULE */
domains
predicates
  goe
 clauses
  goe:-
   graphics(1,1,4),
   write("PRODUCTIVITY EVALUATION AND DIAGNOSIS"),
   readchar(_),

    graphics(1,1,4),
    write("THIS WILL HELP YOU SET RIGHT"),
    write("YOUR PRODUCTIVITY PROBLEMS"),
   readchar(_),
   text,
    write( Press space bar).
```

```
/*            THIS PART OF PROSPER AIMS TO EVALUATE */
/*            THE  PRODUCTIVITY PERFORMANCE OF A */
/*              COMPANY PRODUCING ENGINEERING GOODS */
/*  WITH THE HELP OF THIS MODULE THE DOMAIN EXPERT */
/* CAN INTRODUCE HIMSELF AND HIS FIRM   */

    domains
       name,area,opgroup=string
       Totdays=integer
       file=tod
    database
    predicates
         go

         title
         messg
         enquir
```

```
clauses
    go:-

     makewindow( 1,7,7,"",0,0,25,80),

     shiftwindow(1),clearwindow,

     title,

     messg,

     enquir,

     nl.


      title:-

        write(" WELCOME   TO      PROSPER     "),NL,write(""),nl,

        write(""),nl,write(""),nl,write(""),nl,write(""),nl.

    messg:-
      write("  Are you concerned about the produtivity of your firm?"),nl

      write("If your answer is yes, then read the following"),nl,

      write(""),nl,write(""),nl,write(""),nl,

      write("This software will evaluate, monitor and diagnose"),nl,

      write("productivity of labour and machinery in your firm.") ,nl.
```

```
write("Please write the name of your firm."),nl,

readln(Name),nl,
write(Name),nl,

write("You have to mention the specific shop floor area "),nl,

write("for which you want to assess productivity. It is"),nl,

write("advised that you take a set of machinery as the "),nl,

write("unit."),nl,

write( "The area under study is ____________________"),nl,

readln(Area),nl,write(Area),nl,

write("The area under consideration is,Area,"),nl,

write("What is the name of the operation group you wish to study ?"
),nl,

readln(Opgroup),

write(Opgroup),

write("What is the period for which you want to do the study ?"),nl

readint(Totdays),

write(Totdays),

Totdays=Totdays*1,

openwrite(tod,"td.dat"),

writedevice(tod),

closefile(tod).
```

```
/*    MODULE FOR CALCULATION AND DATABASE GENERATION    */


global domains

  noprod       = p(setup,notool,elec,brkdwn)

   work        = q(Stdjph,Pieces,Hrsreg,Ophrs)

   geo         = r(Chat,Oval,Line)

   toole       = s(Stock,topis,alter)

   miss        =t(Msghrs,Ir)

        nonpro,setup,notool,elec,brkdwn,

        Stdjph,Pieces,Hrsreg,Ophrs,msghrs,

        Our,Oput,Mur,Pdty  =real

        alter,ir,Chat,Oval,Line=char

        stock,topis=integer

        file     =pdt,hrs,tos,tod,tol,

        top,nop,mos,ele,ira,ova,

        pis,sto,cha,lin


/*global domains

        nonpro,setup,notool,elec,brkdwn,

        Stdjph,Pieces,Hrsreg,Ophrs,msghrs,

        Our,Oput,Mur,Pdty  =real

        alter,ir,Chat,Oval,Line=char

        stock,topis=integer*/
```

```
predicates

     readprod(noprod)

     readwork(work)

     readgeo(geo)

     readtool(toole)

     readmiss(miss)

     run

     gun

     sun

     bun

     fun


goal

     makewindow(1,27,96,"  DATABASE FOR NONPRODUCTIVE HOURS",0,0,25,80),

     shiftwindow(1),

     run,

     makewindow(2,31,11,"  DATABASE FOR NONPRODUCTIVE HOURS",0,0,25,80),

     shiftwindow(2),

     gun ,
```

```
    makewindow(3,27,96,"  DATABASE FOR TOOL STOCK",0,0,25,80),
    shiftwindow(3),
    sun,
    makewindow(4,27,96,"  DATABASE FOR MISSING OPERATORS",0,0,25,80),
    shiftwindow(4),
    bun,
    makewindow(3,31,11,"  DATABASE FOR REJECTED PIECES",0,0,25,80),
    shiftwindow(3),
    fun .

clauses

    readprod(p(Setup,Notool,Elec,Brkdwn)):-
        nl,nl,nl,
        write("   IF YOU ARE UNABLE TO FILL THE DATA THEN"),nl,nl,
        write("   PRESS F1 FOR HELP "),nl,nl,
        write("        How much is the setup time? "),nl,nl,
        readreal(Setup),
        write("       How much is the  notool?"),nl,nl,
```

```
readreal(Notool),

write("        How much is the  elec?"),nl,nl,

readreal(Elec),

write("        How much is the  brkdwn ?"),nl,nl,

readreal(Brkdwn),

Nonpro=Setup+Notool+Elec+Brkdwn,

write("THE TOTAL NON PRODUCTIVE HOURS IS   ", Nonpro,"   HRS").


 openwrite(nol,"nl.dat"),

 writedevice(nol),

 write(Notool),

 closefile(nol),

 openwrite(sot,"se.dat"),

 writedevice(sot),

 write(Setup),

 closefile(sot),

 openwrite(ele,"el.dat"),

 writedevice(ele),

 write(Elec),

 closefile(ele).
```

```
Ophrs=Ophrs*1.0,

Oput=Stdjph*Pieces,

Mur=Oput/Hrsreg,

 write("THE MACHINE UTILIZATION RATIO IS     ",Mur),nl,nl,

 Our=OPut/Ophrs,

 write("THE OPERATOR UTILIZATION IS          ",Our),nl,nl,

 Pdty=Mur*Our*1.25,

 openwrite(pdt,"pd.dat"),

 writedevice(pdt),

 write(Pdty),

 closefile(pdt),

 openwrite(hrs,"hr.dat"),

 writedevice(hrs),

 write(Hrsreg),

 closefile(hrs),

 openwrite(pis,"ps.dat"),

 writedevice(pis),

 write(Pieces),

 closefile(pis).


 write("THE PRODUCTIVITY IS                  ",Pdty),nl,nl.
```

```
gun:-
    readwork(Q),nl,write(Q),nl,nl,
    write("Is this the data of the daily production record OK ? (y/n)"),
    readchar(Ch),Ch='y'.
gun:-
    nl,nl,write("Alright, You may enter again"),nl,nl,gun.


readtool(s(Stock,Topis,Alter)):-
    nl,nl,nl,
    write("   IF YOU ARE UNABLE TO FILL THE DATA THEN"),nl,nl,
    write("   PRESS F1 FOR HELP "),nl,nl,
    write("        How much was the closing stock the required"),nl,
    write("        tool at the end of the day in question ? "),nl,nl,
    readint(Stock),
    write("        How many work pieces were issued during"),nl,
    write("        the period of production ?"),nl,nl,
    readint(Topis),
    write("        Whether alternate tool was available (Y/N)?"),nl,nl,
    readchar(Alter),
    write(Alter),nl,
    Stock= Stock*1.00,
    Topis=Topis*1.0,
     openwrite(top,"tp.dat"),
```

```
     writedevice(top),
     write(Topis),
     closefile(top),
     openwrite(tol,"tl.dat"),
     writedevice(tol),
     write(Alter),
     closefile(tol),
     openwrite(sto,"st.dat"),
     writedevice(sto),
     write(Stock),
     closefile(sto).

sun:-
    readtool(S),nl,write(S),nl,nl,
    write("Is this the data of the tool stock record OK ? (y/n)"),
    readchar(Ch),Ch='y'.
sun:-
    nl,nl,write("Alright, You may enter again"),nl,nl,sun.
```

```
readmiss(t(Msghrs,Ir)):-

     nl,nl,nl,

     write("   IF YOU ARE UNABLE TO FILL THE DATA THEN"),nl,nl,

     write("   PRESS F1 FOR HELP "),nl,nl,

     write("        Total missing hours of the operators onthis machine ?"),nl
,nl,

     readreal(Msghrs),nl,


     write("        Was there any Industrial Relations problem "),nl,

     write("        during the period in question (Y/N)?" ),nl,nl,

     readchar(Ir),

     Msghrs=Msghrs*1.0.

      openwrite(mos,"ms.dat"),

      writedevice(mos),

      write(MsghrsPdty),

      closefile(mos),

      openwrite(ira,"i.dat"),

      writedevice(ira),

      write(Ir),

      closefile(ira).


bun:-

     readmiss(T),nl,write(T),nl,nl,

     write("Is this the data of the Absenteesm  Record OK ? (y/n)"),

     readchar(Ch),Ch='y'.

bun:-

     nl,nl,write("Alright, You may enter again"),nl,nl,bun.
```

```
readgeo(r(Chat,Oval,Line)):-

     nl,nl,nl,

     write("   IF YOU ARE UNABLE TO FILL THE DATA THEN"),nl,nl,

     write("   PRESS F1 FOR HELP "),nl,nl,

     write("         Were there many pieces rejected due to chattering (Y/N)?")
,nl,nl,

     readchar(Chat),nl,

     write(Chat),

---------------------------------------------------------------------------

     write("         Were there many pieces rejected due to "),nl,

     write("         oval shape of the in_process pieces (Y/N)?" ),nl,nl,

     readchar(Oval),

     write(Oval),

     write("         Were there many pieces rejected due to "),nl,

     write("         lines formed on the job (Y/N)?"),nl,nl,

     readchar(Line),nl,

     write(Line).

      openwrite(cha,"ch.dat"),

      writedevice(cha),

      write(Chat),

      closefile(cha),
```

```
openwrite(ova,"ov.dat"),
writedevice(ova),
write(Oval),
closefile(ova),
openwrite(lin,"li.dat"),
writedevice(lin),
write(Line),
closefile(lin),


fun:-
    readgeo(R),nl,write(R),nl,nl,
    write("Is this the data on the machine failure OK ? (y/n)"),
    readchar(Ch),Ch='y'.
fun:-
    nl,nl,write("Alright, You may enter again"),nl,nl,fun.
```

```
/* THIS IS THE REASIONIG SECTION */

domains
   name,area,opgroup=string
   hrsreg,tothrs,stdjph,ophrs,pdty,
  mur,our, oput,msghrs,nonpro,elec,setup,notool=real
   pieces,totdays,topis,stock,brkdwn=integer
   ir,chat,ovol,alter=logical

/*   domains of file type  */
     file= pdt,hrs,tos,tod,tol,
     top,mos,ira,nop,ele,ova,
     lin,pis,sto
 global domains
    name,area,opgroup=string
    hrsreg,tothrs,stdjph,ophrs,pdty,
    mur,our, oput,msghrs,nonpro,elec,setup,notool=real
    pieces,totdays,topis,stock,brkdwn=integer
    ir,chat,ovol,alter=logical

/*  global  domains of file type  */
     file= pdt,hrs,tos,tod,tol,
     top,mos,ira,nop,ele,ova,
     lin,pis,sto

  database
```

```
predicates
     go
     calco
     gogo
     messg
     enquir
     okay
     kool
     begin
     manpo
     noplan
     shift
     idle
     ipro
     down
     trip
     elef
     set
     tool
    online
    ovule
    chatter
     tooling
     toolgeo
     toolna
     toolal
```

```
clauses
      begin,
      manpo,
      down,
      tool,
      nl.

     /*           THE RULE BASE BEGINS HERE

     begin:-
       okay,
       kool,
       nl.

     /* rule 1 */

   okay:-
      openread(pdt,"pd.dat"),
      readdevice(pdt),
     Pdty<0.7.

/* rule2 */
   kool:-
```

```
    openread(pdt,"pd.dat"),

    readdevice(pdt),

   Pdty> 0.7,

   write("            The productivity of your firm is considered"),nl,

   write("            to be satisfactory . You may however read"),nl,

   write("            the following analysis of the produtivity factors")
nl,

   write("            applicable to the group in question."),nl,

   write(""),nl,write(""),nl,write(""),nl, write(""),nl.

  manpo:-

     noplan,

     shift,

     idle,

     ipro,nl.

     /* rule 3 */

noplan:-

    openread(hrs,"hr.dat"),

    readdevice(hrs),

    openread(tos,"ts.dat"),

    readdevice(tos),

   Hrsreg < 8*0.75*Totdays,
```

```
write("            The hours booked for production is not  adequate. ")
,nl,
write("            Directives  must be sent to the  PPC department. "),
nl,
write("            An explanation should be asked for ."),nl,
write(""),nl,write(""),nl,write(""),nl, write(""),nl.


   /* rule 4 */

shift:-

 openread(tos,"ts.dat"),
 readdevice(tos),

 openread(tod,"td.dat"),
 readdevice(tod),

 Tothrs/Totdays < 6,

 write("            The shift incharge as not put the operators "),nl,
 write("            on duty  at the right time.  "),nl,
 write("            He should be  contacted for explanation.  "),nl,
 write(""),nl,write(""),nl,write(""),nl, write(""),nl.
```

```
   /* rule  5 */

 idle:-

   openread(tod,"td.dat"),

   readdevice(tod),

   openread(mos,"ms.dat"),

   readdevice(mos),

   Msghrs/Totdays > 2,

  write("           The  hours for which the workers  were  "),nl,
  write("           booked idle is on the higher side"),nl,
  write("           If warning has already been given "),nl,
  write("           last month than mobilize the labour "),nl,
  write("           relations cell  to deal with it.  "),nl,
  write(""),nl,write(""),nl,write(""),nl, write(""),nl.


          /* rule 6 */

ipro:-

   openread(ira,"i.dat"),

   readdevice(ira),

IR = 'Y',
```

```
write("            It is  very urgently necessary to tackle"),nl,
write("            the Industrial Relations problem "),nl,
write("            present in the area concerned."),nl,
write("            It should be  accorded high on the"),nl,
write("            priority  of the management."),nl,
write(""),nl,write(""),nl,write(""),nl, write(""),nl.


 down:-
    trip,
    elef,
    set,nl.

   /* rule 7 */

trip:-
 openread(tod,"td.dat"),
 readdevice(tod),

 openread(nop,"np.dat"),
 readdevice(nop),

 Nonpro/Tothrs > 0.15,

write("           The non productive hours other than  "),nl,
write("           absenteesm,lack of planning are due to"),nl,
write("           the  several reasons summed up below. "),nl,
write(""),nl,write(""),nl,write(""),nl, write(""),nl.

   /* rule 8 */
```

```
elef:-

 openread(tod,"td.dat"),

 readdevice(tod),

 openread(ele,"el.dat"),

 readdevice(ele),

Elec/Totdays > 0.5,


write("          The slump in productivety could  be due to"),nl,

write("          frequent electricity. The concerned officials "),nl,

write("          of the electricity board should be informed."),nl,

write(""),nl,write(""),nl,write(""),nl, write(""),nl.

  /* rule 9 */

set:-

 openread(tos,"td.dat"),

 readdevice(tos),

 openread(sot,"se.dat"),

 readdevice(sot),

Setup/Tothrs >0.15,

write("          Apparently, the  setup time is high."),nl,

write("          This could  be due either to the inexperience"),nl,

write("          of the operator doing the setting"),nl,

write("          or to the lack of proper fixture."),nl,

write("          It is advised that the shift in charge  "),nl,

write("          enquire into the problems and"),nl,
```

```
    write("          decide on further course of action."),nl,
    write(""),nl,write(""),nl,write(""),nl, write(""),nl.

tool:-
   tooling,
   toolgeo,
   toolna,
   toolal,nl.

        /* rule 10 */

    tooling:-

     openread(tos,"ts.dat"),
     readdevice(tos),
     openread(nol,"nl.dat"),
     readdevice(nol),
     Notool/Tothrs < 0.15,

    write("          Your group has a tooling problem"),nl,
    write("          It could be due to several reasons."),nl,
    write("          you can follow further in the diagnosis."),nl,
    write(""),nl,write(""),nl,write(""),nl, write(""),nl.
```

```
          /* rule 11 */

       toolgeo:-

         openread(top,"tp.dat"),

         readdevice(top),

         openread(pis,"ps.dat"),

         readdevice(pis),

          Pieces/Topis < 0.15,

         chatter,

         online,

         ovule,nl.

/* rule 12 */

        chatter:-

         openread(cha,"ch.dat"),

         readdevice(cha),

             Chat='Y',

        write("         The high rate of regrinding could be due"),nl,

        write("         cattering of the tool.The percentage of"),nl,

        write("         pieces lost due to cattering is high."),nl,

        write("         This could be due to improper tool geometry"),nl,
```

```
write("          It could also be due to vibrations due to"),nl,
write("          spindle misalignment"),nl,
write("          Inform the tool crib manager.    "),nl,
write(""),nl,write(""),nl,write(""),nl, write(""),nl.

  /*  rule 13 */

onlin:-
 openread(lin,"li.dat"),
 readdevice(lin),

 Line='Y',

write("          Remind the operator to about the tool"),nl,
write("          specifications and proper setting procedures"),nl,
write("          There is problem with the  holding chucks"),nl,
write(""),nl,write(""),nl,write(""),nl, write(""),nl.

  /*  rule 14 */


ovule:-

 openread(ova,"ov.dat"),
 readdevice(ova),
 Oval='Y',
 write("          Vibration problem or spindle alignment problem  "),n
l,
 write("          or fixture or setting problem can cause "),nl,
 write("          oval pieces.Inform the maintance group."),nl,
 write("          Direct the  shopfloor manager to the erring operator
"), nl,

 write(""),nl,write(""),nl,write(""),nl, write(""),nl.
```

```
/*  rule 16 */

toolal:-

    openread(tol,"tl.dat"),

    readdevice(tol),

   Alter='Y',

    write("          Direct manager to release the alternate tool. "),nl,

    write("          The manager must make a note of the same in the "),n

    write("          in the appropriate remark column of the tool registe
."),nl,

   write(""),nl,write(""),nl,write(""),nl, write(""),nl.
```

```
   /* rule 15 */
toolna:-

   openread(sto,"st.dat"),
   readdevice(sto),
   openread(tol,"tl.dat"),
   readdevice(tol),
   Stock=0,
   Alter='N',
   write("          Place an urgent telex to the regular"),nl,
   write("          supplier. (MATTER MOST URGENT. EXPEDITE!  ETC.)"),nl

   write("          Send purchase officer to tool crib"),nl,
   write("          and find out the position.  And direct "),nl,
   write("          him to take further necessary action. Ask "),nl,
   write("          for a report of current stock position data."),nl,
   write(""),nl,write(""),nl,write(""),nl, write(""),nl.
```

OUTPUT OF P.E.D.

THE PRODUCTIVITY PERFORMANCE OF A
COMPANY PRODUCING ENGINEERING GOODS

Please write the name of your firm.

B R ENGG CO.

You have to mention the specific shop floor area for which you want to assess productivity. It is advised that you take a set of machinery as the unit.

ENTIRE SHOP FLOOR.

database

(23,21,11,11.4)

(12 3,14,4)

(21,4,'y')

('y','y','n')

The productivity of your firm is considered to be satisfactory . You may however read the following analysis of the produtivity factors applicable to the group in question.

The hours booked for production is not adequate. An explanation should be asked for .

The shift incharge as not put the operators on duty at the right time.

He should be contacted for explanation.

The hours for which the workers were booked idle is on the higher side

If warning has already been given last month than mobilize the labour relations cell to deal with it.

The non productive hours other than absenteesm,lack of planning are due to the several reasons summed up below.

The slump in productivety could be due to frequent electricity. The concerned officials of the electricity board should be informed.

Place an urgent telex to the regular supplier. (MATTER MOST URGENT. EXPEDITE! ETC.) Send purchase officer to tool crib and find out the position. And direct him to take further necessary action. Ask for a report of current stock position data.

```
program prosper(input);

(* DECLARATION SECTION *)

var

a1,a2,a3,a4,a5,a6,a7,a8,a9,

b1,b2,b3,b4,b5,b6,b7,b8,b9,b10,b11,b12,b13,

c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12,c13,c14,c15,c16,c17,

d1,d2,d3,d4,d5,d6,d7,d8,

e1,e2,e3,e4,e5,e6,e7,e8,e9,e10,e11,e12,e13,e14,

a,b,c,d,e,

organ,procr,intr,eqpt,

flay,comm,invest,recado,

laban,testeq,drawn,poseg,qcmeth,total:integer;
(*    OUTPUT  FILES    *)

OUTF: FILE OF INTEGER  ;

ORG : FILE OF INTEGER  ;

PRO : FILE OF INTEGER  ;

INT : FILE OF INTEGER  ;

FLA : FILE OF INTEGER  ;

EQP : FILE OF INTEGER  ;

COM : FILE OF INTEGER  ;

INV : FILE OF INTEGER  ;
```

```
REC : FILE OF INTEGER ;
LAB : FILE OF INTEGER ;
TES : FILE OF INTEGER ;
DRA : FILE OF INTEGER ;
POS : FILE OF INTEGER ;
QCM : FILE OF INTEGER ;

begin

(* ASSIGNMENT OF OUTPUT*)

ASSIGN( OUTF, 'DATA.DAT' ) ;
ASSIGN( ORG, ' ATA.DAT' ) ;
ASSIGN( PRO, 'DATP.DAT' ) ;
ASSIGN( INT, 'DATI.DAT' ) ;
ASSIGN( FLA, 'DATF.DAT' ) ;
ASSIGN( EQP, 'DATE.DAT' ) ;
ASSIGN( COM, 'DATC.DAT' ) ;
ASSIGN( INV, 'DATN.DAT' ) ;
ASSIGN( REC, 'DATR.DAT' ) ;
ASSIGN( LAB, 'DATL.DAT' ) ;
ASSIGN( QCM, 'DATQ.DAT' ) ;
ASSIGN( TES, 'DATT.DAT' ) ;
ASSIGN( DRA, 'DATD.DAT' ) ;
ASSIGN( POS, 'DATS.DAT' ) ;
```

```
(*ASSIGN( OUTF, 'PRN');*)

graphcolormode;

graphwindow(40,40,260,140);

graphbackground(9);

textbackground(15);


(* THIS IS THE INTRODUCTION PART*)

writeln('                  welcome to');

writeln;writeln;writeln('                  PROSPER'); writeln;writeln;

writeln('This software will help you in many ways');writeln;

writeln('Are you concerned about the quality of');writeln;

writeln('  (a) the products of your firm?');

writeln('  (b) the products of your supplier?');

writeln('  (c) the produce of a third party?'); writeln;

writeln('If yes is the answer then leave it to us');writeln;

writeln('Prosper will analyse  the data provided');writeln;

writeln('by you.');writeln;writeln;writeln;

writeln('          Just press "return"');readln(kbd);

clrscr;graphbackground(6);textbackground(15);


writeln('            PLEASE FOLLOW ALL THE INSTRUCTIONS CAREFULLY');

WRITELN('          (1)    Read the Instruction Sheet carefully');writeln;

writeln('          (2)    If you dont have one, then type "Y" at the approp
place');
```

```
writeln;writeln('          (3)   There are two parts in the instruction sheet .')
;
writeln('          (a) Ask for these (b) Observe the following');writeln;
writeln('          (4) Ask for everything that is wanted and consider');
writeln('              refusal as "not available"');writeln;
writeln('          (5)    After completing the above you have to answer certain q
uestions');writeln;
writeln('          (6)    You have grade on the basis of the information availabl
e about the firm ');writeln;
writeln('                 You have to give integer grading from 0 to 10');
writeln('          (7)    If you donot know how to answer them then type "help"')
;writeln;
writeln('          (8)    Do not miss any question .');writeln;
writeln('          (9)    Before you attempt any question see to it that you have
');
writeln('                 obtained all the tnformation wanted in instruction (3).
');
writeln('                    ALL  THE  BEST');
readln(kbd);clrscr;graphbackground(11);textbackground(12);


(* THE QUESTIONAIRE BEGINS HERE *)



writeln;writeln;
writeln('A:      QUALITY  CONTROL ORGANIZATION');writeln;
writeln('A1. Is quality control  a');
writeln(' separate and distinct part');
writeln('of the suppliers organization?');
readln(a1);
```

```
 writeln('A2. Are adequete Written procedures');
writeln('  defining quality control operations');
writeln(' and functions established and is');
writeln(' their sufficient evidence that');
writeln(' they are being adhered to ?');
readln(a2);

writeln('A3 Are gages and test equipment ');
writeln('inspected and approved to design ');
writeln('specifications being used for parts');
writeln(' inspection or test ?');
writeln('Are gage receiving inspection records');
writeln(' available ? Are tool and gage');
writeln(' inspectors provided ?');
readln(a3);

writeln('A4 Are suitable Inspection areas');
writeln(' provided ? Are they well organized,');
writeln(' clean, adequately lighted, and');
writeln(' sufficiently equipped?');
readln(a4);
```

```
writeln('A5. Do they have a system of promptly');
writeln('notifying management scrap or rejection?');
readln(a5);

writeln('A6. Does the suppiers have a system for');
writeln(' investigating,effecting corrective');
writeln('action and answaring customer rejections?');
readln(a6);

writeln('A7. If applicable are quality control personnel');
writeln(' familiar with control items and their critical');
writeln('characteristics,and are these properly');
writeln('indentified on Quality Control documents?');
readln(a7);
clrscr;graphbackground(7);textbackground(13);



writeln(': "B. INCOMING PARTS AND MATERIAL QUALITY CONTROLS"');
writeln;writeln;

writeln('B1. Are all incoming shipments submitted');
writeln('to inspection for disposition before beingused in product ?');
readln(b1);
```

```
writeln('B2. Are incoming inspection records maintained');
writeln('for each shipment received ? Do they they indicate');
writeln('that inspection frequencies are adequate ?');
readln(b2);

writeln('B3. Are written inspection instructions provided');
writeln('and used as a supplement to the part drawing as intended?');
writeln('Do inspection instructions and records specify and');
writeln('records the shipment quantity and sample size for');
writeln('major and minor characteristics?');
readln(b3);

writeln('B4. Do records indicate that parts and material');
writeln('are subjected to laboratory analysis at specified frequency?');
writeln('Are frequenceies adequate and is laboratory dispostion');
writeln('indicated for each submission?');
readln(b4);

writeln('B5. Are adequate gages and test equipment ');
writeln('provided to measure all characteristics of incoming');
writeln('parts and material ? How many gages are provided for OUR parts ?'
readln(b5);
```

```
writeln('B6. Do records indicate that gaging and test');
writeln('equipment are periodically inspected? ? What is the frequency ?');;
readln(b6);

writeln('B7. Are inspection dispositions clearly indicated on');
writeln('each package or container of every shipmenr received?');
readln(b7);

writeln('B8. Do records indicate that ES or that special tests');
writeln('are performed by sub-supplier with audlts performed by the supplier');
readln(b8);

writeln('B9. Is there evidence that the source of supply is ');
writeln('immediately notified of rejections by telephone or telegram and f');
writeln('with written rejection communication ?');
readln(b9);

writeln('B10. It there documentation to verify that the suppliers');
writeln('purchasing activity assists in effecting corrective action source ?');
readln(b10);

writeln('B11. Are sorted or reworked shipments identified and');
writeln('resubmitted to inspection for disposition ?');
readln(b11);
```

```
writeln('B12. Are adequate qualified manpower and work area');
writeln('provided to assure that incoming inspection');
writeln('function are efficiently performed ?');
readln(b12);

writeln('B13. Are rejected shipments clearly identified and ');
writeln('segregated in an adequate hold area pending return');
writeln(' or other dispo Is there evidence that rejected material');
writeln('  is disposed of within a reasonable time limit ?');
readln(b13);
clrscr;graphbackground(1);textbackground(15);


writeln('"C. IN-PROCESS QUALITY CONTROLS"');
  writeln;writeln;

writeln('C1. Are written inspection instruction sheets');
writeln('prepared and used for all inspection operations ?');
readln(c1);

writeln('C2. Are all characteristics affecting function,');
writeln('life, or appearance highlighted on the inspection instruction sheets');
readln(c2);
```

```
writeln('C3. Are the latest engineering drawings and specifications');
writeln('available at inspection operations ?');
readln(c3);

writeln('C4. Are adequate gages, checking aids, and test');
writeln('equipment provided, properly located and used ?');
readln(c4);

writeln('C5. Are records available that show gages and');
writeln('test equipment are periodically inspected ? What is');
writeln('the inspection frequency ?');
readln(c5);

writeln('C6. Do records show that first piece inspection');
writeln('set-up approval is required for manufacturing operations');
writeln('that are not continuous ?');
readln(c6);

writeln('C7.  Is there evidence that spot and projection');
writeln('welds are destructively checked for button size and');
writeln('that fusion welds are sectioned to check penetration');
writeln('and weld size at adequate frequency ?');
readln(c7);
```

```
writeln('C8. Do thay have records to show that in process floor');
writeln('inspections are performed at regular specified  frequencies ?');
readln(c8);

writeln('C9. Do they have records to show that process capability');
writeln('studies were completed prior to application of recognized');
writeln('statistical quality control techniques, such as sampling');
writeln('plans. X and R charts, etc. ?');
readln(c9);

writeln('C10. Are completed lots of parts or material');
writeln('held pending inspection or test disposition ?');
readln(c10);

writeln('C11. Is there evioence hat lots of parts are');
writeln('sorted whenever defects are found? Is defective material, clearly');
writeln('identified and segregated from approved material ?');
readln(c11);

writeln('C12. Are inspection and rest dispositions clearly');
writeln('indicated on lots of parts or material ?');
writeln('Are suitable controls in effect to accurately, retain parts');
writeln('identification and inspection disposition throughout');
writeln('in - plant processes ?');
readln(c12);
```

```
writeln('C13. Are production operations  promptly corrected');
writeln('or shut down until corrected ? Does inspection have authority to');
writeln('production operations that are not producing acceptable quality ?');
readln(c13);

writeln('C14. Are sorted or reworked parts and materials submitted to');
writeln('inspection for disposition before');
writeln('further processing ?');
readln(c14);

writeln('C15.It required, are magnetic particle, X-ray or other');
writeln('special tests being performed at adequate frequency ?');
readln(c15);

writeln('C16.Do records indicate that complete layout inspections');
writeln('and tests are performed  at adequate frequencies ?');
readln(c16);

writeln('C17.Are cleaning facilities and processes adequate');
writeln('to assure cleanliness of parts in accordance with their functiona');
readln(c17);

 writeln;writeln;
 clrscr;graphbackground(14);textbackground(5);
```

```
writeln('"D. CHEMICAL AND METALLURGICAL QUALITY CONTROL"');
writeln;writeln;
writeln('D1. Is the laboratory and laboratory personnel a part');
writeln('of the suppliers quality control organization ?');
readln(d1);

writeln('D2. Are laboratory test instruction sheets prepared');
writeln('for each part number and properly used ?');
readln(d2);

writeln('D3. Are the latest company engineering drawing and');
writeln('specification provided to the laboratory ?');
readln(d3);

writeln('D4. Do records indicate that laboratory testing');
writeln('equipment is periodically inspected and calibrated ?');
readln(d4);

writeln('D5. Do records indicate that laboratory material analysis');
writeln('frequencies are adequate for purchased and plant-make part');
writeln('and material ?');
readln(d5);

writeln('D6. Are samples of washing or cleaning solutions');
```

```
writeln('checked at regular scheduled frequency ?');
readln(d6);

writeln('D7. Are laboratory tests made of painted parts for');
writeln('thickness, color, and other requirements specified by Engineering');
writeln('adequate frequency ?');
readln(d7);

writeln('D8. Do records indicate that purchased or plant-make');
writeln('heat treated and plated parts are tested at adequate frequency  ');
writeln('What types of heat treatments are performed at this plant or');
writeln('by sub-suppliers for products this plant produces ?');
readln(d8);
clrscr;graphbackground(2);textbackground(15);



writeln('"E. OUTGOING QUALITY CONTROLS"');
writeln;writeln;

writeln('E1. Is final disposition required for all parts and');
writeln('clearly indicated on individual assemblies or');
writeln('parts shipping containers ?');
readln(e1);
```

```
writeln('E2. Are written inspection instructions prepared and');
writeln('used by final inspection personnel ?');
readln(e2);

writeln('E3. Are the latest applicable Ford engineering drawings');
writeln('and specifications availiable in the final inspection area ?');
readln(e3);

writeln('E4: Are adequate gages, checking aids and test equipment');
readln(e4);

writeln('provided and used ? Number of gages provided ?');
writeln('E5. Do records indicate that gaging and test equipment');
writeln('are periodically inspected ? What is the frequency ?');
readln(e5);

writeln('E6. Is 100% or adequate sampling inspection used in');
writeln('accordance with proper company standards ?');
writeln('If not 100% describe their sample plan.');
readln(e6);
```

```
writeln('E7. Are defective parts of assemblies properly icentified');
writeln('and segregated from approved material ?');
writeln('If so does their system provided for immediate');
writeln('disposition of defects by rework or scrap ?');
readln(e7);

writeln('E8. Do records indicate that ES and other special tests');
writeln('are being performed at specified frequencies ?  Describe the');
writeln('test and the frequency ?');
readln(e8);

writeln('E9. Does the supplier have a procedure for periodic for');
writeln('audit of approved products ? if so they disassemble ES test');
writeln('units to check proper manufacturing, proper assembly, wear');
writeln('pattern cleanliness and other');
writeln('major characteristics ? Is an inspection check-off sheet');
writeln('provided and retained for records ?');
readln(e9);

writeln('E10. In the event of ES test failure, is proper action');
writeln('taken in accordence with Ford Q-101 instructions ?');
readln(e10);

writeln('E11. Are reworked parts and assemblies re-submitted to');
writeln('inspection for final disposition ?');
readln(e11);
```

```
writeln('E12. Are controls suitable to prevent  movement of');
writeln('defective parts and assemblies to the shipping area, and prevent');
writeln('accidental shipment ?');
readln(e12);

writeln('E13. Are parts and assemblies werehouse for shipping clearly');
writeln('identified and controlled for rotation.');
readln(e13);

writeln('E14. Are packing and handing procedures adequate to');
writeln('preserve product quality through shipping ? Is packing in accord');
writeln('the company standards for packaging?');
readln(e14);


(*THE CALCULATION PART BEGINS HERE  *)

a:=a1+a2+a3+a4+a5+a6+a7+a8+a9; WRITE( OUTF, A)  ;
b:=b1+b2+b3+b4+b5+b6+b7+b8+b9+b10+b11+b12+b13; WRITE( OUTF, B)  ;
c:=c1+c2+c3+c4+c5+c6+c7+c8+c9+c10+c11+c12+c13+c14+c15+c16+c17; WRITE( OUTF, C)
;
d:=d1+d2+d3+d4+d5+d6+d7+d8; WRITE( OUTF, D)  ;
e:=e1+e2+e3+e4+e5+e6+e7+e8+e9+e10+e11+e12+e13+e14; WRITE( OUTF, E)  ;
```

```
organ:=a1+a7+b12+d7; WRITE( ORG, ORGAN)  ;

procr:=a2+b3+c5+d5+d6+e2; WRITE( PRO, PROCR)  ;

intr:=b3+c1+c2+d2+d3+e2;  WRITE( INT, INTR)  ;

flay:=a4+b7+b13+c10+c13+c17+e1+e14; WRITE( FLA, FLAY)  ;

eqpt:=a3+b5+c4+c7+c15+d2+d5+e4; WRITE( EQP,EQPT )  ;

comm:=a5+b9+e13; WRITE( COM, COMM)  ;

invest:=a6+c10+e13;  WRITE( INV, INVEST)  ;

recado:=b2+b3+b4+b10+c5+c6+c8+c9+c16+e5+e8; WRITE( REC, RECADO)  ;

laban:=b4+d8; WRITE( LAB, LABAN)  ;

testeq:=b6+d4+e5; WRITE( TES, TESTEQ)  ;

drawn:=c3+d3+d4+e3; WRITE( DRA, DRAWN)  ;

poseg:=b11+b13+c11+112+e7*e12; WRITE( POS, POSEG)  ;
```

```
qcmeth:=b1+b11+c8+c9+c10+c14+d7+d8+e1+e6+e9+e11;write(qcm,qcmeth);
total:=a+b+c+d+e;

 writeln(total);

 writeln('                          QUALITY ASSURANCE SURVEY REPORT');writeln;writeln;w
riteln;

 writeln('                         Thank you for your cooperation. ');writeln; writeln;

if total>380 then

writeln('         On the basis of our report we found your quality control satisfa
ctory');

if total<300 then

writeln('         Our survey report indicates that quality control in your company
 is not adeqate');

if (total>300) and (total<380) then

writeln('          We reject your offer tentatively .To become eligible see below'
);writeln;

writeln('        The following information may be of use to you :');writeln;

if a<50 then

writeln('           * You have a weak  quality control organization and you ');

if a<50 then
```

```
writeln('            dont have adequate qualified manpower');

if b<60 then

writeln('          * The report indicates that your materials and purchase');

if b<60 then

writeln('            department  doesnt have proper  QC  ');

if c<100 then

writeln('          * The in-process inspection of your firm is grossly lacking');

if d<45 then

writeln('          * The product-mix of your firm necessitates you to acquire');

if d<45 then

writeln('            certain testing facilities .You do not posses many of them.'
);

if e<85 then

writeln('          * Your out-going QC and  packaging leaves a lot to be desired.
');WRITELN; WRITELN;

writeln('     For more specific advice you may seek the help of PROSPER II ');

end.
```

```
run:-
    readprod(P),nl,write(P),nl,nl,
    write("Is this the data of the daily production record OK ? (y/n)"),
    readchar(Ch),Ch='y'.
run:-
    nl,nl,write("Alright, You may enter again"),nl,nl,run.

readwork(q(Stdjph,Pieces,Hrsreg,Ophrs)):-
    nl,nl,nl,
    write("   IF YOU ARE UNABLE TO FILL THE DATA THEN"),nl,nl,
    write("   PRESS F1 FOR HELP "),nl,nl,
    write("          How much is the standard job per hour rating ? "),nl,nl,
    readreal(Stdjph),
    write("          How many are the pieces ?"),nl,nl,
    readreal(Pieces),
    write("          How many hours were registered ?"),nl,nl,
    readreal(Hrsreg),
    write("          How many were the operator hours booked?"),nl,nl,
    readreal(Ophrs),
    Stdjph=Stdjph*1.00,
    Pieces=Pieces*1.0,
    Hrsreg=Hrsreq*1.0,
```

```
/*PROSPER IS THE PROTOTYPE OF A QUALITY ASSURANCE EXPERT SYSTEM*/

/*  A CONSIDERABLE PART OF THE  KNOWLEDGE BASE HAS  NOT */

/*  BEEN SHOWN HERE.  HOWEVER ITS CHARACTER REMAINS INTACT */

  domains

       a=integer

  total,prodra,qcmeth,a, a1=integer

       name=string

      file= toto,qcm,prod,layo

   predicates

      verify

       ngo

     layo

      pro

  goal

     makewindow(1,31,7,"",Ø,Ø,25,8Ø)



clauses

   ngo:-

        shiftwindow(1),

                                                  write(""),nl,write(""),nl,
        write("                         QUALITY ASSURANCE SURVEY SYSTEM").NL,

        nl,nl,nl,

       write("           You are in to the specific advise section "),nl,nl

       write("           What is the name of the firm ?"),nl,
```

```
openread(toto,"tot.dat"),
readdevice(Toto),
read(Total),
closefile(toto),

   TOTAL <  225,
 write("    (A)    The Quality Control Organization of [illegible]

layo:-
openread(layo,"la.dat"),
readdevice(layo),
read(Lay),
closefile(layo),
LAY < 55,
 write("    (B)  Your layout leaves a lot to be desired"),nl
 write("     The bins have to be [illegible]
 write("     has to be kept clean."),nl.
write("      The layout [illegible]
write("      satisfactory.  The measures [illegible]
write("      been [illegible]
write("      We wish that you pay more attention [illegible]
write("      regard to [illegible]
```

```
pro:-

    openread(prod,"pro.dat"),
    readdevice(Prod),
    read(Drawn),
    closefile(prod),

    openread(qcm,"qc.dat"),
    readdevice(qcm),
    read(Qcmeth),
    closefile(qcm),

     PRODRA <21
     QCMETU <55,
     write("    (C)    There are sevaral flaws in your [illegible]
     write("       relating to [illegible]
     write("       (iii) out-going  quality control [illegible]
     write("       [illegible]
     write("       for reference.").nl.nl
```

BIBLIOGRAPY

1. Clocksin, W.F., and C.S. Mellish, "Programming in Prolog". Berlin: Springer Verlag

2. Forsyth, Richard. "Expert Systems: Principles and Case Studies", London: Chapman Hall

3. Hayes Roth, Frederick, et al. " Building Expert System". Reading, Mass: Addison Wesley, 1983. (This book is basically LISP Oriented.)

4. Carl Townsend. "Introduction to TURBO PROLOG", SYBEX.

5. Kellers Robert, "Expert Systems Technology" Yourdon Press, ENGLEWOOD CLIFFS, NJ 07632.

6. Bruno, G., Elia A. et al Politecnico Di Torimo, ITALY "A Rule Based System to Schedule Productions" IEEE Computer 1986, June.

7. Holsapple, C.W., and Whinston AB, "Business Expert System" 1988 Richard Irvin Inc. Computer Science Series.

8. Groover MP & Zimmerman E.W. J&R, "Computer Aided Design and Manufacturing" PHI 1984

9. Edwards, CE "Advanced Techniques in TURBO PASCAL" SYBEX 1987

10. Nath. Sanjiva "PROLOG: Features for Programmers" By MIS. Inc 1986

11. "Aspects of Integrated Software"(CR) Procedings of N C C, Las vegas 1984

12. "The Nature of Software Integration" Info age, October, 1984. (CR)

13. "Synergistic Software Integration for Micro-Computers" Systems and Software 3 no. 2, 1984