An Investigation and Evaluation into the 'Usability' of Human–Computer Interfaces using a Typical CAD System.

By

John Desmond Rickett.

A thesis presented to the
Council for National Academic Awards
in partial fulfilment of the
requirements for the degree of
DOCTOR OF PHILOSOPHY.

School of Mathematics, Computing and Statistics
Leicester Polytechnic.

In collaboration with
FEMVIEW Ltd.

October 1987.

ABSTRACT

An Investigation and Evaluation into the 'Usability' of Human-computer Interfaces Using a Typical CAD System.

J. D. Rickett

This research program covers three topics relating to the human-computer interface namely, voice recognition, tools and techniques for evaluation, and user and interface modelling.

An investigation into the implementation of voice recognition technologies examines how voice recognisers may be evaluated in commercial software. A prototype system was developed with the collaboration of FEMVIEW Ltd. (marketing a CAD package). Proposals for future research using the prototype system suggests the need for field trials to assess its usefulness in a working environment and to gain insights to end-user attitudes. A new generation of voice system is proposed based around a phoneme-based pattern matching paradigm, natural language understanding facilities and intelligent knowledge-based systems capable of building on knowledge by inference and deduction.

In order to assess the 'usability' of the FEMVIEW CAD software a subject-based formal evaluation was conducted which involved:-

(1) the analysis of responses to a multi-user survey of end-user attitudes;
(2) collecting behavioural performance measures from students learning to use the software.
(3) cognitive and affective data obtained from laboratory experimentation using experienced users of the CAD package.

A theoretical approach to evaluation leads to the hypothesis that human-computer interaction is affected by personality, influencing types of dialogue, preferred methods for providing help, etc. A user model based on personality traits, or habitual behaviour patterns (HBP) is presented. Proposals are given to use the HBP model in future self-adaptive interfaces. Results from experimentation to justify the model are inconclusive.

Finally, a practical framework is provided for the evaluation of human-computer interfaces. It suggests that evaluation is an integral part of design and that the iterative use of evaluation techniques throughout the conceptualisation, design, implementation and post-implementation stages will ensure systems that satisfy the needs of the users and fulfil the goal of 'usability'.

The major contributions made to the knowledge of this subject can be summarised as follows:

(1) the practical problems of implementing voice recognition technologies in commercial software;
(2) the development of a new personalised user model which accounts for individual's idiosyncrasies;
(3) methods for applying simple evaluation techniques in order to assess software 'usability';
(4) a practical framework for developing usable software.

Contents.

Contents.

List of Figures.

List of Tables.

List of Appendices.

Chapter 1

Introduction.

Chapter 1.

Introduction.

1.1. Initiative for the project.

This research project was proposed as a consequence of several questions raised by members of the Human Computer Interface Research Centre (HCIRU). A fortran-based independent user processor which was used in several commercially available engineering CAD systems had been developed within the unit (Bramer,1983, Bramer, 1984). This standard input processor provided a fast, flexible and easy to use command interface. Reappraisal of this interface was necessary in the light of changing design philosophies. With greater emphasis on user-oriented design and with the growing trends towards voice recognition careful evaluation was necessary. While the study concentrates on the evaluation of one software engineering package, the principles of operation are typical of many CAD packages and therefore it is possible that observations made in relation to this single package may be generally applicable over a number of CAD systems.

1.2. C.A.D. and FEMVIEW.

The primary task of Computer Aided Design (CAD) systems is design, the mental functions of the CAD user should therefore be focused on the task rather than the commands which will carry out particular functions (Majchrzak et al, 1986). Design is a creative process requiring complex human learning which is primarily 'active' rather than 'passive'. In this respect using CAD systems is not unlike using other packages such as word processors (Carroll & Mack, 1984).

The FEMVIEW package represents an engineering CAD package whose primary concern is the finite element analysis of engineering designs. It addresses itself to the twofold problem of assessing that the finite element model has been generated correctly, and of enabling the user to present the results of his analysis in an informative and illuminating way. In order to achieve these goals, a variety of techniques for model visualisation, the selection of elements, and the display of results are provided. Although this study has concentrated on the FEMVIEW software, in practice, the package is usually used in conjunction with its sister program FEMGEN which provides the means of generating the models which are subsequently viewed using FEMVIEW. CAD designs are therefore developed through the combined use of both packages. Both systems make use of the fortran based machine independent command processor and both constitute aspects of CAD.

1.3. The Need for Research into Voice Recognition.

Since voice recognition is covered in some detail in Chapter 3 very little need be said here. However, at the start of this research project voice recognition was a topic receiving much acclaim. Whether from extensive media coverage afforded by manufacturers of voice products or moves by the ALVEY directorate to generate industry backed research and development, the publicity caused software houses with large investments in commercial packages some concern and, as a result, this research program was proposed. It presents an unbiased study of voice recognition technologies examining both methods of implementation, and its future prospects.

1.4. Defining the User Interface.

The word interface has many different connotations and even within the field of computer science there appears to be some disagreement as to where the term should and should not be used. The problem would appear to stem from the use of the abstract concept of an interface in the practical field of computer technology. The definition of the human-computer interface assumed throughout this thesis is described as:-

> 'all the components of the computer affecting interaction between the user and the computer'.

Even though it is possible to to consider the interface as a separate component of the system (Edmonds,1982, Alty, 1984), to satisfy the goals of the evaluation, application dependent aspects of the system are also considered.

1.5. Evaluation: The Key to Successful Interface Designs.

Having defined the user interface our next task is to examine ways of designing effective interfaces. Edmonds (1981) considers the key to success is evaluation, which provides feedback on interface design. Evaluation consists of a structured set of activities which enable a system to be tested in an objective way and to provide recommendations as to how the design can be improved to resolve the problems highlighted by the testing. An evaluation will cover both static and dynamic aspects of the system and subjective as well as objective measures of the users performance. In the evaluation of the FEMVIEW interface three major areas are addressed:-

(1) Does the software fulfil it's role in industry? (i.e. Is it being used as intended?)

(2) Are the system's facilities matched to the cognitive and aesthetic capabilities of the users?

(3) Are the control and display aspects of the interface suited to the sensori-motor processes of the user?

1.6. What is Usability?

In recent years system 'usability' has become an increasingly important topic for discussion. While the use of a single term to encompass many aspects of human-computer interaction may be considered neither meaningful nor helpful to the system designers (Stevens, 1983) it nevertheless concentrates attention on the most critical aspect of design and is therefore useful to our conceptual representation. Usability is defined as:-

> 'appropriate development of system utilities to fulfil the needs of the users and match functionality to the cognitive capabilities of the individual.'

Usability should also include motivation since motivation provides users with the incentive to master the system, overcoming obstacles and increasing search effort until eventually becoming a local expert. Figure 1.1 shows the role of usability in human-computer interaction. In this diagram utility represents the functions of the system affecting interaction.

1.7. Aims of the Investigation.

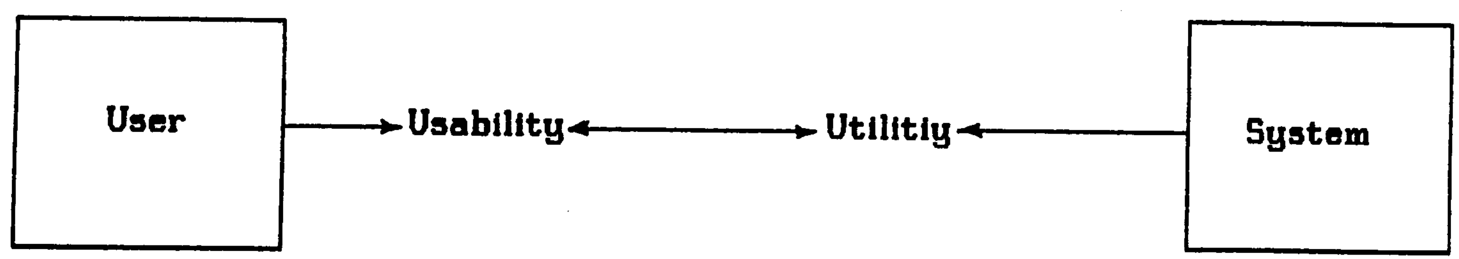
The aim of this research is to design and develop additional facilities to the existing command processor of the FEMVIEW system and to test the use of these facilities on end users. An evaluation of the 'usability' of the system will be based on an analysis of both qualitative and quantitative measures of user and computer performance.

1.8. Structure of the Thesis.

This thesis discusses three major topics, namely voice recognition, user and interface models, and evaluation procedures. It contains six chapters:-

Chapter 2 gives a descriptive account of how a prototype system was developed for the project. A human factors walkthrough highlights some of the practical problems associated with applying speech input to existing software as well as the benefits of this method of data entry. It concludes with an innovative view of the next generation of speech recognition systems.

Figure 1.1. The intervening role of usability.



From Eason (1984).

Chapter 3 lays the ground work for Chapters 4 and 5. It provides an overview of current techniques used in evaluation and uses these techniques in three related experiments:-

(1) A multi-user survey sent to users of the CAD system.

(2) An in-house survey of students learning to use the system, using an on-line logging facilities. (Students were also asked to make their own evaluations and their collective results are summarised.)

(3) An evaluation using experienced engineers with varying knowledge of the software.

Chapter 4 presents the idea that personality is reflected in human-computer interaction and that the study of personality characteristics can provide designers with a better understanding of the users. A comparison is made between user's performance in a benchmark test against expected performance based on their individual personalities (as predicted by a personality questionnaire).

Chapter 5 draws on the experiences of using different evaluation techniques to question the validity of established techniques and provide a framework for evaluating 'usability' in computer systems. It suggests that evaluation is an integrated part of design and the iterative use of evaluation techniques throughout the design, implementation and post-implementation life-cycle of software provides the key to successful systems.

Finally, Chapter 6 reviews the project in the light of experience and suggests an agenda for future research activity. This includes the future of evaluation techniques and voice systems in developing 'usable' human-computer interfaces. It also speculates on the use of embedded user models based on personality traits within self-adaptive interfaces of the future.

1.9. Acknowledgements.

There are many people to whom I am indebted and without whom this research project would not have been possible.

Firstly I must thank my supervisor Dr. Brian Bramer who has always been available to help and give advice when it was needed and who has shown patience and consideration when things haven't gone as smoothly as expected.

Secondly I would like to thank Professor Andrew Parkin who has had this project forced upon him after my previous supervisor Professor Ernest Edmonds moved to Loughborough University, but who has taken time and effort to become involved. He has been an invaluable source of inspiration.

My thanks also to:-

Tony Clarke (LUTCHI[*]) for his help throughout this project, particularly with the psychological aspects of this research;

Dave Porritt (FEMVIEW Ltd.) and the other members of the FEMVIEW organisation who have given much of their valuable time to help in this research;

George Thompson (Automated Engineering Division, Rugby) for demonstrating how FEMVIEW is used within their organisation;

George Henderson (Lloyds Registrar of Shipping, London) for demonstrating the companies use of voice recognition technologies;

Dr. Peter Innocent (Leicester Polytechnic) for his help with parts of the analysis;

Dr. Al Humrich (LUTCHI*) for many hours of proof reading and valuable advice;

to everyone who took part in the experiments for providing the data that forms the basis of this thesis.

Finally, I'd like to add a special thanks to all my work colleagues, (especially those with whom I share an office) for their encouragement throughout the project.

* LUTCHI - Loughborough University of Technology, Computer-Human Interface Research Centre.

Chapter 2

Pilot Study of Implications of Voice Recognition in Computer Systems.

Chapter 2.

Pilot Study of Implications of Voice Recognition in Computer Systems.

2.1. The Need for Research into Applications of Voice Recognition Technologies.

In recent years, much research effort has been put into the design and development of voice recognition technologies. Largely as a result of advancing chip technologies (Burger,1984), the performance/price ratios of voice systems have improved resulting in their increasingly widespread availability. Manufacturers of these systems are quick to highlight the benefits of this additional medium for human-computer interaction in typically 'hands-busy' applications (Anderson et al, 1985). Current recognition rates (i.e. the number of correctly recognised utterances) are usually quoted by manufacturers to be in excess of 95% whereas users of such systems rarely, if ever, achieve this kind of figure. As Biermann et al (1985) have pointed out, this difference is due to to the fact that the manufacturer's data are usually collected by having users read lists of words under optimal conditions. Biermann and colleagues used a speech recognition device in a problem solving task involving spoken natural language and found word error rates in the 8-12% range. Increased availability and such discrepancies between the manufacturer's results and experiences of users provided incentive for further research in this field. Also, with large investments in software packages it is important for software houses to stay at the forefront of technology in order to remain competitive. For this reason FEMVIEW agreed to become part sponsors of the research program.

2.2. Voice Applications.

Applications of voice recognition technologies cover a wide spectrum of industrial areas. These applications have built up in the last few years. A survey made at the start of this research project in 1984 found very few companies using voice recognition in 'live' systems. Today, the literature documents a wide variety of applications for voice systems. To date, however, very little evidence of widespread benefits from these systems has been shown.

Existing 'hands-busy' applications include, for example, cockpit management (Marsh,1984, Logica,1984, Cooke, 1984), although these systems are still in the experimental stage (Laporte, 1985). Several engineering applications are also potentially good customers for voice systems, such as the transcribing of complex engineering drawings into the computer for analysis (Bramer & Rickett,1984); and stock control systems (Ashton,1985, VSI,1986), both of which currently require extra manpower.

Other areas of industry, such as within the health service, can obviously benefit from technologies of this type, in particular as aids for both the physically and voice handicapped (Anmed,1985, Brundage,1985, Joost & Petry,1979, Petry & Joost,1981, Rodman,1985).

In the field of telecommunications, where communication is restricted to sound frequencies in the vocal range, voice systems provide a preferable medium for communications since they remove the need for specialised hardware such as modems. Voice systems are now being used in the USA for banking services via the telephone (Maguire,1984) while in this country systems are being developed along similar grounds, e.g. vcice-controlled dialogues for multi-feature office telephone systems (Leiser & Alberdi, 1987).

Within office environments, typical applications include word processors containing vocabularies of 300-10000 words (Goldhor,1985, Kurzweil,1985). The introduction of keyboard emulation software provides voice recognition capabilities to all keyboard based applications. However, this method of implementation has serious drawbacks in all but the simplest of dialogues (see section 2.7.4.1 - software modifications for the adaption to voice).

A variant of the 'hands-busy' problem commonly associated with CAD packages is the 'third-hand' problem. Commands are typically made up of system directives followed by cursor positioning using a peripheral device such as a mouse, lightpen or tablet and pen. Voice recognition can therefore remove the need for users to transfer between the peripheral device and the keyboard. The FEMVIEW system uses this style of dialogue and is therefore potentially a good application for this additional input medium.

2.3. The Prototype System.

The first task of this research project was to develop a prototype system based around the existing FEMVIEW software. To enable the use of FEMVIEW on many different computer systems and graphics displays, the software uses keyboard entry of commands which follow a strict command syntax. The user is prompted for command input by the display of a menu and message via the command processor (Bramer, 1983, 1984). Options from the menu can then be entered individually or as a command sequence (i.e. command words from a sequence of menus separated by spaces), the command line terminator being the carriage return key. To simplify user interaction and to enable experienced users to enter command sequences very quickly some specific techniques are available:-

(1) The entry of command words using a minimum or shorthand typing facility enables the command processor to uniquely identify a single command from the menu. For example, to select STOP from the menu:-

STOP
CATALOGUE
CATEGORISE

only the character S need be entered. However, to select CATALOGUE would require, for example, CL to differentiate it from CATEGORISE (e.g. CE).

(2) Once a command sequence has been entered, default paths through the command structure are set up enabling the user to enter only the command words that change in successive sequences. For example, suppose the user enters the command (command words are shown in full):-

DISPLAY NODES FROM 50 TO 100

to display on the screen a sequence of nodes from the model. To change the upper end of the node display the following could be entered:-

TO 200.

In this case the sequence 'DISPLAY NODES FROM 50' would be assumed. Care must be taken to ensure the command entered does not appear in an earlier menu in the default sequence.

Facilities which could confuse a new user, such as the default searching technique of 2 above, are normally switched off. Experienced users can switch these on to improve interaction.

In the prototype evaluation system the FEMVIEW command processor, which was implemented on an Apollo Domain DN300 workstation, was upgraded to accept commands not only from the keyboard, but also:-

(1) By positioning a cursor over the required command word in the menu (moving a tablet pen, mouse or touch pad) and hitting the button to select the command.

(2) By typing an integer number corresponding to the position of the required command word in the menu (e.g. to select CATALOGUE in the above menu the number 2 would be entered). Note: if the menu included integer or real number input this facility was automatically disabled.

(3) By accepting character command strings, identical to those from the keyboard, from an RS232C serial line.

The latter method of data input enabled a voice input device to accept spoken commands, convert these to the equivalent character string (corresponding to a keyboard entry) and then transmit these over the serial line to be processed by the command processor. The voice input system used was a VOTAN VPC 2000 IBM PC-compatible bus plug-in card attached to an Olivetti M24 microcomputer (see Figure 2.1). A program in the Olivetti controlled the VOTAN card and handled the RS232C serial link (Figure 2.2). This mechanism was selected as the simplest way to attach a voice input system to the Apollo Domain DN300. The development of the new Apollo Domain DN3000 with its integral IBM PC/AT-compatible bus could enable a suitable plug-in card to be connected directly. In addition, the modified command processor included a built-in monitor giving statistical feedback on objective measures of users' performances during an interactive session. The monitor recorded, at a keystoke-level of analysis, all the computer's activities during each interactive session, which included times for each keypress, mouse movement, and voice entry. This raw information could then be examined by other programs and statistical packages to determine details of error rates, system/user response times, use of help facilities, overall command usage, task completion times, etc. The monitor allowed easy access to a variety of information and proved useful throughout the research project. More details of its use are given in Chapters 3 and 5.

2.4. Selection Criteria for a Voice Recognition System.

When examining commercially available voice recognisers, selection was based cn the following criteria which were felt necessary to fulfil

Figure 2.1. System Layout.

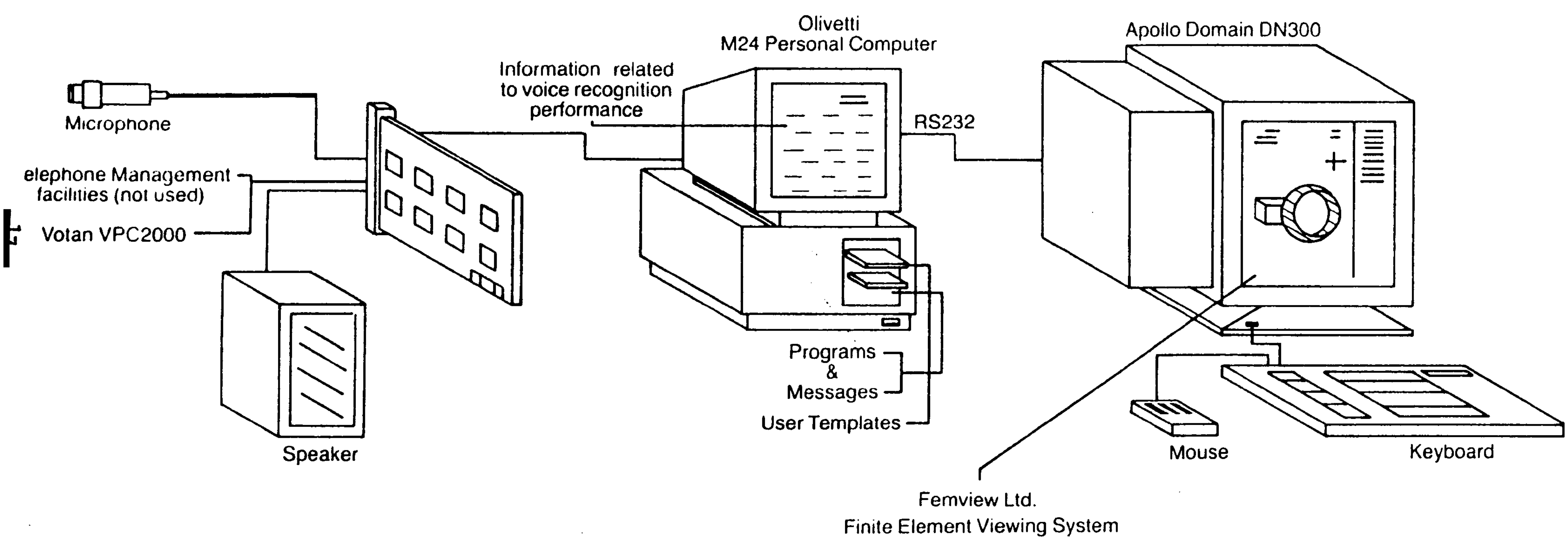
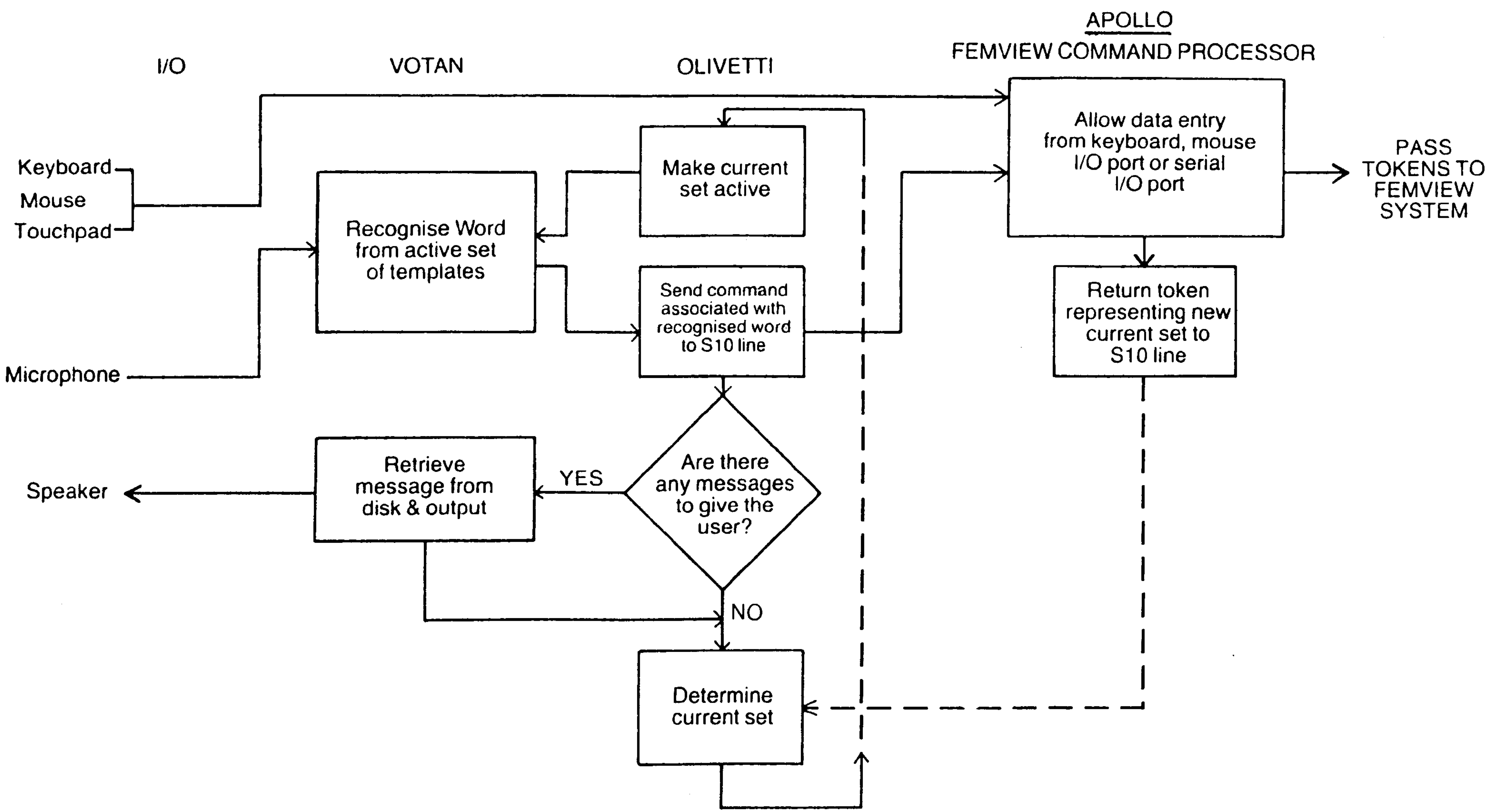


Figure 2.2. System flow.



the overall objective of the research program:-

(1) continuous (or connected) speech recognition;

(2) good performance/price ratio;

(3) software flexibility (i.e. the programmer must have software control over the active set of voice templates);

(4) a total vocabulary size sufficient to hold the full range of commands used within FEMVIEW (i.e. approximately 200 words);

(5) an active vocabulary size (i.e. the number of word choices available at any one time) greater than the maximum number of options available at any point in the FEMVIEW dialogue (i.e. approximately 20 words);

(6) a 'good' interface between the hardware and software.

2.5. Voice Recognisers Evaluated for the Prototype System.

The commercially available voice recognisers examined in detail for this prototype system are listed below with a brief overview of each. Other systems were also examined but were rejected either because they were not continuous recognisers or they were not available within the UK. All costs shown are based on 1984 figures. Although this information is ephemeral, it helps to show the constraints on the research.

Votan VTR 6000. This is a general purpose voice terminal that connects to any computer supporting an RS232C interface with XON/XOFF protocol. It provides continuous speaker dependent voice recognition. It can be configured either as the primary I/O device (replacing the keyboard) or incorporated into an existing computer configuration. Voice key firmware provides a method by which existing programs can be driven by verbal commands without the need for software modifications. It has a maximum vocabulary size of approximately 200 words, with up to 75 words active at one time. (cost £4 500).

Votan VPC 2000. This device provides similar capabilities to the VTR 6000 but is an IBM PC-compatible bus plug-in card (cost £3 100).

Marconi SR-128XX Speech Recogniser. This system provides connected speaker-dependent recognition for up to 240 words. It is a standalone system that connects to the host in the same manner as the VTR 6000. Loading of voice templates is achieved via a built-in mini-cassette recorder and recognised words are indicated on a 40 character plasma display mounted on the front panel. (The retail price of this system was approximately £10 000 although an educational discount was discussed.)

Logica's LOGOS I, II and III. LOGOS provides continuous speaker dependent recognition with a user-programmable word syntax. The system is controlled by a VDU or host computer. LOGOS offers a maximum vocabulary of between 120 and 600 words, with between 20 and 240 words active at any one time. LOGOS III contains an EPROM for resident storage of voice templates even after power down. Since each system is custom made for the buyer the system is very expensive. (LOGOS I costs between £18 000 and £50 000, depending on the chosen configuration, and LOGOS II and III were approximately £10 000 each).

2.6. Coding the Dialogue.

The design of the command processor was such that the structure of the dialogue was loaded into the processor at the start of each interactive session. This allowed the dialogue to be altered, or the command processor to be used in different applications, without source code modifications. This idea was adopted in the design of the dialogue for the voice recognition system. A program was written to interpret the FEMVIEW dialogue from the input file used by the command processor and from it, to generate the program to run on the voice recognition unit. Each keyword in the file became a voice template and each menu list became a set of voice templates. The main program in the voice recognition unit then worked in much the same way as the command processor, recognising a single word utterance from an active set of voice templates, with the next active set being determined by the particular utterance and the currently active voice set. The following shows how this is achieved in practise by taking a simple dialogue and working it through to the voice program:-

Command definition.

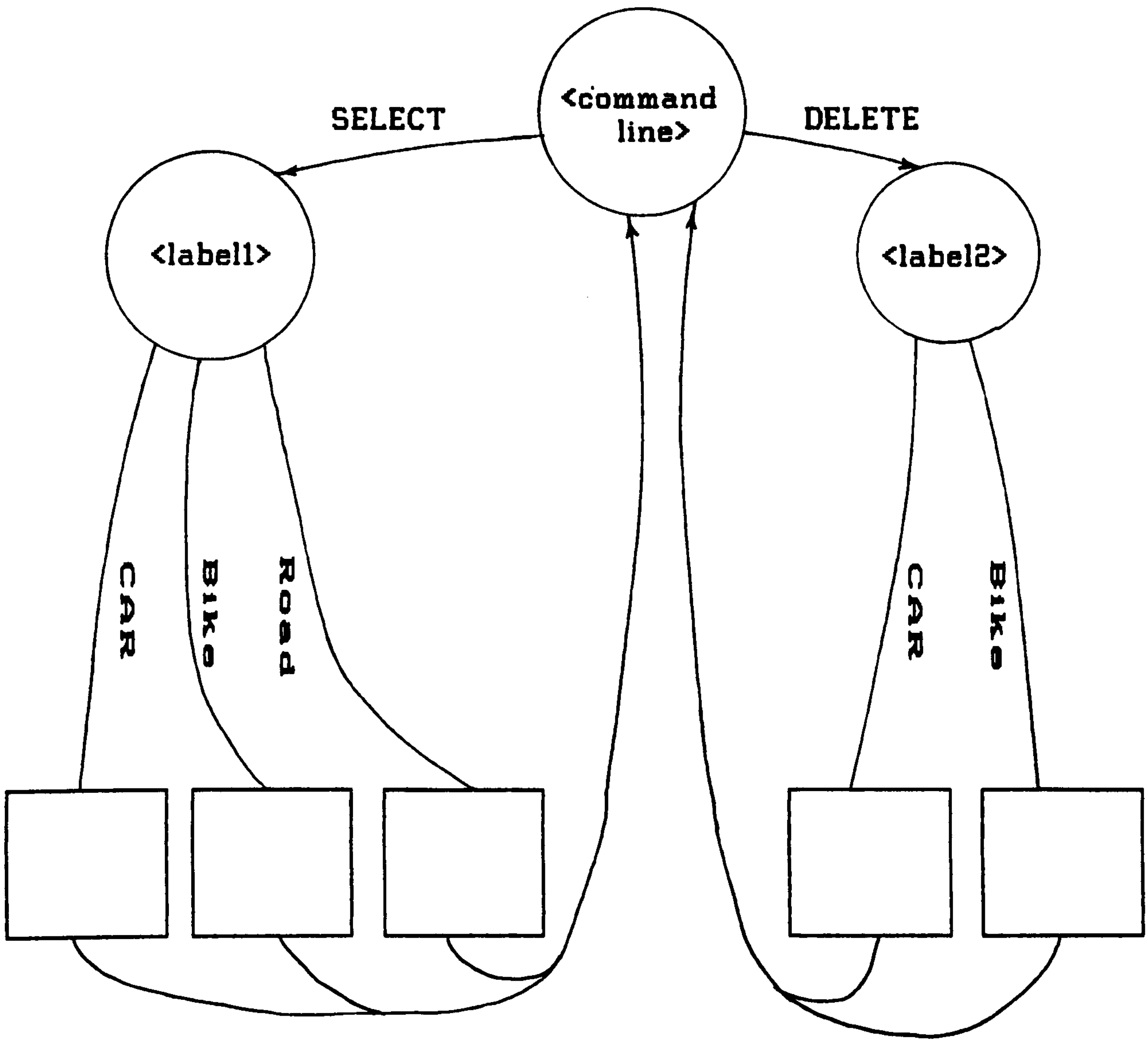
<command line> ::= SELECT <label1>
DELETE <label2>

<label1> ::= CAR
BIKE
ROAD

<label2> ::= CAR
BIKE

Figure 2.3. shows this formal grammar in the form of a state diagram (Parnas, 1969). The file format required by the command processor

Figure 2.3. Dialogue syntax in state graph notation.



n.b. No error states are required since recognition is restricted to keywords in the active set only.

Figure 2A. Describing the dialogue.

```
Line     File contents
 no.
  1    ; Command menu (comments only)
  2    1    SELECT
  3    2    DELETE
  4    ; label menu
  5    3    CAR
  6    4    BIKE
  7    5    ROAD
  8    /
  9    ; Menus follow
 10    ; Main menu no.
 11    1
 12    ;keyword/ destination menu
 13         1          2
 14         2          3
 15    /
 16    ; label1 menu no.
 17    2
 18    ;keyword/ destination menu
 19         3          1
 20         4          1
 21         5          1
 22    /
 23    ; label2 menu no.
 24    3
 25    ;keyword/ destination menu
 26         3          1
 27         4          1
 28    /
 29    /
 30    ; End-of-file
```

Figure 2.5. Convering the dialogue into voice templates and recognition syntax.

```
£def dialog example1
                                                              ! From lines
£def template   SELECT
£text           SELECT     comm="SELECT"
£text           SELECT response="SELECT 001 002"             ! 11 & 13
£text           SELECT   prompt="Say: SELECT"

£def template   DELETE
£text           DELETE     comm="DELETE"
£text           DELETE response="DELETE  001 003"            ! 11 & 14
£text           DELETE   prompt="Say: DELETE"

£def template   CAR
£text           CAR     comm="CAR"
£text           CAR response="CAR  002 001   003 001"        !17&19,24&26
£text           CAR   prompt="Say: CAR"

£def template   BIKE
£text           BIKE     comm="BIKE"
£text           BIKE response="BIKE 002 001   003 001"       !17&20,24&27
£text           BIKE   prompt="Say: BIKE"

£def template   ROAD
£text           ROAD     comm="ROAD"
£text           ROAD response="ROAD  002 001"                ! 17 & 21
£text           ROAD   prompt="Say: ROAD"

£def set M001 SELECT DELETE       !from lines 11(001), 13 & 14
£def set M002 CAR BIKE ROAD       !from lines 17(002), 19,20 & 21
£def set M003 CAR BIKE            !from lines 24(003), 26 & 27
```

to implement this dialogue is shown in Figure 2.4 while Figure 2.5 shows how this input file is subsequently converted into voice templates and dialogue syntax.

The example shows that the active set of keywords, at any point in the dialogue, is determined by the different sets (M001 to M003 in the program). The dialogue syntax is written into the response line of each word template. In this example, if M001 is the active set and the recogniser hears the word SELECT, the response line says:-'IF the currently active menu is (M)001 THEN make (M)002 the new active menu.' Since the same keyword may have a different syntax when associated with another menu, for each occurrence of a keyword more information is added to the response line. Therefore in the case of the keyword BIKE the response line is as follows:

£text BIKE response="BIKE 002 001 003 001"

This says 'IF the currently active set is (M)002 and the word BIKE is recognised THEN make set (M)001 active ELSE IF the active set is (M)003 make (M)001 active.' If, however, the syntax was such that after SELECT BIKE an identification code was required, the options of the identification code being in menu 4, the response line would read:

£text BIKE response="BIKE 002 004 003 001"

Providing software to develop the voice program based on the information given to the command processor could maintain voice recognition capabilities even after dialogue alterations. It also means that any future systems with this type of hierarchical dialogue could easily have voice recognition added with the minimum of programming effort. Also, by automating the generation of the program to control the voice card, syntax errors are avoided. This proved to be highly desirable in the prototype system since compiling the voice program took in excess of 1 hour.

2.7. Evaluation of the Prototype System.

To evaluate the voice system, three approaches were adopted. Firstly, objective measures of speed and error rates were calculated using the prototype system. This was followed by subjective measures based around a favourability scale questionnaire. Finally, the third evaluation technique, known as 'expert-based' evaluations or human factors walkthrough employed the expert knowledge and intuition of the author.

2.7.1. Objective Performance Measures.

To collect quantitative data about speed of voice versus keyboard and mouse, a simple experiment was conducted which involved four experienced users carrying out a series of simple tasks common to the requirements of the software's everyday use. In these experiments, the dependent variable is the time required to complete the tasks and the independent variable the choice of input device. Subjects were asked to complete the task using each device separately. This within-subjects design enabled a comparison between conditions within each subject's scores to be made. The figures obtained from these experiments are shown in Table 2.1.

Table 2.1. Speed of operation using keyboard, mouse and voice input.

	Keyboard	Mouse	Voice
Subject 1	109	174	168
Subject 2	102	198	170
Subject 3	144	205	204
Subject 4	140	211	196
Average time	124	197	185
Est. of variance	21	16	18

Note: All times shown in seconds. Average times are taken from the four subjects.

To ensure that the input devices were the sole independent variable for this experiment the benchmark tests were very simple. It consisted of four tasks involving:-

- Model selection
- Zooming and rotation
- Material identification
- Model subsectioning
- Presentation of loading results.

Each subject performed the benchmark once using each of the three input devices. The order of devices was varied between subjects to reduce the effects of increased familiarity when repeating the exercise several times. Extraneous factors affecting the overall time to complete this benchmark were the typing speed of each individual, the time taken to read the benchmark and the network load at the time of the experiments. As with many other studies of this kind, the results from these experiments are very much software dependent and should not be generalised outside the environment of this particular piece of software. Support for this statement can be seen from the variety of differing claims that have been presented by various authors when comparing voice input with other input devices:-

Leggett and Williams (1984). Twenty subjects entered and edited segments of program code using a speech input device and a keyboard input device. Using keyboard entry 70% of the tasks were completed compared to 50-55% using speech.

Poock (1982). A comparison between speech input and typed entry for command and control operations (e.g. logging into different host computers, reading messages, deleting files, transferring files, etc.). Speech input, in this case, was 17% faster than typing.

Visick, Johnson and Long (1984). This study compared speech and keyed input devices for entering the destinations in a parcel sorting task. The keyboard device consisted of some 50 labelled keys, one for each destination. When users' hands were busy at the sorting task, speech yielded a 37% improvement in entry time.

Martin (1987). A VLSI chip design package had speech input capabilities added. Experiments were conducted which involved subjects completing a set number of task problems in a restricted time. Users were able to complete 62% of the tasks when speech input was available, and only 38% when speech input was not available. Other experiments using the same system directly compare speech input against mouse clicks, single keypresses, and full-word typed commands. These results are summarised in Table 2.2.

The most significant result that can be shown by the study of speed of operation using the prototype system is that for experienced users the minimal keying option provided by the command processor (see section 2.3) makes this method of data entry some 33% faster than voice input and 37% quicker than the mouse. Voice input proved slightly faster than mouse entry because, like keyboard entry, it allowed users to type ahead of the screen. Supportive evidence of these observations can be shown using the t-test to compare keyboard entry with voice input (the quicker of the other two devices):-

Null Hypothesis (Ho): There is no significant different in speed of operation between the two input methods.

Table 2.2. Average times to complete a command sequence (seconds).

Speech vs. Mouse:		
speech input	mouse	advantage
6.05	6.03	0%
Speech vs. Single Keypresses:		
speech input	single keypresses	advantage
5.47	6.81	24%
Speech vs. Full-word Typed Commands:		
speech input	full-word commands	advantage
5.47	11.36	108%

From Martin (1987).

Alternative Hypothesis (Hi): Keyboard entry is faster than spoken inputs.

One tailed test, significance level = 0.01

Degrees of freedom (m + n - 2) = 4 + 4 - 2 = 6

Square of (s^2)= ((n - 1)s_{kbd} + (m - 1)s_{voice})/ mn
variance

$$s^2 = (3 * 402.8 + 3 * 325.1)/ 4*4$$

$$s^2 = 363.9$$

therefore s = 363.9 = 19.1

$$t = \frac{(\overline{voice} - \overline{kbd})}{s ((m + n)/ mn)}$$

$$= \frac{60.95}{13.49}$$

$$= 4.52$$

From the t-tables,

t(1%) 6d.f. = 3.71

Since the value obtained from the data is outside the acceptable region, we can reject Ho in favour of Hi and conclude that there is strong evidence to support the claim that (in this application) keyboard entry is faster than speech input.

2.7.2. Error Rates.

Like performance measures, error rate comparisons between different input devices are extremely misleading. Are transcription errors made while typing as serious as a misinterpreted spoken utterance or an incorrectly located mouse click? The answer to this is very much application dependent. Fortunately, in the case of the prototype system, all operations are 'non-critical' - even deleted projects and models can be fully reinstalled into the database. The only penalty in these instances is the extra time and effort placed on the user. Because of this, no measures of error rates were recorded except to show the effects of increasing the number of word choices on error rates (see section 2.8). As with speed of operation, other literature on this subject provides a variety of differing results:-

Poock(1982) stated that typing produced 183% more errors than speech in the comparison of speech and typed command and control input.

Visick, Johnson and Long (1984). In the comparison of speech and keyed inputs in the parcel sorting task, speech yielded an error rate of 40-80% compared to the keyboard error rate of less than 5%.

Cochran, Riley, and Stewart (1980). Subjects entered the interconnections in a circuit layout (i.e. the netlist) using keyboard and speech input devices. Speech input produced less than 1% errors as opposed to keyboard's 1-5% errors although speech input required a longer time.

Obviously, the wide range of results presented by these different authors indicates the importance of task and ergonomic considerations in the application of voice recognition technologies.

2.7.3. Human Factors Evaluation.

Preliminary evaluations based on objective measures can show whether adding voice to an existing product is viable in terms of reduced manpower or increased productivity. However, human factors play an important role in the success of any voice recognition application. Inconsistent recognition, for whatever reason (fatigue, illness, background noise), will have a negative affect on users' attitudes towards the system and should therefore be minimised wherever possible. While the person who considers spoken input of commands as a beneficial extension to the system will make efforts to compensate for the technology' shortcomings, the opponent is likely to highlight the limitations in an unproductive manner. Acceptance may also be influenced by individuals' personalities in the sense that a self-assured extrovert may adapt readily to a voice system while the apprehensive introvert may find it very disconcerting to use.

A questionnaire was developed to obtain subjective measures of end users' attitudes towards voice recognition both in the prototype system and in general applications. A copy of this questionnaire is given in appendix A1. Unfortunately, because the questionnaire could only be answered following a demonstration of the prototype system the quantity of responses was too small to represent a significant population and hence no conclusions may be drawn. In appendix A2 the results obtained from this survey have been summarised (for reference purposes only).

2.7.4. Human Factors Walkthrough.

In this section the problems and benefits of the use of voice systems in commercial products are discussed. The discussion results from the practical experiences of the author - no empirical evidence is given in support of the claims made.

2.7.4.1. Problems with Voice.

A summary is given of the problems encountered while developing the prototype system to use voice recognition for command input. For a more detailed review of specific problems and their solutions see Rickett & Bramer (1986) and Rickett (1987) (both are supplied in appendix A3).

Suitability of command syntax.

Keyboard based dialogues are designed for precision and minimal typing effort. When spoken, these dialogues are disjointed and unnatural. For example, a command in the software to rotate the displayed model could be:-

EYE ROTATE LEFT 90 <return>.

A format more suitable to spoken input may be:-

ROTATE the model LEFT through 90 degrees.

The introduction of redundant words makes this unsuitable for a keyboard based system. Because there is a direct conflict here between the two modes of input, there are no easy solutions to this problem. The object would be to develop commands that are syntactically and grammatically correct and yet do not introduce redundancy into the dialogue. To achieve this may require extensive dialogue modification and much time and effort.

Distinguishing between linguistically similar words.

Because recognisers cannot distinguish between linguistically similar words, voice oriented dialogues must be developed. In the prototype system some words could only be selected by entering a number representing its position in the menu. In existing systems, changes in dialogue can cause confusion to experienced users and should be avoided where possible.

Large Vocabularies.

Most applications software makes use of a large total vocabulary. For example, the prototype system uses 197 different words plus integer and real number input, filename input, and special control characters and commands. For the system to be totally keyboard independent it is estimated that approximately 300 word templates are required with over 100 different recognisable sets (i.e. command menus). The chosen recogniser did not provide this capability and as a result some commands must still be entered using the keyboard. If applications are to include large vocabularies, the manufacturers of the speech system should be required to show that the system can support the vocabulary.

Training systems for large numbers of words can be extremely time consuming and a major factor in discouraging people from initially using the system.

Software modification for the adaption to voice.

Keyboard emulation provides a means of adapting existing packages to use voice recognition for command input without the need for software modifications. Although appearing to be a very useful feature, for systems that are to be commercially viable this method for interfacing the technologies is not suitable for the following reasons:-

(1) there is no integration between the different input devices (i.e. it is difficult to change between spoken input and other input devices at random), and

(2) incorrect or erroneous data can affect the synchronisation between the active set of voice templates and the current software's options.

Once the synchronisation between the command processor and voice recogniser is lost, recovery can become very difficult, if not impossible. Therefore, for all but the simplest programs, software modifications to enable effective use of the speech system are recommended.

Naive user misinterpretation.

To naive users, a system's ability to recognise voices (without prior training) could be incorrectly associated with the ability to perform more sophisticated procedures, such as natural language processing and reasoning. Voice recognition technologies are therefore not necessarily a means of solving the problem of encouraging 'technology-shy' employees to use computer systems, even though it is arguably a more natural form of communication.

Ergonomic considerations for the use of voice systems.

When using spoken input commands it is important to consider the working environment. Most recognisers are susceptible to background noise and better recognition accuracy can be obtained by minimising this. If this is not possible, then training should be done in the environment where the voice system is to be finally used. Sound proof microphones, such as the type used by radio and television commentators, can also help reduce the effects of background noise.

2.7.4.2. The Benefits of Speech Input.

Provided a would-be designer can avoid the problems outlined above, what general advantages does speech input have? The advantages documented below are those highlighted by the use of the prototype system. Here again, no empirical evidence is given to support these claims.

Suitability to 'non-critical', 'busy-hands' applications.

Voice systems are particularly suited to 'hands-busy' 'non-critical' applications since it provides an additional medium for communication between the user and the computer. High error rates restrict its use mostly to non-critical applications.

Novelty value of voice recognisers.

The 'Hi-tech' image and 'novelty' value to a commercial organisation is of considerable importance for sales and promotion of their products. It is possible then, that justification for the development of a voice medium in existing applications could be made on these grounds alone.

Frees users from the keyboard.

A limitation of most current input devices is the requirement that users sit in close proximity to the screen to operate the system. With voice recognition the only physical device the user needs is the microphone, thereby giving the user much more freedom of movement. This has proved useful for demonstration purposes, where a demonstrator can stand well clear of the display, speak directly to clients and issue spoken commands to the software package. It also allows for remote data entry.

User controls the speed of operation.

Devices that make selections from menu lists or icon displays can be very convenient to the novice or casual user. However, as experience is gained, these methods for human-to-computer communication can be slow and therefore frustrating to use. Voice, like keyboard entry, allows buffering of commands thus giving the user control over the speed of data entry.

Individual Preferences.

For experienced computer users, the poor recognition rates for spoken commands make it an impractical mode of input. For the casual (or infrequent) user, where speed of operation is not a critical factor and high error rates are tolerable, choice of input device can be determined by the preference of the individual.

Obviously, human factors walkthrough's of this kind are influenced by the experiences of the individuals, the equipment used and the experiments conducted. The above discussion has attempted to concentrate on specific issues of general interest.

2.8. Heuristic Knowledge for the Selection of Voice Recognition Technologies in Existing Applications.

In order to provide potential buyers of voice recognition technologies with a simple guide of environmental, application and user considerations, the following checklist has been produced. Four examples of its use are then given. Table 2.3 has been simplified for the purpose of generalisation. In practise, weighting scales could be applied specific to the needs of the application. Advancing technologies will also change the weightings shown in the table. The way to use the table is as follows:

Within each of three categories, namely, environment, application and users, a value is given in respect of the proposed application. For example, if the voice system is to be used in a quiet room a score of 2 is obtained (from 1a), and if the existing application uses a menu

Table 2.3. Speech recognition selection criteria.

1. Environment.			
(a)	Background noise variability.		
	large variability	minimal variability	no noise
	0	1	2
2. Application.			
(a)	Type of data entry.		
	totally locator	direct mode/natural lang.	menu mode
	0	1	2
(b)	Speed of data entry (from Nye 1982).		
	< 1 word per 2 secs	1 word per 2-5 secs	not critical
	0	1	2
(c)	Vocabulary size.		
	> 500	100 to 500	< 100
	0	1	2
(d)	Tolerance to errors.		
	Critical	few critical commands	non critical
	0	1	2
(e)	'Hands-busy' or 'third-hand' applications.		
	no	yes	
	1	2	
(f)	Reductions in manpower utilisation.		
	no	yes	
	1	2	
3. Users.			
(a)	User attitude towards voice technologies.		
	bad	good	
	1	2	
(b)	Number of users.		
	many different users	only a few users	
	1	2	

type dialogue a score of 2 is given (from 2a). The individual scores for each aspect are then added to give a final value in the range 4 to 16.

If the score for any of the first five questions is zero then voice recognition (in the current state of the technology) is probably not viable for the application.

A score between 9 and 11/12 (provided no individual score is zero) would indicate that the addition of speech input would be possible in the application, although few clear benefits are indicated by the table. Therefore careful analysis of the costs and benefits must be considered.

Finally, a score of over 12 (without any zero scores) would appear to show a suitable application for current voice recognition technologies.

To test this table consider four applications:-

(1) Voice recognition for instrument control in aircraft and helicopters.

(2) Voice recognition capabilities in word processors.

(3) The FEMVIEW system.

(4) A system for the on-line collection/retrieval of dental records.

Scores for each of the above applications based on the heuristic

Table 24. Suitability of four applications based on heuristic rules.

	Cockpit managt.	Word processors	FEMVIEW	Dental recording
Environment.				
B/ground noise	0/1*	1	1	1
Application.				
Type of data	2	1	1	1
Speed of entry	2	0	1	2
Vocaby. size	2	0	1	2
Errors	1	1	2	1
'Hands-busy'	2	1	1	2
Reduce manpower	2	1	1	2
Users.				
User attitude	1	1	1	1
No. of users	2	1	1	2
	15	7	10	14

* Research towards compensating for constant background noise is currently being conducted with a view to providing voice control in aircraft environments (Cooke,1984, Marsh,1984).

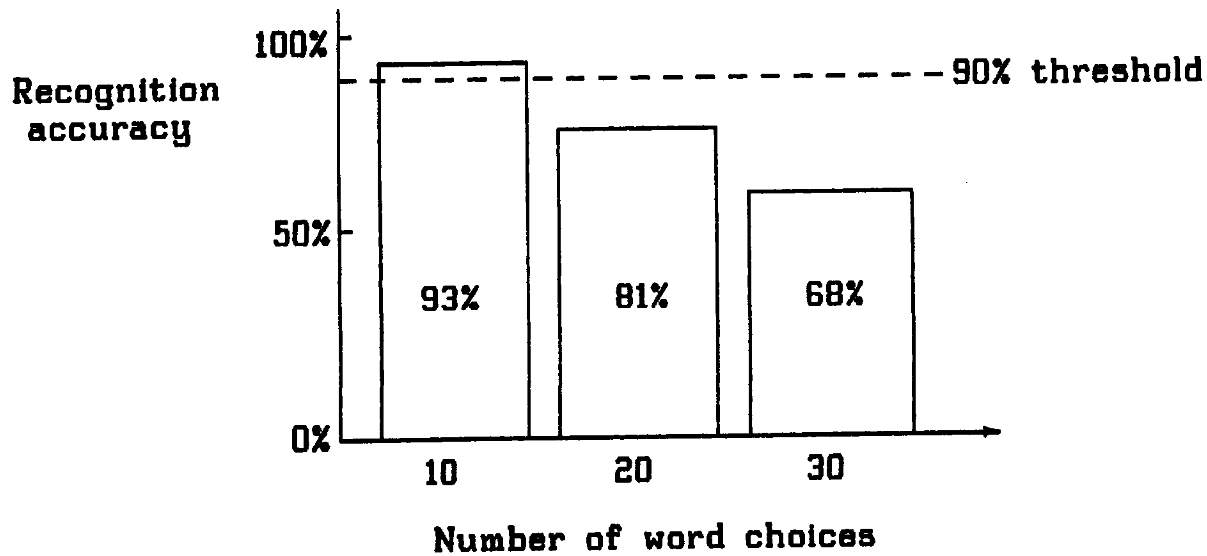
rules presented in table 2.3 are shown in table 2.4.

Cockpit management for non-critical tasks such as radio communications and map reading operations are 'hands-busy' operations with, typically, menu-driven dialogues consisting of small vocabularies. This makes cockpit management a potentially good candidate for voice recognition. However, the biggest problem yet to be satisfactorily resolved are the effects of very high background noises.

Word processing would appear to be an ideal application for voice recognition capabilities and it is towards this type of application that many manufacturers of speech recognition systems are moving (Kurzweil,1985, Goldhor,1985). However, with current technologies there remains a direct correlation between the number of word choices allowed at any one point in the dialogue and error rates. This can be demonstrated by performing the following simple experiment using the voice recognition system:-

To begin the experiment, the numbers 1 to 10 are entered into the speech system and 100 recognition trials made noting the number of correct and incorrect recognitions (no recognition by the system is treated as an incorrect recognition in the experiment). Then the numbers 11 to 20 are added to the original list and another 100 trials conducted. Finally, the numbers 30, 40, 50,..,100, 0 and 200 are added and the experiment repeated once more. Results show how, in this rather exceptional case, increasing the number of word choices has a marked affect on recognition rates. The results of this experiment are shown in Table 2.5.

Table 2.5. The effects of increasing vocabulary size on recognition accuracy.



This graph would appear to indicate that the maximum number of word choices allowable in a system at any one point in the dialogue, under perfect conditions and for a recognition accuracy of over 90% would be about 12 words. However, in this example numbers are used which are one of the most demanding benchmarks for any voice recogniser. (See Waterman, 1985 for an explanation of digit confusion in speech systems.) The use of carefully selected keywords can considerably reduce error rates although the correlation between the number of word choices and error rates will remain. Therefore, based on the heuristic rules presented above, speech input for word processing is not viable.

In the case of the FEMVIEW software there are no advantages to using voice input of data. When voice is used, the speed of data entry is reduced and error rates are increased. Therefore, the capital expenditure and manpower resources needed for its implementation could not be justified (except on research grounds).

Finally, consider the benefits of a voice recognition system in a dental surgery for the collection/retrieval of patients dental records. A keyboard based system would undoubtedly require extra manpower resources to record information whilst the dentist is examining the patient's teeth. Since this type of data collection would typically require simple dialogues with small vocabularies and speed of data entry is not critical, this application would appear to be particularly suited to speech technologies. Singh (1987) has implemented such a system as part of a joint research project sponsored by Leicester Polytechnic and Unilever Ltd.

The heuristic rules offered above are intended only as a rough guide to the casual reader with the purpose of illustrating how the limitations of current speech recognition technologies will affect its applications. It should be stressed that potential customers of these technologies should make independent feasibility studies based on the requirements of the particular applications and resources available. This is a greatly expanding field for development and in time the limitations currently associated with speech systems and documented above will diminish and the potential for voice controlled applications will greatly increase.

2.9. The Next Generation of Speech Recognition Systems.

The above discussion has concentrated on the application and evaluation of current speech recognition technologies. With existing system methodologies for speech recognition it may seem that the image presented by Hal (in the film '2001: A Space Odyssey') of computers which can communicate intelligently with humans would seem a long way off. In practise, the technology for such systems is already available. Larry Harris, founder of the journal, ' Artificial Intelligence' believes that voice recognition technologies combined with natural language processing provides the key to future speech systems. He states:-

> 'The combination of voice with natural language is likely to have the same impact on end-user computing that the addition of sound had on the motion picture industry. In combining the two technologies, the science fiction image of Mr Spock speaking to his computer and getting an answer is, for the first time, within the vision of the foreseeable future.'
>
> (Quoted in Killmon, 1986)

Natural language understanding, which has now come under the umbrella of artificial intelligence, is difficult for computer simulation, the three major factors for this being (from Rich, 1983):-

(1) The complexity of the target representation into which the matching is to be done. This implies that to extract information from English sentences often requires the use of additional knowledge about the world described by the sentence, such as in the following story:-

> Bill told Mary he would not go to the movies with her.
> Her feelings were hurt.

(2) The type of mapping: one-one, many-one, one-many, many-many. Very few languages provide totally one-to-one mappings from statements or sentences to single target representations of the information they contain (for use by the computer). English has a many-to-many mapping, in which there are many ways to say the same thing and a given statement may have many meanings. To implement such mappings in computer programs requires a great deal of both linguistic and non-linguistic knowledge.

(3) The level of interaction of the components in the source representation. In most languages changing a single word in a sentence can alter its entire structure and meaning. The interaction of each component (word, symbol, or whatever) of the statement must therefore be mapped with consideration to the other components in the statement.

There are presently several working natural language understanding systems (Weizenbaum,1966, Wilensky,1982, Harris,1978). However, much research into effectively overcoming these problems is still required.

Traditional approaches to speech analysis, adopted by most of the early systems and still used in many systems today, involved storing word templates by saving audible frequency values over discrete timeframes. Recognition of a spoken word with a pre-stored template is determined by a comparison of the two, using one of a variety of different algorithms. Examples of the pattern matching paradigm are given in Velichko & Zagoruyko (1970), Itakura (1975) and Pols (1971). While these methods of recognition remain, recognisers will continue to be speaker dependent and have high error rates. Work by Johnson et al (1983) suggests that large vocabulary, speaker independent systems can be achieved using a phoneme based approach to recognition. They have used Prolog rules to extract phonemes from spectrogram printouts. Acoustic phonetic recognition is not new and literature relating this approach to voice recognisers was presented as early as 1974 (Broad & Shoup,1974, Cohen & Mercer,1974). One of the most successful speech-understanding systems in existence using this type of technology is HEARSAY-II (Erman et al,1980). This methodology for recognition, when linked to large phoneme-based dictionaries, could solve many of the problems associated with current recognisers.

To accommodate both advanced natural language processing capabilities and phoneme-based recognition algorithms high powered, multiprocessing machine environments will be required. Workstations with large amounts of processing power (e.g. Apollo Domain and Sun workstations) are becoming more commonplace in industrial environments yet even more powerful, sophisticated machines are required.

Finally, if the overall goal is to produce system interfaces of the type shown in science fiction novels where the users participate in intelligent, and often not so intelligent, conversations with the computer then the last component of the new technologies will be large intelligent knowledge-based systems capable of building real world knowledge from conversations by inference and deduction, in much the same way as humans do. A system using this type of probabilistic reasoning within the limited domain of therapies for patients with bacterial infections is the MYCIN program (Shortliffe,1976). MYCIN uses rules to reason backwards to the clinical data available from its goal to find significant disease-causing organisms. Once it has found the identities of such organisms, it attempts to find a therapy by which the disease(s) may be treated.

2.10. Conclusion.

Voice systems are available that can be used to enhance existing commercial software packages. Using current technologies, not all packages would benefit from voice input and careful consideration must be given to the suitability of existing applications. The success of voice systems is dependent on the application, the environment and users' attitudes which accounts for large variations in results from objective evaluations of different systems. As with all peripheral input devices, implementation (where possible) should be integrated with other input media giving users the option to choose which method to use based on suitability and preference.

The author has produced two papers outlining the practical problems associated with developing systems to use voice for the input of commands (Rickett & Bramer, 1986 & Rickett, 1987). Both these papers can be found in appendix A3. They are based on the practical experiences of the author and act as useful reference documents for potential developers of such systems.

Chapter 3

Developing Techniques for the Effective Evaluation of Man–computer Interactions.

Chapter 3.

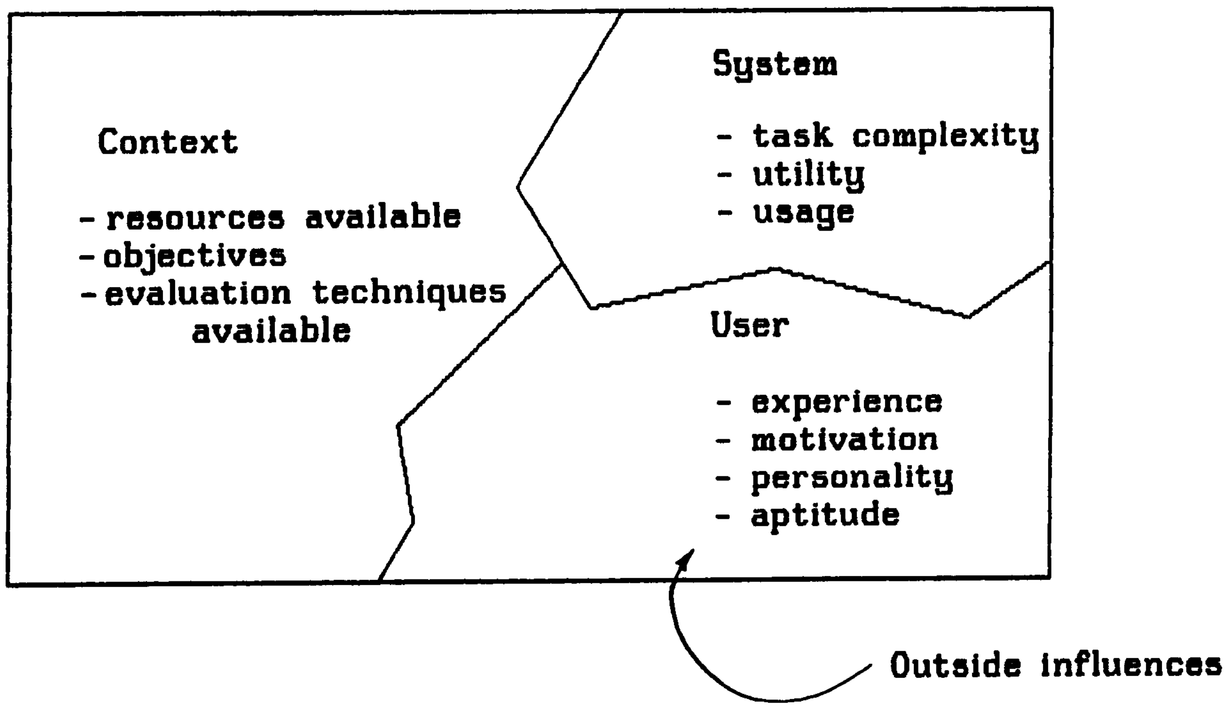
Developing Techniques for the Effective Evaluation of Man-computer Interactions.

3.1. Introduction.

In the past, approaches to evaluating man-computer interactions have been based on quantitative measurements, designed and conducted by computer scientists. Although many of these evaluations produced valuable feedback to software engineers about the functioning of their software, the experiments were limited by the very nature of the approach taken. In many of these cases, the evaluation team had an incomplete picture of the environment in which they were working. The key to successful evaluations stem from a complete knowledge of the environment. This includes not only the system design but also the intended users and the tools available for studies of this nature (see Figure 3.1).

The traditional methods for evaluation, namely speed of use and error rates, were favoured by evaluators because they produced easily quantifiable results. These experiments concentrated on a very small part of the totality of the system and often failed to match findings with actual system usage. For example, a software package may be extremely difficult to use, slow and error prone, but may be successful because of users' motivations and desires to overcome the obstacles. New techniques for evaluation have moved away from the computer-centred evaluation towards a user-centred approach. Since it

Figure 3.1. The evaluation environment.



is this user, not the designer, who will spend many hours and often many years using the computer system, it is towards him that any evaluation claiming to investigate 'usability' (as defined in Chapter 1) must be aimed. Despite the advantages, to date, few conclusive evaluations have adopted this user-based approach because of the difficulties in producing quantifiable data for analysis. This chapter examines current methodologies for system evaluation.

Having implemented the prototype system as described in section 2.3, it was discovered that few effective methods for evaluating this system were available. Furthermore, a literature search into evaluation techniques revealed that few established methodologies for evaluating the 'usability' of computer systems generally existed.

To understand more about evaluation procedures, two experiments were carried out and a multi-user survey of users' attitudes towards the software was conducted (i.e. using production and design engineers with various experience of the software). The first experiment examined how naive users learned to use the software. The experiment was conducted by giving the software to students and then asking them to make their own evaluations of the product (see section 3.7). The second group of experiments involved experienced engineers taking part in a series of benchmark tests under laboratory conditions. Data collection techniques ranged from traditional reaction times to personality surveys. (The second group of experiments is given more detailed coverage in section 3.8.)

3.2. Different Approaches to Evaluation.

Howard & Murray (1987) suggest that there are five main types of formal evaluation, namely:-

- expert-based
- theory-based
- user-based
- market-based
- subject-based.

An expert-based approach, or human factors walkthrough, has two major components - an expert and a system. The domain expert employs expert knowledge, scientific principles and intuition to evaluate the interface. Human factors walkthroughs can be very useful at all stages of design but the data collected may be incomplete and open to bias.

Theory-based evaluations consist of a model of the user and a model of the interface. Mapping relationships between formal representations of both user and system are examined with the view to identifying any mismatches.

The user-based approach relates to a personal evaluation of a system by the user and is reflected in terms of patterns of system use.

A market-based evaluation is usually conducted by organisations to examine the success, or otherwise, of a product on the market before making a purchase.

Subject-based evaluation techniques are perhaps the most widely used method for evaluation. The main components are metrics, subjects, tasks and systems. Usually subjects are observed performing various tasks under laboratory conditions.

In section 2.7.4 a human factors walkthrough produced important information relating to the use of voice recognition systems in commercial software, although in this instance, it proved difficult to support intuitive observations with quantitative measurements. A theory-based approach is adopted in Chapter 4 when both user and interface models are examined in more detail. In this chapter the discussion concentrates on subject-based evaluation techniques.

3.3. A Subject-based Approach to Evaluation.

In subject-based evaluations data is recorded at four levels:-

- physiological
- behavioural
- cognitive
- affective.

At the physiological level, visual scanning patterns have been recorded using electro-myography (Howard & Murray, 1987) and/or video recordings. The latter was used by Martin (1987) in a comparison of speech input and keyboard entry for VLSI design. The video method showed that users who did not have speech input available spent more time looking back down at the keyboard. Martin used this to argue that speech input lessens the users' workload, in the sense that the users do not have to glance down at the keyboard so frequently. Video recordings show the activation of users' motor processes and provide a more complete picture of the process of interaction and thus physiological measures are an important part of an evaluation.

Behavioural data encompasses the traditional objective measures and include system/user response times, error rates, task correctness, system use patterns, learning ability, and so on. The most used and widely documented measures of evaluation are those of response times and error rates (Kidd, 1982, Monk, 1985). Their continuous use even today indicates the success of these methods in providing useful information on which to base design decisions. Kidd (1982) points out that system response time is crucially important in interactive dialogue for two main reasons:-

(1) The user will expect a response from a system within certain time limits and will become anxious or frustrated if he/she does not receive a response from the system when expected.

(2) When performing a cognitive task, users have a strong drive for 'psychological closure' - i.e. to reduce the load on their short term memory.

For more information on response times see Shneiderman (1980) and Miller (1968).

User error rates give a good benchmark to 'ease-of-use' and provide a method for spotting potential causes of user dissatisfaction. A study of the types, causes and frequencies of errors was conducted by Norman (1983). Errors can also be seen as a destructive feature when striving towards user 'Rolls' (Brady, 1986). Rolls are described by Brady as a state of mind where the mind efficiently organises data, makes decisions, and executes those decisions precisely.

The most common technique for the collection of behavioural data is the use of on-line logs (Neal & Simons, 1983, Penniman & Dominick, 1980). These logs provide information, at various grains of analysis, about system and users' activity during an interactive session. On-line logging has the advantage that once installed, data collection is ongoing and needs little maintenance. Because it is unobtrusive, subjects soon forget its presence and therefore, data is not influenced by artificial conditions. It does however, have the disadvantage that it provides a basis for asking questions about a system without providing any answers. For example, in the evaluation of system A versus system B a monitor might indicate that subjects' take, on average, 10% longer to achieve goals using system B than system A. Although this may suggest evidence for the preference of system A over system B, closer examination will be necessary to show the causal factors. Nevertheless, when combined with other techniques, on-line logs can be an extremely useful evaluation tool.

Cognitive and affective levels of data include users' attitudes and opinions, their knowledge and ability and patterns of system use. Methods for data collection include:-

- questionnaires
- benchmark tests
- aptitude tests
- interviewing/debriefing
- introspection
- protocol analysis

- factor analytic methods such as repertory grids
- predictability analysis.

3.4. Collecting Data at the Cognitive and Affective Levels: An Overview of Current Techniques.

Questionnaires.

The use of questionnaires as a method for recording difficulties experienced by users and for measuring user attitudes has always been the subject of much discussion and criticism. Because of the intrinsically subjective nature of questionnaires, the quality of results is limited by the validity and completeness of the user's understanding of his own behaviour and by his ability to verbalise that information. It is also reported that subjective questionnaire ratings often fail to correspond to objective performance measures (Ramsey & Atwood, 1979). (Evidence will be shown in Chapter 5 to support this claim.) This does not reject the use of questionnaires as a method for data capture but suggests that careful consideration must be given to the level of detail extracted by the use of this technique. Some support for this argument can be found in Root & Draper (1983).

Benchmarking.

Benchmark tests have long been used in the comparison of computer hardware. Lewis & Crews (1985) provide a review of the evolution of benchmarking as a computer performance evaluation technique. The term 'benchmarking' in the computer field is generally considered 'a set of executable instructions which may be used to compare the relative performance of two or more computer systems' (Morris & Roth, 1982). Joslin (1965) extended the benchmark idea to give estimates of the actual processing time required to run the estimated workload. He defined these 'application benchmarks' as a mix of routines that run on several different computer configurations in order to obtain comparative performance figures on the capabilities of the various configurations to handle the specific applications. In his paper he warns that the results from application benchmark tests can only be as good as the benchmarks themselves. If the benchmarks are not representative of the workload being tested, the results could prove more harmful than no benchmark at all. In this research program benchmarking has be taken one step further and the concept of an 'evaluation benchmark' is introduced. A typical model of the user's workload at various levels of complexity is presented in the evaluation benchmark. The tasks performed using the software correspond to a simulation of its intended use in the working environment. Tasks are described in the form of broadly defined procedures, whereby the command sequences for achieving the tasks are at the user's discretion. Each task is an important step in achieving an overall goal but the order in which they are described are not necessarily the optimum way of achieving that goal. Users are encouraged to read through each exercise in the evaluation benchmark before starting the interactive session, thus allowing them to develop alternative planning strategies if they so desire. The study of the processes by which the user attempts to solve the benchmark tests can provide important clues to potential mismatches between the perceptual/cognitive processes of the user and the functionality of the computer software.

Aptitude tests.

Aptitude tests have been a popular method in the past for the selection of computer programmers. However, satisfactory demonstrations of their validity do not exist. Aptitude measures include:-

- verbal reasoning
- reasoning
- abstract reasoning
- number ability
- diagramming

(from Palomo, 1987).

Most people are reluctant to take tests of this nature without strong incentives, and to predict user's performance based on a test score would appear rather shortsighted. Since the essential task of our evaluation is to fit the software to the user, aptitude tests which could arguably be a means of selecting users suited to the software are outside the scope of this evaluation.

Interviewing.

Interviewing of system users can be an informative method of gaining qualitative data about attitudes towards the software. It does however, suffer from two drawbacks:-

(1) like questionnaires, the data is not always reflective of users'

abilities, and

(2) the data is open to the bias of the individual users.

Monk (1985) has found that better results can be obtained by debriefing users at the end of an experimental session. He suggests the use of formal questionnaires for use during the debriefing session in order to ensure that the key issues are covered.

Introspection.

Introspection is a method whereby the subject simply reflects on how he/she works. Although it can generate insights and new ideas, introspection is unique to each individual, and the conclusions that one person reaches may not be shared by others (Shneiderman, 1980).

Protocol analysis.

Protocol analysis is a method of recording users perceived thought processes. It is achieved by encouraging users' to talk through their reasoning behind problem solving during an interactive session. It produces qualitative data that can provide a broad survey of phenomena and problems in a task domain (Mack et al, 1983). This permanent record or transcript can be reviewed for behavioural patterns. Carrying out the protocol analysis for substantial numbers of individuals is difficult, time consuming and expensive. Because of this, few evaluators have, as yet, adopted this technique.

Repertory grids.

Repertory grids were first developed by Kelly (1955) as a methodological component of Personal Construct Theory. They allow the examination of relationships between personal constructs within a specific domain. Personal constructs are the interpretation of experienced events by an individual. Beail (1985) provides an overview of the technique with its applications in clinical and educational settings while Shaw (1980) has used the technique in a computing evaluation capacity. While the technique provides some interesting methods of data collection and analysis, producing suitable repertory grids is highly involved and the data can suffer in the same manner as data from questionnaires.

3.5. Multi-User Survey Questionnaire for the Quantitative Measure of Subjective Data.

3.5.1. Aims and Objectives.

The purpose of this questionnaire was threefold. Firstly, it was desired to monitor the overall trends in the use of the FEMVIEW software as an indicator of the acceptance of this package in an industrial environment. Secondly, aspects of the FEMVIEW interface independent of the use to which the software was being made needed to be identified. Thirdly, it was hoped to obtain end-users who were prepared to participate in practical experiments for this research program.

This survey also attempted to:-

(1) study the usefulness of questionnaires as a method for the capture

of data relating to users' attitudes towards interface features;

(2) identify areas of the FEMVIEW software generally considered to be substandard and to report these back to FEMVIEW;

(3) identify aspects of the system generally considered to be desired by the end-users.

Since the total use of this package was restricted to 28 companies throughout the UK, the number of respondents to the questionnaire was expected to be low. This meant that few assumptions about the population could be made. This, coupled with the small sample used in this survey, restricted analysis of the data collected to nonparametric statistical methods.

3.5.2. Questionnaire Format and Layout.

Several criteria were considered when designing this questionnaire. Since there were relatively few companies using the software, it was important to get a high proportion of responses to have enough data with which to work. For this reason the questions were given as multiple choice wherever possible. This reduced the time needed to complete the questionnaire and the amount of effort on the part of the respondents. Another method used to help encourage a high percentage of replies was the careful selection of company employees who were known by FEMVIEW to be co-operative. By directing the questionnaires to companies through these employees a higher proportion of replies could be expected.

The questionnaire contained five sections:-

- company use
- individual use
- the commands
- additional information
- practical evaluation.

Company use was concerned with whether FEMVIEW is still being used within the organisation, how much it is used and who are the users. Individual use looks at the type of users, their experience and an approximate amount of time they use the system. The commands looks at the various aspects of the command processor (or user interface with the exclusion of the graphical displays). It is subdivided into a further five sections:-

- command syntax and semantics
- help facilities
- keyword entry
- menu prompts
- error messages.

The additional information section was provided to allow users to add their own comments, observations or suggestions. Finally, the practical evaluation section asked respondents if they were prepared to participate in any further experiments that may be required during this research project.

Appendix A4 gives an example of the questionnaire while Appendix A5 shows a list of the user's replies to the questions.

3.5.3. Analysis of Responses.

Three copies of the questionnaire was sent to each of the 28 UK companies using the FEMVIEW software. Full permission was granted to make copies of this questionnaire if required. A total of 40 responses were received from 21 companies, providing the raw data for this analysis. The reported users of the software (within the 21 companies represented by the replies received) consisted largely of engineers. The numbers from the other groups of users were not large enough to represent a suitable population:-

Engineers	=	127 (83.0%)
Systems Support	=	4 (2.6%)
Computer Programmers	=	4 (2.6%)
Management	=	1 (0.7%)
Computer Operators	=	1 (0.7%)
Students	=	12 (7.8%)
Others	=	4 (2.6%)
Total number of reported FEMVIEW users in 21 companies	=	153 (100%)

Information obtained from the questionnaire within the various sections that asked for users' comments was collected and feedback given to FEMVIEW relating to the requirements of the end users. Users' replies are broadly grouped into eight categories:-

- input features

- output features
- dialogue design
- error messages
- interface design
- application related features
- help facilities
- training.

(See appendix A6 for a full list of comments.)

In general, areas of the questionnaire requiring written text were not extensively utilised. Five questions from the command section allowed users to show either a favourable or unfavourable attitude towards different parts of the command processor. Results from these questions show a favourable attitude by respondents to the command processor:-

favourable responses	=	108 (58%)
adequate responses	=	57 (31%)
unfavourable responses	=	21 (11%)
Total	=	186 (100%)

3.5.4. Trends in the Use of the FEMVIEW software.

Overall Trends.

One of the main advantages of a questionnaire such as this is that it provides a practical methodology which can give a reliable prediction of the use of the software. Since the primary interest of this experiment is to indicate any significant increase in system use since installation (i.e. an indication of users accepting the software), the most appropriate and powerful test that can apply to this data is the sign test. Question 1.3 of the questionnaire states:- 'Has the regular use of FEMVIEW in your company:- Increase/Remained unchanged/ Decreased?' The analysis of responses follows:

Null hypothesis (Ho) : There is no significant change in the overall use of the FEMVIEW software after installation.

Alternative Hypothesis (Hi) : The use of FEMVIEW has increased since its installation.

One tailed test, significance level = 0.05.

Total no. of company replies = 21.

P = Q = 0.5.

No. showing a positive difference (N) = 15.

No. showing a negative difference (x) = 2.

Table D (Siegal, 1956) N = 15, x = 2 P(1-tailed) = 0.004.

Since this value is outside the acceptable region, Ho can be rejected in favour of Hi and we can conclude that there is evidence to suggest that the use of FEMVIEW has generally increased since its installation in different companies.

Individuals' use of the software.

Having shown using the sign test that the overall use of the FEMVIEW software is on the increase one might expect to see a similar increase in individuals' use. Again, applying the sign test:-

Null Hypothesis (Ho) : There is no significant change in individuals use of the FEMVIEW software after installation.

Alternative Hypothesis (Hi) : Individuals use of FEMVIEW has increased since its installation.

One tailed test, significance level = 0.05.

Total no. of user replies = 39.

P = Q = 0.5.

No. showing a positive difference (N) = 20.

No. showing a negative difference (x) = 8.

Table D (Siegal, 1956) N = 20, x = 8 P(1-tailed) = 0.252.

Since this value is within the acceptable region we can conclude that there is no evidence to suggest that individuals' use of FEMVIEW has increased since its installation. Responses to question 2.3 from the questionnaire, which asks users' to explain any reason for changes in use over time, show the major reasons for a decrease in FEMVIEW use by some individuals to be their changing duties (either because of promotion or work on new projects). It is also noted that the software is only used when it is required by a project and therefore there may be long delays between subsequent usage.

3.5.5. Separation of the Interface from the Application.

From a designers point of view, the separation of the interface from the application has many advantages (Edmonds, 1982). The questionnaire examined features of the command processor as opposed to application specific questions. Within each of the five sections an area was left for suggestions, likes and dislikes. An additional section was also included for information not covered by any of the specific categories. It became clear from the nature of the replies that respondents either were unwilling or unable to make the distinction between aspects of the command processor and the application. This would appear to suggest the need for interfaces to be designed as an integral part of the application. Appendix A6 shows a list of the comments and suggestions made by the FEMVIEW users. (A copy of this was forwarded to FEMVIEW so that suitable amendments could be made to the software where it was felt appropriate.)

3.6. On-line Monitors for the Unobtrusive Collection of Behavioural Data.

The purpose of the on-line log was to provide empirical data about users' interaction with the FEMVIEW software. An identification mechanism was developed whereby each time a user ran the software, a monitor file was created with a unique filename generated by a combination of the users' login name, the current time and the date. This ensured that no monitor data was ever overwritten. It also has the advantage that logs could be sorted and analysed by the user's name (this is possible because each user has his/her own login name), dates and/or time of day. The monitor recorded, at a keystroke level of interaction, the following information:

Identity:-

- user name, the date and time

Input devices:-

- time for each keystroke
- log of all mouse movements
- time for all mouse selects
- time for all voiced entries
- time of all synchronisations with voicecard (not used)
- time for each valid keyword/command line entered

Error Rates:-

- number of backspaces made
- number of line rejects entered

Response Times:-

- initial system set-up time
- execution time for each valid command
- total time in system.

Since such fine grain analysis of users' interaction can produce large amounts of data, it was important to provide a simple means of accessing relevant data within each monitor. To do this a method of identifying each activity by an operator code was adopted. A list of these codes is given in appendix A7. This reduced the amount of effort required for analysing the data later. For example, if the aim is to examine overall system response times for processing the command

VIEW HIDDEN LINE,

a typical monitor for this command may be as follows:

```
        .
        .
        N  029.01  <mouse movement>
        Z  030.12  <mouse stopped>
        M  030.87  <left mouse button>
        C  031.45  LINE
        F  031.67  VIEW HIDDEN LINE
        S  105.67  <return to command processor>
        .
        .
```

(This is an operation that shows the view of a selected model with the parts of the mesh not visible to the human eye removed from the picture.)

In this example, the mouse has been moved over the word LINE in the menu and the left hand mouse button pressed. The command processor has recognised that a valid command has been entered and the full command VIEW HIDDEN LINE has been passed to the software for processing. Indication that the command processor is ready to accept more input is indicated by the S operator.

To calculate the average system response time for this command the following algorithm is applied:-

```
BEGIN
     Initialise variables
     READ a monitor file
     WHILE there are no more files to read
     BEGIN
          REPEAT
               READ 1st. character in line
               IF 1st. character is an F operator THEN
               BEGIN
                    READ time(t1) and command string
                    IF string is 'VIEW HIDDEN LINE' THEN
                    BEGIN
                         READ next lines time(t2)
                         total_response_time(trt) :=
                              trt + t2 - t1
                         number_of_occurrences(noc) :=
                              noc + 1
                    END
                    ELSE
                         IGNORE rest of the line
               END
               ELSE
                    IGNORE rest of the line
          UNTIL end-of-file
     END  (* loop until all monitor files have been read *)
     average_system_response_time := trt / noc
END
```

Once established, this algorithm can be used (with small alterations) to extract information about user response times, error rates, percentage use of keyboard, voice and mouse input modality and so on.

3.7. Learning to use FEMVIEW: A Student Perspective.

Under the direction of Dr. Peter Innocent, Dr. Brian Bramer and myself, eight groups of students, three students per group, were asked to learn to use and evaluate the FEMVIEW system. A time restriction was imposed by the course timetable to twelve weeks total time allowed for the assignment and six weeks computer time. Each group was asked to make an initial pilot study and then to concentrate their efforts on just one aspect of the FEMVIEW system to evaluate and report on.

The eight groups chose the following areas on which to concentrate their efforts:-

<u>Group</u>	<u>Topic of concern</u>
(A)	Visual aspects of the user interface.
(B)	Effects of hardware on naive user performance.
(C)	Causal factors approach to errors.
(D)	A reference card for increased usability.
(E)	Developing goal-oriented on-line help.
(F)	A Prolog-based command processor.
(G)	The semantics of dialogue.
(H)	Goal/functionality mismatch.

The purpose of the evaluation by students had three main objectives. Firstly, it was used as a means of generating fresh ideas on evaluation methods. Secondly, it allowed detailed analysis of areas of study not covered in the author's research. Thirdly, it provided a way of studying the use of FEMVIEW during the learning process on a wide scale and so to provide data on learning curves, user/system performance (speed and error rates) over a prolonged timescale and with a relatively large number of users.

Some limitations were encountered, unforeseen in our initial plan, that affected not only the results obtained but also users' attitudes to the system. One major problem was the effect of large numbers of users on a single-user system. The data base was set up for single user entry and its subsequent multiple use caused the system to crash on numerous occasions in the first week. Also, the implementation used by the students did not initialise the mouse correctly and therefore all options requiring mouse locations were unavailable to the students. Although both problems were quickly resolved, many users having already experienced difficulties, as a result of the bugs, had already formed unfavourable attitudes towards the software. The second limitation on the use of the software was the type of students used in the experiment. The students were all from the M.Sc. Man-Computer System course. Their knowledge of computing was extensive while their knowledge of engineering was limited. Since the software package under evaluation was primarily used by design and manufacturing engineers, this particular group of students was perhaps not the most suitable group of users. The goal of the students was to understand the software in order to evaluate it, whereas the goal of the engineer is to use the software as a tool for achieving a specific task. Since user goals are an important feature of the interface that directly influences functionality (Clarke, 1986), generalisations about the findings are to be treated with caution.

3.7.1. Overview of Student Evaluations.

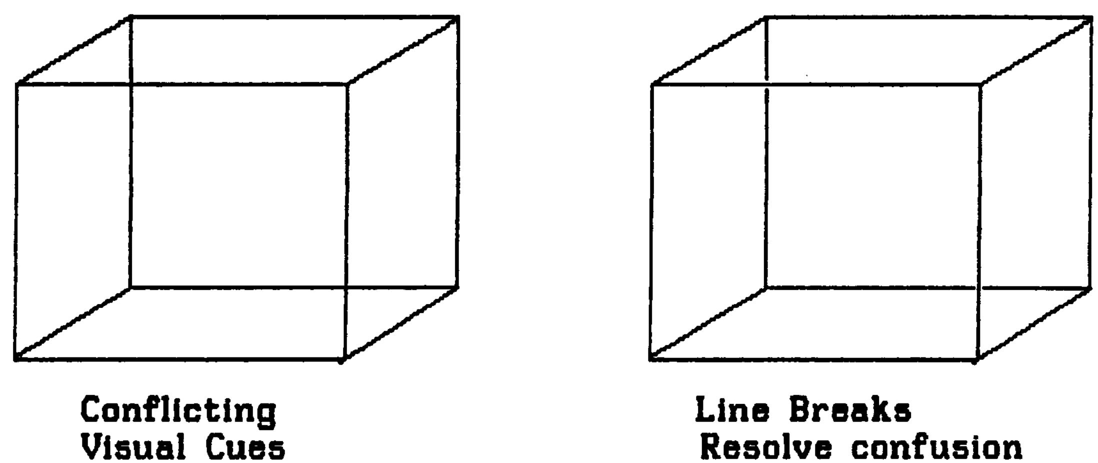
The three members of each group worked together to formalise ideas and conduct experiments on which to test hypotheses. In many cases, each member of a team took on a different role, for example, evaluator, user and observer. This enabled the students to study the system from different perspectives. Each group attended a viva at fortnightly intervals during which they discussed their progress. At the end of the twelve weeks a written report was presented by each group and a presentation given. Below is a brief summary of each of the eight groups work.

Group (A) examined the relationship between spatial awareness and skills in the use of FEMVIEW. Their hypothesis was that users with good spatial awareness will find manipulating models in the FEMVIEW system easier than users with poor spatial awareness. Although time constraints restricted the number of experiments and as a consequence no conclusive results were established, their work suggests:-

(1) Object manipulation is preferred over the eye manipulation currently used by the FEMVIEW software. Misinterpretation by conflicting visual cues can be resolved by line breaks distinguishing visual from non-visual meshes, as in Figure 3.2.

(2) Initial model presentation should reflect either its most popular orientation, a stable state or rotated to give a strong vertical axis.

Figure 3.2. Improving spatial awareness by additional visual cues.



The students also suggested an improved axis map representation, an undo feature, and tutorial sessions for new users.

Group (B) indicated that the FEMVIEW software is a 'functional' as opposed to a 'polished' interface system. Their hypothesis was that naive user performance was increased by improved hardware power. Although they deduced that this was disproved by the experiments, they went on to state that enhancements should include option highlighting to reduce mis-selection and navigational errors, improved help facilities, redesign of the dialogue to improve command connotation, and improved error handling facilities.

Group (C) examined causal factors for errors. Eliminating or controlling the cause of errors will generate increased enthusiasm and users will be more likely to accept the system. They conclude by recommending:-

- informative error messages
- improved help facilities
- dialogue alterations for improved robustness.

Group (D) produced a reference card using a command language grammar. They tested the hypothesis that the users' learning curve was improved with the addition of the reference card. Results were based on qualitative rather than quantitative feedback from users. They concluded that users preferred the reference card over the manual for quick referencing of commands although in the experiment only a small subset of the full command set was put onto the card and the card was tested on only a limited number of users.

Group (E) provided an on-line help facility (using a second computer) based on a goal-oriented approach to help as opposed to the command-oriented help provided by the manual. As with the reference card of group (D), they argued that broader knowledge of system functionality was achieved by task-oriented help facilities.

Group (F) used a conceptual model of the FEMVIEW dialogue consisting of entities and goals. These could then be translated into Prolog rules. This has the potential advantages of providing improved help facilities and/or goal-oriented default paths through the dialogue.

Group (G) examined the semantics of the dialogue to different user classes based on job descriptions and found that the perceived understanding of words in a dialogue vary depending on the subjects' background. They argued that the system dialogue was not directed towards an engineering perspective and that this forms the root cause of dialogue misinterpretation.

Group (H) argued that user's knowledge of the functionality of the software was not goal-oriented and therefore users took a bottom-up approach to solving tasks as opposed to a more favourable top-down approach.

Unfortunately, direct references of the students work can not be used since no copies of the final reports have been retained. The accounts given above are based on detailed notes made during the viva's and final presentations.

3.7.2. How the Students Performed.

During the six week period, some 43 hours of FEMVIEW use were recorded during which time over 4500 commands were processed by the software. Input to the software was restricted to keyboard and mouse entries. Keyboard entry was more heavily utilised than the mouse (76% keyboard as opposed to 24% mouse). No significant changes in use between the two input devices were observed during this period. Although some users preferred using only keyboard entry for command input, nobody used only the mouse. Users' spent an average of 18 minutes using the software each session.

7% of keystrokes were errors and 5% of commands were rejected before completion. 8% of interactions ended prematurely because of system failure. There was no evidence to suggest a correlation between error rates and experience.

The four most used commands in the software over the 6 weeks of computer use were:-

- Eye rotate	10.5%	command	usage
- Data display	8.7%	"	"
- Help	6.1%	"	"
- View mesh	4.9%	"	"

3.8. Evaluating the System: An Engineer's Perspective.

The final process of evaluation was to invite end-users to take part in laboratory experiments at Leicester Polytechnic. Five subjects took part in the evaluation. The software used for this experiment was a new release (FEMVIEW version 4.0) which had only been available on the market for a few months. (All previous experiments including those involving voice recognition had been performed using FEMVIEW 3.5.)

3.8.1. Method of Experimentation.

Subjects.

The subjects' experiences were as follows:

Subject 1 - many years experience with the software, including involvement with its design;

Subject 2 - an experienced FEMVIEW user, including two months experience with version 4.0 of the software;

Subject 3 - an experienced FEMVIEW user but with no experience of version 4.0;

Subject 4 - some use of the system, but no knowledge of the command sets or functionality of the system and no experience with version 4.0;

Subject 5 - the author, experienced with both versions of the software and also the evaluation procedures.

Equipment.

FEMVIEW is a machine independent piece of software that runs on many high-powered machines with high-resolution graphics. For continuity of results, all experiments were conducted on an Apollo Domain DN300 with high-resolution monochromatic display. The Apollo uses a 32-bit, 32 Mbit/sec. VLSI processor with a 17inch 1024 x 800 pixel display unit. The system is networked to a 70 Mbyte Winchester disk.

FEMVIEW version 4.0 has an improved dialogue over version 3.5 allowing every command to be accessed by a single key (i.e. each word in any menu is first letter unique from the others in the same menu and hence solving the problem described in section 2.3.(1) of this document). A new facility of version 4.0 is the addition of viewports which allows for several views to be presented on one screen. Other software additions did not alter the evaluation procedure.

Procedure.

Each subject was asked to perform four exercises involving, in total, some 30 tasks. Each task represented a typical operation performed using the software. (A copy of the benchmark test is given in appendix A10) The level of complexity of the tasks varied throughout the benchmark. To help the users during their interaction, the manual, paper and a pen were made available.

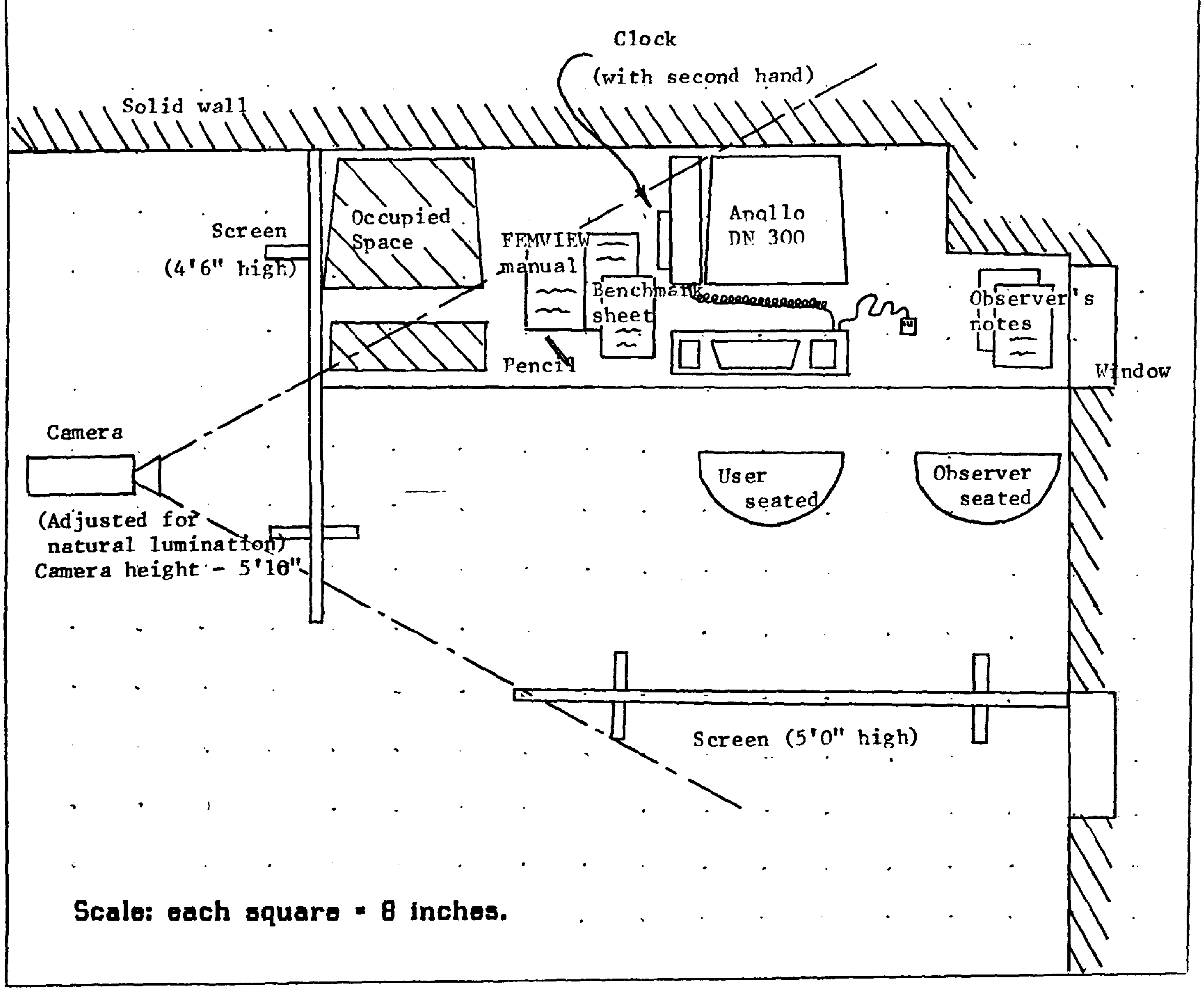
A time constraint was put on the users for purely administrative purposes. No pressure was placed on the subjects to complete all the tasks within this time period. An observer sat next to each subject in order to encourage them to talk through their actions while they were working. The observers where staff members who were experienced computer users but with no knowledge of the FEMVIEW software. They were instructed not to help the subject during the benchmark exercise except in the event of a system failure where recovery procedures would be required.

At the end of the experiment the subjects were debriefed by the observers who asked them to answer questions from a questionnaire. The questionnaire covered some 33 dimensions of the perceived quality of the software interface and the effectiveness of the evaluation techniques.

While the inclusion of the author in these experiments may be considered to have affected overall results. Every attempt was made to answer the questions honourably and without bias.

Each experiment, including debriefing, was video taped to show the physiological activities of the subjects. Figure 3.3 shows the layout of the work area. It was not necessary to film the screen, as with many other experiments of this nature since the on-line monitor recorded the screen activities. Replaying activities was possible by a few simple edits of the on-line log. Although physiological experiments have shown interesting clues to users thoughts and attitudes by 'facial leaks' which require placing the video camera directly in front of the users, it was felt that this may be extremely distracting to the users and increase the neuro-physiological effects on the experimental results. It is also a very skilled job to identify characteristics of interaction from facial expressions. For these reasons, the camera was placed at the side of the user looking over a

Figure 3.3. Workplace for subject-based formal evaluation.



4 foot 6 inch screen. In this way the user could not see the video camera without making a conscious effort to do so.

About 45 minutes of video tapes and monitor readings were collected for each of the five subjects in the study (providing a total of 3 hours and 45 minutes of video data). Analysing this information involved many hours of transcribing conversations from the protocol analysis, examining and recording activities, summarising monitor data, testing hypotheses and drawing up results.

3.8.2. Objective Performance Measures.

Examining the objective measures of performance provides a simple method of comparing controlled experimental use of the software with the uncontrolled use examined previously. Table 3.1 summarises each subject's performance.

3.8.3. Terminal Protocols.

To analyse the verbal protocols made by the users during their interaction with the computer a 'Goal Structure Model' is adopted which allows us to represent the planning behind a sequence of dialogue. This method was suggested by Morton et al (1979) and is reviewed in Hammond et al (1981). Using this method we can predict the occurrence of certain classes of errors at certain stages in the dialogue and contrast the user's internal representation of the state of the machine with the true state of the machine. The identification of system states particularly prone to error has consequences for the

Table 3.1. How users performed during the benchmark test.

Subject / Measures	1	2	3	4	5
Total number of tasks completed (max. 30 tasks)	24	24	17	4	24
Time to complete exercise 1 (secs.) (5 tasks)	233	212	491	1346	252
Time to complete exercise 2 (secs.)	1324	1524	1888	--[1]	1140
Average time per task	65	72	144	337	48
No. of commands issued (ex.1)	13	12	18	23	11
No. of commands issued (ex.2)	58	65	38	--[1]	55
Total no. of keypresses	662	896	677	331	630
Percentage backspaces made	2.1	4.8	2.1	6.0	0.7
Percentage reject lines made	1.5	11.0	0.8	13.7	3.5
Total no. of different commands issued.	23	24	20	13	22
Number of references to the manual.	None	6	16	13[2]	None

N.B. Subjects 1, 2 and 5 all completed 3 out of 4 exercises. While subject 3 completed exercises 1 & 2 and subject 4 did not complete exercise 1.

[1] Subject 4 did not start exercise 4 within the appointed time.
[2] Although subject 4 makes 13 distinct references to the manual most of his time is spent using the manual and experimenting with command options.

type of feedback that the system should present to the user, such as which states should be marked on the user's display.

Figure 3.4 gives an example of a terminal protocol with key statements highlighted and the extraction of core propositions and the underlying variables from the core propositions. For the full transcript for this subject see appendix A8.

In the displayed chronicle, the terminal protocols are displayed alongside the time stamp which shows at what time during the video analysis the dialogue took place. Events in the dialogue are shown by a '£' sign followed by a categorisation label and comment. Comments within square brackets show significant movement by the user's motor processes while comments within braces have been added by the author to add clarity to the chronicle. Statements underlined identify the core propositions from which the underlying variables can be extrapolated.

The dialogue transcribed in Figure 3.4 indicates that the user has made an error by entering the incorrect command. Although he immediately recognises an error has occurred, rather than reject the complete line, he continues to follow the path forced by the computer and re-enters the previous command. This action leads us to suspect problems in the software interface. It may be that the user has felt that by selecting SET APPEND ELEMENTS he has overwritten his previous command and therefore must re-enter the complete command line, or it may be that the user has simply forgotten that the line reject command exists. In either case this simple example shows a mismatch between the user's internal representation of how the system works and the actual functionality of the software.

Figure 3.4. Example terminal protocol.

```
The benchmark requests that the user define a temporary subset of
the current model consisting of the finite elements in the range
between 1 and 50 and to display the defined subset with hidden
lines removed.

At this point, the user has defined the subset and is now trying
to display only the defined subset on the screen. The correct
command for this is SET SHOW DEFINED.


025 User: [looks at screen]
026       So it's going to be [looks at keyboard] SET
027       [looks at screen]
030       [looks at keyboard] APPEND    £Dialogue: mistake

{enters E making the command SET APPEND ELEMENTS}
                                        £Dialogue: lost in
                                                  hierarchy
034       I'm getting lost here         £User: recognises error
036       [types in silence]
                                        £User:Redundant activity
{Now enters 1 TO 50 and therefore duplicating his last command}

040  Obs: So why did you get lost?

042 User: [looks at screen]
043       Wait a minute.
044       [looks at benchmark test]
049       I'm trying to find [reads from benchmark]...Display the
          defined subset.
053       I thought it was SET SHOW     £User: solves problem
054       [goes back to the keyboard]
062       DEFINED is it!

064       [leans back in seat and scratches head]
                                        £User: Closure
{new display is drawn on screen}

070       Right.....
```

A by-product of the limited storage capacity of short term memory is that there is great relief when information, relating to a particular task, no longer needs to be retained (Shneiderman, 1980). As a result, users have a strong drive to complete any task and gain relief. This particular user clearly shows this psychological closure by momentarily relaxing in his chair before moving to the next task.

Although in this case the error was unavoidable and was probably due to a momentary loss of concentration on the part of the user, his actions following the error clearly indicate a system fault. Errors of this kind are particularly common in systems using single keypresses for data entry, and it indicates the need for clear interpretation of reject or undo features within the system.

3.8.4. Summarising the Results from Using Protocol Analysis.

After analysing the terminal protocols the following conclusions were drawn which cover aspects of the FEMVIEW system and interface. The observations are produced solely from the analysis of the terminal protocols. The purpose of this list is to provide an indication of the type and amount of data that can be collected using protocol analysis. Several of the findings shown below are consistent with information collected by other techniques. (Transcripts for subjects 1 to 4 can be found in appendix A8). The source of the observations are shown in parentheses at the end of each item:-

(1) Both the on-line help facilities and the manual act as a reference documents rather than learning aids for inexperienced users, and both are uninformative and lacking in examples and hints for

effective use. (Subject 3 & subject 4.)

(2) The system suffers from a lack of feedback to the users, many commands may be set without any feedback to the user of successful selection. The general lack of feedback to user's is compounded by uninformative error messages making 'learning-by-example' extremely difficult. (Subject 3 & subject 4.)

(3) Confusion arises from the hierarchical structure of the dialogue where no obvious relationship is found between the utility of the commands and the keyword in the initial menu for selecting that command. For example, switching between viewports (DRAWING VIEWPORT...) and adding text to pictures (DRAWING CONTENTS TEXT...) are both selected through the DRAWING option from the initial menu and yet the two commands have no obvious relationship to each other. Another example is the selection of loadcases (RESULTS LOADCASE...) and the distortion of the model as a result of applying a loading (RESULTS NODAL DISPLACE...). The latter command is issued only as a result of the former command and yet both are selected via the same initial keyword. (Subject 3 & subject 4.)

(4) Confusion is also caused by the perceived connotation of keywords and their utility, e.g. SECTION and SET were confused during the benchmark exercise. (Subject 4.)

(5) Attributes of a model may be selected but are not saved to the data base unless the model is displayed with the attributes present (on some options only). For example a subsection of a model may be selected using the command SET APPEND ..... but this

option will not be saved unless the command SET SHOW DEFINED has been entered. Also attributes associated with a displayed model do not appear to be passed across when a model is copied from one viewport to another. (Subject 3 & subject 4.)

(6) The line reject command (/) is not clearly defined. There are no instructions how to use it or explanation of the effects from its use. i.e. what exactly is rejected - the command line, the attributes associated with a command e.g. OPTIONS DASHED (displays all subsequent models in dashed format) hence OPTIONS DASHED / could be interpreted as cancelling the options dashed command. (In practise this is achieved by entering OPTIONS DASHED OFF.) A restart (reset) command to allow user's to start again without having to logout and log back in again may also be useful. (Subject 1 & subject 3.)

(7) Although methods are available to take shortcuts through the dialogue, such as stringing of commands between semicolons, no instructions are given to indicate this to the users. Some commands can not be entered using the shortcut methods and thus when attempting to take shortcuts more time may actually be spent than if each command had be entered separately. (Subject 1 & subject 2.)

(8) Some commands are not read via the command processor (e.g. filenames) and therefore the dialogue structure is inconsistent. For example, it is not possible to reject a command using the line reject symbol (/) during the entry of some filenames and other inputs read from outside of the command processor. (Subject 5.)

(9) Interrupting the drawing of a model is only possible by rebooting the system. Therefore much time can be wasted while incorrect models, or model presentations, are being drawn. This can easily be resolved by software modifications, but may considerably increase the drawing time for correct displays since constant paging of the input buffer will be required. (Subject 2.)

(10) There exists some inconsistency between different implementations of the software, e.g. on some implementations cursor hits are selected using the space bar and on others a mouse button is used. There also appears to be some inconsistency between implementations for producing screen dumps. (Subject 1.)

3.8.5. Debriefing Users after Interaction.

The method of debriefing users after their completion of the benchmark test took the form of an attitude scale questionnaire (Monk, 1985). Using this method avoided any possibility of leading the user and ensured that an objective approach was taken to the debrief. By recording the debrief procedure, users were allowed to make qualifying statements to their answers. For example, when asked whether the manual proved useful in solving the tasks, subject 3 gave an unfavourable reply but qualifies the answer by stating his reasons (i.e. 'because the manual has no index').

The debrief questionnaire was adapted from the feasibility questionnaire of Poulson (1987(a), 1987(b)). This is a general purpose measuring scale which can be used to gather information about the perceived quality of software interfaces. In its original form the questionnaire included some 36 questions covering 24 dimensions of perceived quality. The author removed one dimension from the original list and added some 14 questions and 10 dimensions covering aspects of the manual, the experimental procedures and the users' self opinion. The format of the questionnaire took the form of a seven point scale providing levels of agreement/disagreement to direct statements. The software was described by 17 favourable statements and 17 unfavourable statements. Monk (1985) suggests that by adding the ratings from favourable statements and subtracting the ratings from unfavourable statements an overall favourability rating is given. For the FEMVIEW system this method is calculated below:

$$\text{Favourability} = \sum(\text{favourable ratings}) - \sum(\text{unfavourable ratings})$$
$$= 54.5 - 9.5$$
$$= \underline{45}$$

Alone, this value is of little significance, except to say that the debrief questionnaire revealed overall a generally positive response by users to the software. However, this figure could be useful in making comparisons between the perceived quality of different software interfaces.

Perhaps of more interest are the scores within each dimension. Not only does it provide information pertaining to defects in the software interface but it also provides clues to the quality of the statements used in the questionnaire and the usefulness of this method for collecting subjective data by quantitative measures. Table 3.2 gives an overall average of responses within the 33 dimensions covering the user interface and evaluation techniques. Positive values on the table show an overall favourable response to the particular aspect of the

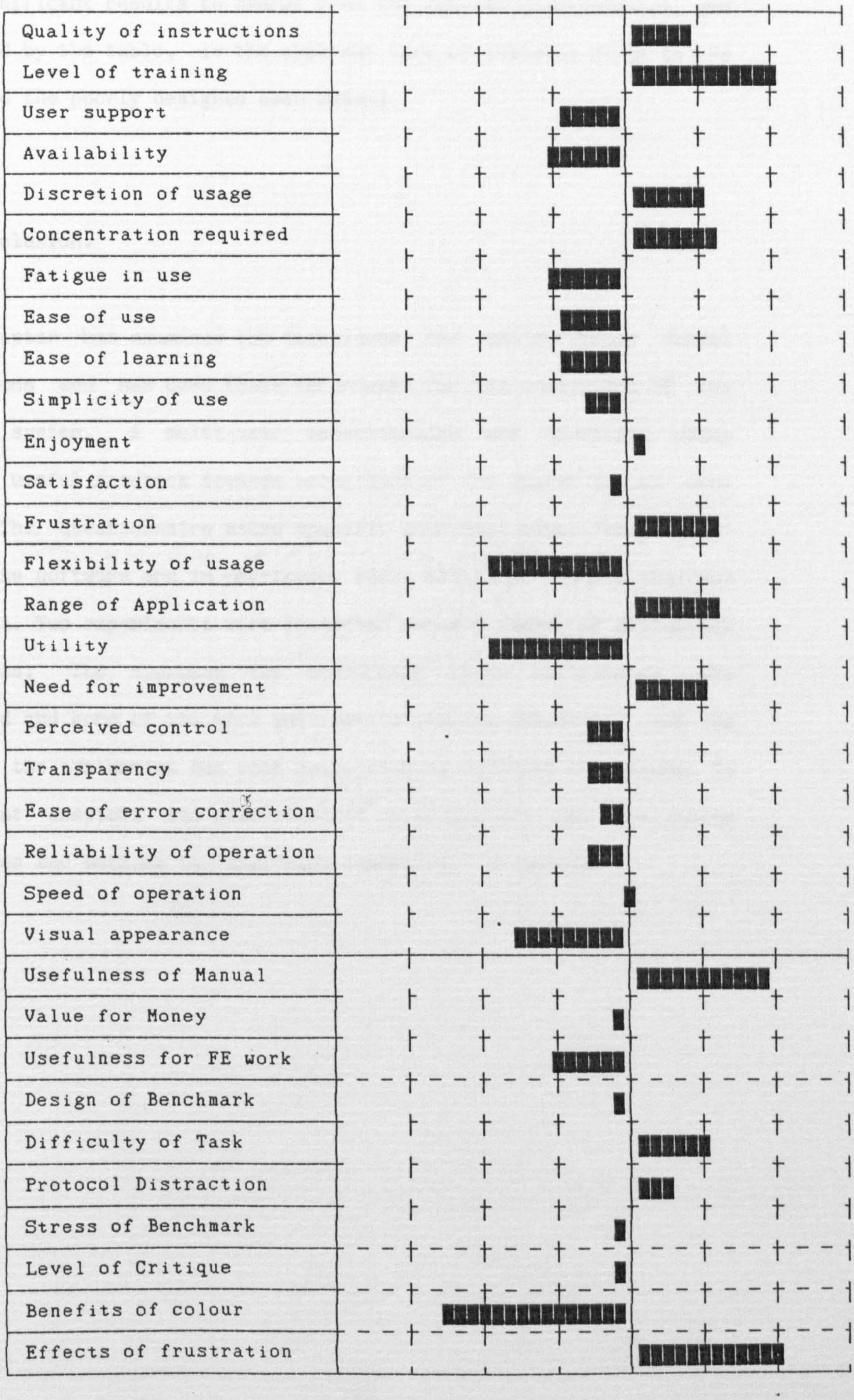


Table 3.2. Average responses within 33 dimensions covering the user interface and evaluation techniques.

interface while negative values indicate unfavourable attitudes. Appendix A8 defines the dimensions used to construct the scales. The most significant results to emerge from the debrief questionnaire, and indicated by the table, is the apparent lack of training given to the users and the poorly designed user manual.

3.9. Conclusion.

This chapter has examined the techniques for subject-based formal evaluations and has used these techniques for the evaluation of the FEMVIEW system. A multi-user questionnaire was developed which provided useful feedback towards acceptance of the system by its end-users. The questionnaire asked specific questions about individuals' use of the software and in particular about attitudes towards the user interface. Two experiments were conducted using a number of evaluation techniques. The approach for conducting these experiments was discussed and some of the more informative results presented. For the moment, the evaluation has been restricted to existing techniques. In subsequent chapters the discussion is developed and new evaluation techniques and methods for analysing 'usability' presented.

Chapter 4

User Modelling from Habitual Behaviour Patterns (HBP).

Chapter 4.

User Modelling from Habitual Behaviour Patterns (HBP).

4.1. Introduction.

In this chapter a more theoretical approach to evaluation is adopted. This includes current uses of both user and interface models in evaluation and design, and assessing the merits and pitfalls of this form of methodology. A new model based on habitual behaviour patterns or 'trait theory' is discussed and performance characteristics based on this model are examined.

Models provide a 'predictive evaluation' of proposed or partial designs (Young, 1985) and a framework for designing and interpreting experiments for evaluation. Young also points out that the term 'user models' has several different 'senses':-

- The designer's model
- The user's conceptual model
- Embedded user models or 'student models'.

Designer models of the user are based on a predictive psychological philosophy used to guide designs and evaluations by predicting human-computer system performance. The users' conceptual model is the mental model or the representation that the user has about the behaviour and control of the computer system (Gentner & Stevens, 1983, Mac an Archinnigh, 1985). Embedded user models are a representation of certain aspects of the user, held as part of the software and employed as a basis for adapting the human-computer interface appropriately for the user (Self,1974). This chapter concentrates on Designer models and their use for design and evaluation. Later, the methodologies developed are extended into embedded user models for future interface designs (section 6.3.3). Also included in this chapter are interface models. The interface model is both a user-oriented and task-oriented description of a system.

4.2. Examples of User and Interface Models.

Perhaps the most common and widely used user model is the 'Model Human Processor' (Card, Moran & Newell, 1983). It is described by a set of memories and processors together with a set of principles. The Model Human Processor can be divided into three interacting subsystems:- the perceptual system, the motor system, and the cognitive system, each with its own memories and processors. Card and colleagues produced 10 principles of operation derived from a science base which they used to describe all operations. They extended the model to take into account the cognitive information-processing activities of the user based on the Rationality Principle. This states that:-

> 'A person acts so as to attain his goals through rational action, given the structure of the task and his inputs of information and bounded by limitations on his knowledge and processing ability:-
>
> Goals + Tasks + Operators + Inputs +
> Knowledge + Process-limits --> Behaviour.'

The GOMS model of users cognitive structure consists of four components:-

- a set of Goals
- a set of Operators
- a set of Methods for achieving the goals
- a set of Selection rules for choosing among competing goals.

For error-free behaviour, a GOMS model provides a dynamic description of behaviour, measured at the level of goals, methods and operators. It is possible using the model to predict the actual sequence of operators a person will use and the time required to do any specific task. In order to construct a particular GOMS model, a task-space analysis is needed, i.e. a specification of the components of the model for the range of tasks being considered. This makes the model difficult to construct at an early design stage since task details may not have been specified.

- The keystroke-level model of Card, Moran & Newell (1980, 1983) is derived from the GOMS model, but differs primarily in its demands for prior task analysis. It again outputs a prediction of task performance time.

Given:-

- a task (possibly involving several subtasks)
- the command language of the system
- the motor skill parameters of the user
- the response time parameters of the system
- the method used for the task,

the model predicts:-

- the time an expert user will take to execute the task using the system, providing he/she uses the method without error.

The execution time of a unit task is calculated from the sum of four physical-motor operators (keying, pointing, homing and drawing), one mental operator and a system response operator. Table 4.1 shows the application of the keystroke-level model to exercise 1 from the benchmark test (section 3.8.1 - Procedure).

Constants are taken from Card, Moran & Newell (1983) and are based on an average non-secretary typist (40 words per minute). It is assumed the minimum typing facility of the command processor has been used but users do not type ahead. The response times shown are averages taken from the monitor logs.

Unlike the GOMS and Keystroke-level models of Card et al, The three-level human-computer interface model of Clarke (1986) is less ambitious in its claims. The aim of the model is to produce a structured series of questions that could form the basis for a future interface requirement. The model is a simple three-level tool that incorporates psycho-social, decision making, and human factors elements. Figure 4.1 shows a pictorial representation of the model. On the human side the interface has three layers:- psycho-social, mental, and sensori-motor while on the computer side of the interface the three layers are:- objects, machine functions, and input/outputs.

The psycho-social level suggests that we are all predominantly social beings striving to achieve goals. For acceptable interfacing at this

Table 4.1. Applying the keystroke-level model to predict error-free performance for expert users completing exercise 1 from the benchmark test.

PREDICTED						OBSERVED
Command string	Keying	Homing	Mental	Response time	Total (sec)	Total (sec)
initial setup	-	-	-	2.2	2.20	2.20
USER<	5*0.28	0.4	1.35	0.3	3.45	2.50
S BENCH<	8*0.28	0.4	1.35	0.3	4.29	3.63
S HOUSIN<	9*0.28	0.4	1.35	2.0	6.27	5.34
M B<	4*0.28	0.4	1.35	2.0	4.87	5.56
S A E 1 T 50<	13*0.28	0.4	1.35	1.0	6.39	6.94
S S D<	6*0.28	0.4	1.35	2.5	5.93	5.46
V H F<	6*0.28	0.4	1.35	7.0	10.43	10.69
E L<	4*0.28	0.4	1.35	1.5	4.37	4.13
R L CASE1<	10*0.28	0.4	1.35	0.1	4.65	6.83
R N D A<	8*0.28	0.4	1.35	0.2	4.19	4.44
P C L 10<	9*0.28	0.4	1.35	6.0	10.27	13.26
F<	2*0.28	0.4	1.35	0.0	2.31	1.59
Total Execution time for exercise 1 (in seconds)					69.62	72.57

'<' indicates the return key.

level the computer must be seen by the human to serve a purpose - it must enable achievement of specific goals, and should meet the user's higher-level needs. At the objects level the computer contains virtual representations of real-world objects. These objects specify the task and information domains in an individual-free context. The interface between these two levels supports the users in their purpose.

The mental functions level of the interface includes all the functions and processes (cognition, intuition, thinking, heuristic strategies, metaphor generation, feeling, etc.) which currently are believed to be most directly involved in human-computer interfacing. The machine functions are used by the user to reach his goals. The level two human-computer interaction aims to match the facilities and the computational speed of the computer to those of the human so that both can interact co-operatively as a decision-making team.

The sensori-motor level involves the perception, the senses, and motor processes of the human. The input/output devices of level 3 act as transducers which match certain dynamic characteristics of the computer with those of the human. The level 3 interface is the only physical interface in the model. It is the conjunction of the perceptual and physical characteristics of the human and the operational, functional, and organisational features of the computer's workstation.

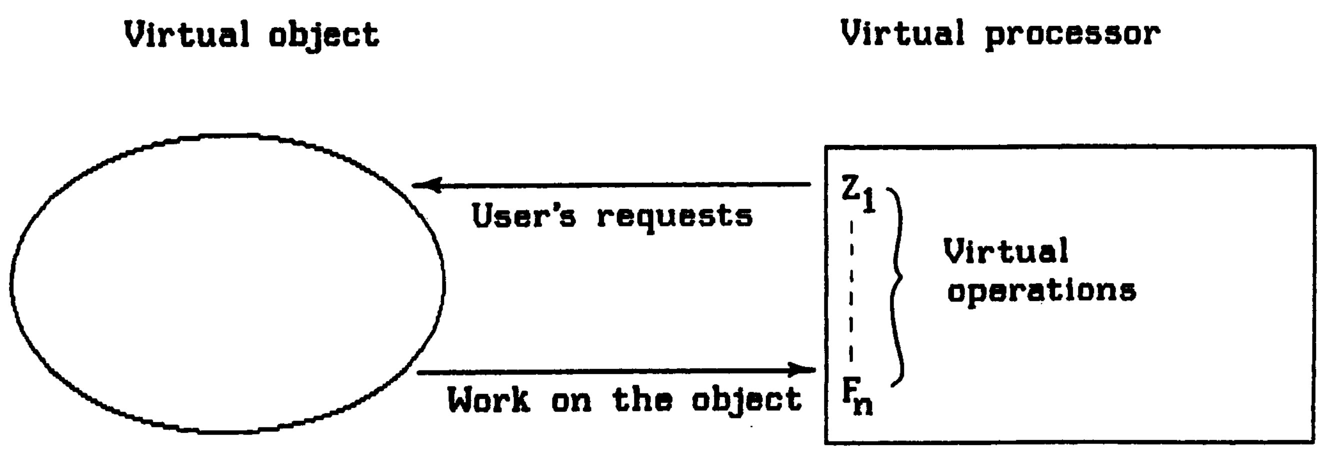
Tauber (1987) presents the concept of the UVM (User's Virtual Machine) which is a representation of the user interface of a system. It is established by cognitive theory which claims to model human mental representations of the outside world in general terms and of systems in detail. The UVM is a complex virtual object which is manipulated by a virtual processor containing a set of functions applicable to the object (see Figure 4.2). Mentally represented semantics of an object manipulating system are defined by three elements:- defining the objects, the operations on the object, and the rules for sequencing the objects and operations. Each task actually performed by the user, with the help of the system, is described by such concepts. The composition of objects are described by a structured collection of 'themes' or primitives. The interface is thus described by:-

- declaration of object types
- conceptual declaration of operations
- user's decisions on operations from the system design.

The UVM is the specification of the conceptual part of the system, the conceptual structure, and processes which affect the user's knowledge for performing his task. Tauber's specification of the user interface is intended to enable designers to define the components of the system that are controlled by the user. By pursuing this approach, Tauber suggests that the user's actual knowledge of a system could be considered by the model and the best way of extending and using this knowledge could be made known to the user.

Other user and interface models can be found described in Young (1985), Clarke (1986), Thimbleby (1985) and Tauber (1987), with a review of predictive models of the user in Card & Young (1985).

Figure 4.2. A first look at the user virtual machine.



PLACE METAPHOR

- System represents virtual objects.

TOOL METAPHOR

- System applies operations on the objects or parts of it.
- System produces/destroys objects.

From Tauber (1987).

4.3. Problems Applying Models: The Need for an Alternative Approach.

In the previous chapter, Table 3.1. shows the actual times for the five subjects to complete exercise 1. The times range between 212 and 1346 seconds. The keystroke-level model, as described in Table 4.1, shows the predicted time for an expert user to complete exercise 1 from the benchmark together with the average times obtained from observing an expert user repeating the exercise on three separate occasions. The reason for the large discrepancy between the prediction from the keystroke-level model and the actual use by the five subjects in the experiments (described in section 3.8) is that:-

(1) no allowance is made in the model for reading time,

(2) users rarely achieve error-free behaviour, and

(3) explaining actions by protocol analysis increases the time between responses.

The exercise was repeated three further times using an experienced FEMVIEW user with only the on-line log to monitor interaction. On this occasion the results were more closely matched (see the averaged observed times on the right-hand-side of Table 4.1). Table 4.2 shows that the model accurately predicted the the time for this user to complete exercise 1 of the benchmark exercise (based on the 21% root-mean-square (RMS) error suggested by Card et al).

As a word of caution, although the table shows the keystroke-level model to have a close approximation with empirical results, for error-free expert user's behaviour, the author remains sceptical of this

Table 4.2. Root-mean-square error for observed v's predicted execution time for the benchmark exercise.

Predicted Total (seconds)	Observed Total M+SE(N) (seconds)	Predicted error (E)	Square of error (E * E)
2.20	2.20+0.1(3)	0.0%	0.0
3.45	2.50+0.1(3)	27.5%	756.3
4.29	3.63+0.2(3)	15.4%	237.2
6.27	5.34+0.3(3)	14.8%	219.0
4.87	5.56+0.3(3)	-14.2%	201.6
6.39	6.94+1.0(3)	- 9.6%	92.2
5.93	5.46+0.4(3)	7.9%	62.4
10.43	10.69+0.8(3)	- 3.4%	11.6
4.37	4.13+0.5(3)	5.5%	30.3
4.65	6.83+0.4(3)	-46.9%	2199.6
4.19	4.44+0.3(3)	- 6.0%	36.0
10.27	13.26+1.3(3)	-29.1%	846.8
2.31	1.59+0.4(3)	31.2%	973.4
69.62	72.57	- 6.9	5566.3

$$\text{RMS Error} = \sqrt{(\Sigma(E * E) / n)}$$

$$= \sqrt{(5666.3 / 13)}$$

$$= \sqrt{435.9}$$

$$= \underline{\underline{20.9\ \%}}$$

model as an evaluation tool except, possibly, for comparisons between different computer systems. Other literature also shows the model to have only mixed success (Roberts & Moran, 1984), and as a piece of applied psychology it remains controversial (Allen & Scerbo, 1983).

The strong cognitive approach used in each of the four models described in section 4.2 has become the basis of current philosophies to interface design. While it remains undisputed that all behaviour stems from the cognitive ability of humans, the method by which the cognitive structure is described remains a controversial issue. The hypothesis of the author is that while a user interacts with a computer to achieve predefined goals, his activities are controlled and affected by his Habitual Behaviour Patterns (HBP) or 'Personality Traits'. The definition of personality put forward by Krech, Crutchfield & Livson (1969) and used in this work is:-

> 'the integration of all of an individual's characteristics into a unique organisation that determines, and is modified by, his attempts at adaptations to his continually changing environment'.

In computer tasks, although the computer specifies the limited task domain, the route to achieving broadly specified goals will vary from user to user. In this research programme it is suggested that personality traits (such as introversion/extroversion) can be seen to be reflected in individuals' use of interactive systems. Therefore, the study of users' personality can give software designers a useful tool for predicting the way in which users will interact with a given computer system.

4.4. Selecting a Suitable Method for the Analysis of Habitual Behaviour Patterns.

In order to test the hypothesis, it was necessary to select a suitable way of collecting data about personalities. The use of personality questionnaires in the field of computing is not a totally new idea. As early as 1971, Weinberg suggested the importance of personality in relation to programming tasks for the selection of programmers (Weinberg,1971). Shneiderman (1980) extends this idea and introduces the Myers-Briggs Type Indicator (MBTI) as a psychological test which gives insight to programmers and their interaction (Myers,1962). The MBTI measures four primary personality dimensions:-

- extroversion/introversion
- sensing/intuition
- thinking/feeling
- judging/perceptive.

Several criteria are important for the successful administration of any personality test. These are as follows:

(1) Any personality test given to computer users must only analyse nonclinical aspects of personality. Many personality tests involve clinical assessments for use by professional psychologists. These tests highlight abnormal personalities by examining responses to questions often of a highly confidential nature.

(2) Analysis of the tests should involve simple procedures free from experimenter interpretation, so removing the need for debatable expert analysis.

(3) The test should be well established providing British norms. Personalities vary between different sexes and groups of people (such as religions, countries, education, etc.). It is therefore important to provide a test suitable for the computer user population.

Using the above criteria, the MBTI test must be rejected from use in this research since not only is it a clinical test but it also only provides scores standardised to United States high school and college students. Based on the above criteria the selected test was the Sixteen Personality Factor Questionnaire (16 PF) (Cattell, Eber & Tatsuoka, 1970, Krug, 1981). The 16 PF is the most widely used test for objective personality assessment. It contains data on nearly 30,000 people with British undergraduate norms included since 1976. Scoring the 16 PF can be performed by hand or computer. Hand scoring involves overlaying answer sheets with score cards and summing the results. These scores are then normalised by reference to an appropriate table and these normalised scores are placed on a linear scale which pictorially shows subject profiles.

4.5. The Sixteen Personality Factors Questionnaire (16 PF).

The 16 PF is an objectively scorable test devised by basic research in psychology to show personality by a brief, simple process. The test is designed for use with individuals aged 16 and above. Three versions of the 16 PF are available:-

Forms A and B (Full version) - parallel forms containing 187 items each for use when a particularly detailed assessment is required.

Forms C and D (Short version) - parallel forms containing 105 items each and incorporating a motivation disorder scale.

Form E (Lower reading level form) - 128 items which are shorter, more concrete, and presented in larger type.

For the purpose of this experiment Forms C and D were chosen, with subjects only completing Form C. Since each Form took 35 to 45 minutes to complete, the selection of this Form was based on the amount of time that users could reasonably be asked to spend on this task.

The 16 PF measures 16 functionally independent and psychologically meaningful dimensions. These are summarised in Table 4.3.

4.6. Reflecting Personality in Human-Computer Interaction.

Before it is possible to reflect personality traits in human-computer interaction it must first establish how personality factors can be converted into predictive formulae. The definition suggests that personality is not fixed, but there is a certain enduring quality to personality, and therefore an individual's personality does not change when there is no reason for it to change, and fortunately changes are not that frequent (Weinberg, 1971). Since the 16 PF characterises personality by a collection of traits the aim of this experiment is to

Table 4.3. The primary source traits covered by the 16 PF test.

	Low Sten Score Description (1-3)	High Sten Score Description (8-10)
A	*Reserved.* **detached, critical, cool, impersonal** Sizothymia[a]	*Warmhearted.* **outgoing, participating, interested in people, easy-going** Affectothymia
B	*Less intelligent.* **concrete-thinking,** Lower scholastic mental capacity	*More intelligent.* **abstract-thinking, bright** Higher scholastic mental capacity
C	*Affected by feelings.* **emotionally less stable, easily upset, changeable** Lower ego strength	*Emotionally stable.* **mature, faces reality, calm, patient** Higher ego strength
E	*Humble.* **mild, accommodating, easily led, conforming** Submissiveness	*Assertive.* **aggressive, authoritative, competitive, stubborn** Dominance
F	*Sober.* **prudent, serious, taciturn** Desurgency	*Happy-go-lucky.* **impulsively lively, enthusiastic, heedless** Surgency
G	*Expedient.* **disregards rules, feels few obligations** Weaker superego strength	*Conscientious.* **persevering, proper, moralistic, rule-bound** Stronger superego strength
H	*Shy.* **restrained, threat-sensitive, timid** Threctia	*Venturesome.* **socially bold, uninhibited, spontaneous** Parmia
I	*Tough-minded.* **self-reliant, realistic, no-nonsense** Harria	*Tender-minded.* **intuitive, unrealistic, sensitive** Premsia
L	*Trusting.* **adaptable, free of jealousy, easy to get on with** Alaxia	*Suspicious.* **self-opinionated, hard to fool, skeptical, questioning** Protension
M	*Practical.* **careful, conventional, regulated by external realities** Praxernia	*Imaginative.* **careless of practical matters, unconventional, absent-minded** Autia
N	*Forthright.* **natural, genuine, unpretentious** Artlessness	*Shrewd.* **calculating, socially alert, insightful** Shrewdness
O	*Unperturbed.* **self-assured, confident, secure, self-satisfied** Untroubled adequacy	*Apprehensive.* **self-reproaching, worrying, troubled** Guilt proneness
Q1	*Conservative.* **respecting established ideas, tolerant of traditional difficulties.** Conservatism of temperament	*Experimenting.* **liberal, analytical, likes innovation** Radicalism
Q2	*Group oriented.* **a "joiner" and sound follower** Group adherence	*Self-sufficient.* **prefers own decisions, resourceful** Self-sufficiency
Q3	*Undisciplined self-conflict.* **careless of protocol, follows own urges** Low integration	*Controlled.* **socially precise, following self-image, compulsive** High self-concept control
Q4	*Relaxed.* **tranquil, torpid, unfrustrated** Low ergic tension	*Tense.* **frustrated, driven, restless, overwrought** High ergic tension

Reproduced by kind permission of NFER-Nelson Pub.

suggest how users will perform during an interactive session given their position between a pair of traits. For example, one pair of traits often used is 'introversion/extroversion'. The MBTI suggests that introverts:-

- like quiet for concentration
- tend to be careful with details
- tend not to mind working on one project for a long time
- are interested in the idea behind their job
- dislike interruptions
- Like to think a lot before they act
- work contentedly alone
- have some problems communicating,

while extroverts:-

- like variety and action
- tend to work fast, dislike complicated procedures
- are often impatient with long slow jobs
- are interested in the results of their job, in getting it done and how other people do it
- often don't mind interruptions
- often act quickly, sometimes without thinking
- like to have people around
- usually communicate well

(adapted from Shneiderman, 1980).

Providing one agrees that these descriptions are accurate, users with personalities at opposite ends of the introversion/extroversion scale might logically be expected to interact with the computer in different manners. If experimental evidence supports this theory, the challenge for the designers of new systems is to build a conceptual representation of habitual behaviour patterns into the software thus providing adaptive interfaces that can be modified according to the individual personality of the user.

In the example above a very simplistic approach is shown using just one personality trait. In practise, the individual's personality is determined by complex interrelationships between different traits giving each of us our own unique personality. Traits such as emotional stability or instability will affect how well a person will tolerate certain conditions or situations while others describe motivation and discipline for tackling a job.

4.6.1. Interpreting the 16 PF Profile Patterns.

It is not the intention of this section to describe in detail the source traits of the 16 PF questionnaire. Detailed coverage can be found in Cattell, Eber & Tatsuoka, (1970) and Krug, (1981). However, for an overview of the 16 primary traits and the 4 secondary traits see Appendix A11.

In both Table 4.3 and the descriptions given in Appendix A11 the labelling of traits is consistent with other text of the 16 PF questionnaire. Each relates to the universal index of personality factors. Alphabetically labelled traits have been found, reciprocally, in studies of personality, while Q1, Q2, Q3 and Q4 are found only in questionnaire responses.

The second order traits described by Qi to Qiv cover a broad spectrum of the primary source traits. Krug has produced a summary of 81 profile patterns based on second order trait scores and covering a wide variety of personality types.

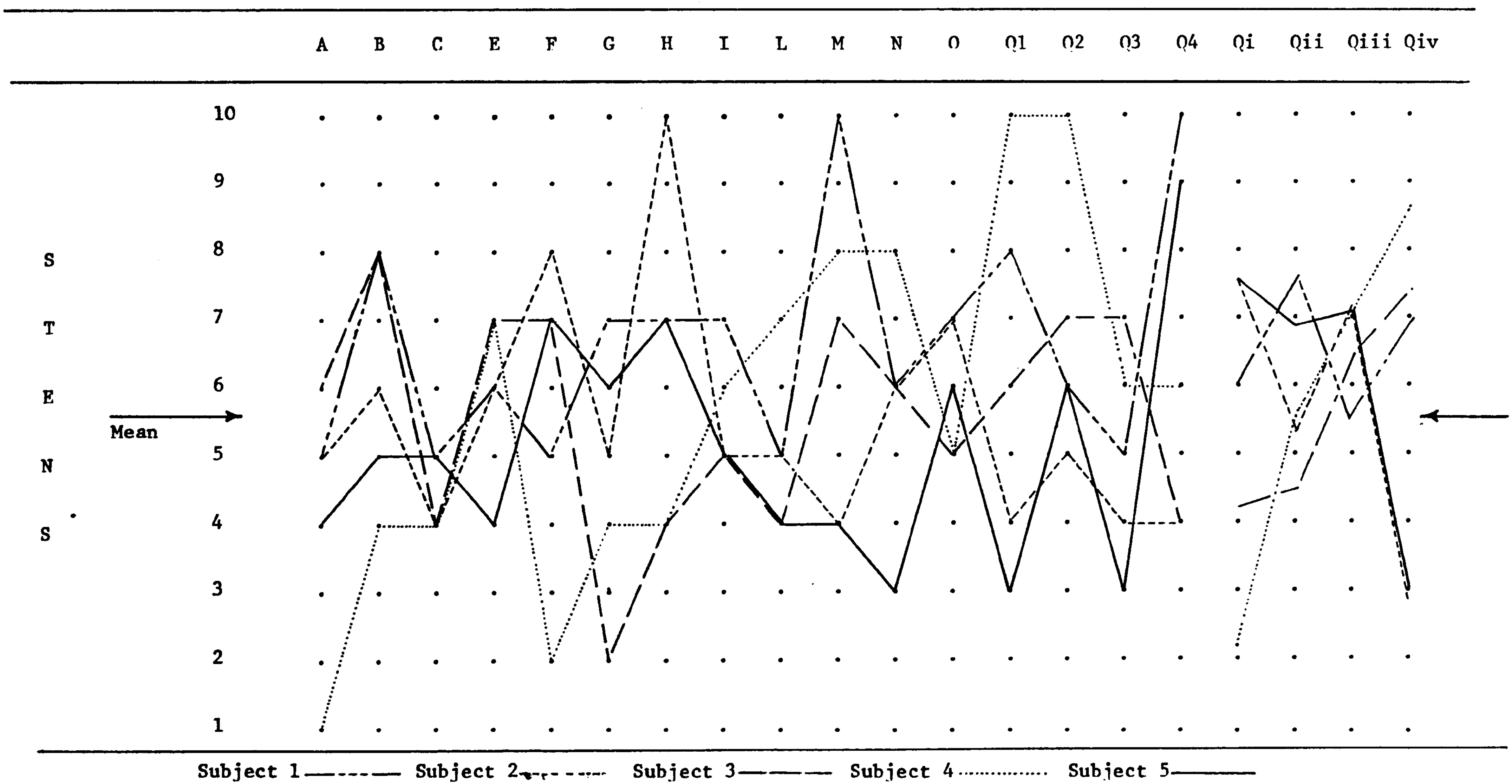
4.6.2. Individual Personalities Described by the 16 PF.

Figure 4.3 shows the personality profiles for the five subjects who took part in these experiments. Sten scores of 1-3 are low scores and correspond to descriptions on the left hand side of Table 4.3, while high sten scores (8-10) correspond to descriptions on the right-hand-side. Sten scores in the central region (4-7) show the average profile. We would expect in a typical profile the majority of traits to show scores in this central region.

Of the five subjects used in this pilot study, the profiles show considerable variation. The author has collected the characteristics into a descriptive form purely for ease of interpretation. All clinical interpretations and some obvious but unrelated characteristics in terms of affecting human-computer interaction, have been omitted from these summaries.

The descriptions are based wholly on the profile patterns produced by the questionnaire. It should be stressed, however, that the descriptions below are not necessarily supported by the individuals themselves or the author. Arguments against the validity of

Figure 4.3. A summary of personality profiles generated from the 16 PF test.



personality assessments are well documented in other literature (Ellis, 1946, Vernon, 1966).

Subject 1.

Subject 1 shows above average intelligence. He is imaginative, absorbed in ideas and fanciful. He tends to be interested in the theory behind things and is generally enthused with his activities although having an occasional tendency to give up. He is highly tense, anxious and frustrated (high ergic tension) and is easily irritated. The second order traits show a common profile pattern, showing only a slightly higher than average score on anxiety. This profile is common in some 8% of the male population.

Subject 2.

Subject 2 shows the common characteristics of an extrovert. He is socially bold and energetic, has little inhibition, is talkative and carefree. He is alert and a quick decision maker, although impulsive. He handles problems at a 'dry' cognitive, objective level. His profile suggests he has a practical, non-critical approach to life without a strong theoretical orientation.

Subject 3.

Subject 3 again shows above average intelligence. His profile shows him to be radical, a law unto himself concerned only with his own interests (expedient). His low conformity score is also associated with unreliable, slack, quitting, self-indulgent personality characteristics with little determination to do well. His second order traits show him to be self-sufficient and disciplined with above average creative potential.

Subject 4.

Subject 4 has perhaps the most significant profile pattern in the group. He shows the typical behaviour of an introvert, reserved, cautious, likes to work alone and is uncompromising. He likes the logical hardheaded intellectual approach yet is imaginative, absorbed in ideas and fanciful. He prefers silent introspection and dislikes communicating. He is generally enthused but on occasion gives up. He has a lot of initiative, likes to cut corners and has an analytical mind. He is a well informed individual, hard to please and generally critical. His profile shows an overall creative-analytic person who prefers situations in which he is left alone to think through problems and arrive at innovative solutions.

Subject 5.

Subject 5 is forthright and straightforward. Less constrained by rules and standards he has simple tastes, is content with what comes but lacks self-insight. He is less well informed than others with a tendency to follow his own urges. He is tense and anxious and easily irritated. He has a quick reaction time and handles problems objectively. This individual's profile shows an elevated score on tough poise. This might imply that this person finds difficulty in relating to others, by being insensitive to the emotional impact his actions might have on others.

4.6.3. Recognising Personality Related Characteristics in Performance.

The personalities described above represent the five subjects' used in the laboratory evaluation of engineers (Section 3.8). In this section key observations from the performances of these subjects' are examined in relation to the individuals' personalities as described by the 16 PF questionnaire.

Subject 1 found the benchmark exercise reasonably straightforward showing few difficulties with his interaction. It is noticeable that the range of commands this subject used differed from the others. While he was precise with the use of commands, he showed initiative and imagination in selecting appropriate pictorial representations. He used his knowledge of the system combined with his interpretation of the objectives of the benchmark exercise, not only to achieve his goals, but also to call into question the suitability of some operations requested by the benchmark. The subject indicated, both during the interactive session and later in informal discussions that the inconsistent part of the benchmark exercises had caused considerable frustration. He appears to have been generally enthused by the benchmark and shows particular interest in the theory behind conducting this evaluation, which is again reflected by his personality. The (subjective) impression of the author is that subject 1 shows a reasonable correlation between his personality and his activities during, and after, the interactive session.

Subject 2 certainly showed no signs of inhibition during the video recording of the benchmark exercise. His responses during the debrief indicate only slight agreement that talking through ideas is off-putting and total disagreement that the simulation was stressful.

These observations agree with the extrovert personality revealed by the 16 PF questionnaire. He showed a considerably high error-rate which can be related to his impulsive nature. This is also reflected at the start of his interaction where he has trouble accessing the FEMVIEW database containing the model descriptions because he neglects to read the benchmark exercise in enough detail (see appendix A8). However, once successfully entered into the system his reasoning behind his actions is clear and objective again corresponding to his personality. The large number of keystrokes made this subject shows that he prefers the 'learning-by-doing' approach (Carroll & Mack, 1984) to the more intellectual approach, again adding support to the extrovert personality described by the 16 PF (this is also supported by his protocol analysis).

Subject 3 makes a very good attempt at the benchmark test considering his lack of knowledge of version 4.0 of the software. In his interaction he shows that he is not meticulous about his interaction, easily disregarding operations with no easy solutions. At one point during his interaction he feels he can not complete exercise 2 and moves onto exercise 3. It is only after discovering that the same problem exists with exercise 3 that he returns to exercise 2 and resolves the problem (in this case manipulating viewports). This is consistent with the subject's quitting nature. However, having resolved this problem, his enthusiasm and motivation are (visibly) seen to increase making him more determined to resolve other problems he might otherwise have ignored (such as scaling factors and the inclusion of text). Once motivated he is persistent and works hard to complete exercise 2 to the end. During his interaction he quite freely admits his own shortcomings in his knowledge of the system but is generally prepared to have a go at solving the problems. Overall, his interaction suggests evidence of a high mental capacity but it would appear he requires frequent feedback for psychological closure to maintain motivation. This is not reflected well by this individual's measured personality characteristics.

Subject 4 performs very badly during the exercise. This can be attributed to a number of reasons:-

(1) the subject had not previously used the software for its intended purpose and therefore had difficulty interpreting the meaning of the benchmark exercise;

(2) the strong introvert characteristics and high independence score suggest that this individual would have performed better if left alone to solve the problems in his own way;

(3) the manual and on-line help facilities provided with the system give little informative help to the naive user.

His personality is reflected in that he is particularly critical of the software in respect to the help facilities and error messages. He also shows an experimenting approach by attempting different commands and trying to use the information gained to learn the system (learning-by-doing). In this case there appears to be some evidence to suggest that his performance was predictable by his personality but the results are clearly influenced by the laboratory conditions and the subjects lack of knowledge concerning the system.

Subject 5 has both in-depth knowledge of the benchmark, the rationale behind its generation and a the system. His interaction was quick and precise although perhaps rather uninventive. Unlike subjects 1 and 2, no attempt to cut corners was made by combining commands, and very little planning strategy adopted. The tough poise second order personality trait suggests a quick reaction time and an objective approach to problem solving while the low score in the conservative/experimenting trait (Q1) suggest greater respect for the established way of doing things. Support for both these characteristics can be seen from this individuals interaction. There is little evidence of tension or anxiety.

Since knowledge of the test results could have biased the author's subjective interpretation of subjects personalities, this experiment suggests at most, that personality may affect user-computer performance. Unfortunately, these experiments are sadly deficient in data to prove or disprove the hypothesis presented. Experimentation using experienced, rather than naive, users is an expensive, time consuming, most difficult task. What this experiment has shown, is that the relationship between the psychology of individual's personality and human-computer interaction requires considerably more research.

4.7. Refining Methods for Identifying Useful Traits and Applying the HBP Model to System Design.

Having looked at personality in relation to human-computer interaction, it is noticeable that many of the 16 source traits appear to have little or no relevance to computer interaction and are therefore superfluous to the study, while others would appear to have a significant affects on interaction. If the important and unimportant traits can be separated, then it would be possible to develop a new questionnaire which could be used specifically to provide useful information about the user population to system designers. This topic is not pursued in this thesis since it would be dangerous to speculate based on only a limited supply of data.

4.8. Conclusion.

In this chapter the hypothesis that people's personality is reflected in their interaction with the computer is examined. The basis of this research has developed out of the need to model the cognitive structure of individuals for future interface design. The pilot study gives some support to the hypothesis in the sense that there has been no clear refutation although the author stresses that further research is required in order to refine the strengths of traits against other influences in determining human-computer interaction. For example, while a person's stability may be a key factor in determining the amount of time he is prepared to wait for a response to an operation, this will also be influenced by the user's knowledge of the operation and system, irrespective of his personality.

From the designer's point of view, the HBP model of users, like the three-level interface model of Clarke (1986), can provide a structured framework for asking questions about the interface design. Questions such as:- 'How can the system satisfy the differing requirements of people at opposite ends of personality traits?' must be answered if future systems are to be 'usable' to all users.

Chapter 5

Towards a Practical Framework for Assessing the Usability of Human–Computer Interfaces.

Chapter 5.

Towards a Practical Framework for Assessing the Usability of Human-Computer Interfaces.

5.1. Introduction.

In this chapter the observations and methodologies presented in Chapters 3 and 4 are combined to develop a suite of evaluation techniques, including both objective and subjective measures (Roberts, Kirby and Candy, 1983), to provide a framework for assessing 'usability'. The primary aim of the designer is to provide usable systems. Barnard et al (1981) suggest that to be truly 'usable', a system must be compatible with users' cognitive skills in communication, understanding, memory and problem solving as well as human perception and action.

5.2. Principles of Design and Evaluation.

The concern of evaluation is the appraisal and amendment of systems. In this respect evaluators are also designers and both should follow the same principles. For the design of usable systems Gould & Lewis (1985) suggest four primary principles, which hold true for evaluation:-

(1) Understand the users. (This includes their cognitive, behavioural, anthropometric and attitudinal characteristics.)

(2) Work closely with the user population.

(3) Simulate real world tasks.

(4) Design, test, measure, redesign and repeat as often as necessary.

5.3. Questionnaires as a Method for Data Capture.

Questionnaires are given separate coverage from other tools for evaluation described in the next section since as a technique they can, and have been used, as the predominant method for evaluating systems.

From the multi-user questionnaire (Section 3.5.4), the sign test appears to provide appropriate data on which to base assumptions about FEMVIEW's use in industry, but details relating to an individual's attitudes towards the interface features, such as menu prompts and help facilities, require further validation. For this reason, the answers from the multi-user survey questionnaire, provided by our five subjects, were compared with their answers given during the debriefing session following the benchmark test. Specifically, two questions were considered:-

(1) 'How informative do you find the manual?'

(2) 'How useful are the error messages?'

Table 5.1. indicates the results given by the five subjects who took part in these experiments.

Table 5.1. Mismatch between questionnaire ratings and debrief answers.

	MANUAL		ERROR MESSAGES	
	questionnaire	debriefing	questionnaire	debriefing
Subject 1	Adequate	poor	Adequate	Adequate
Subject 2	good	Adequate	poor	good
Subject 3	good	very poor	Adequate	poor
Subject 4	Adequate	poor	Adequate	poor
Subject 5	poor	very poor	very poor	good

(The debriefing information has been converted from a 7 point favourability scale to a descriptive form by the author)

In the initial questionnaire, subject 3 considered the user manual to be good, but after completing the benchmark test (in which he made some 16 references to the manual), his opinion of the manual had changed to very poor. Both subject 5 and subject 2 considered the error messages to be poor in the questionnaire, but when asked about the error messages during the debrief, they both gave favourable answers.

These observations suggest that answers to questionnaires are extremely subjective and user's attitudes do not necessarily reflect their actual performance using the software. This agrees with the work of Ramsey and Atwood (1979) who drew similar conclusions.

Benchmarks and debriefs are preferred over questionnaires and introspection since data provided during the debrief can be verified by examination of users' performance during the benchmark tests. As a singular evaluation technique, evidence suggests that data collected by questionnaires is not necessarily consistent with actual practices during interaction. Yet, opinions are, in many cases, very important to product and company success and so in this respect questionnaires can be seen as being useful and informative.

It may be interesting to repeat user questionnaires at various intervals during a package's life-cycle to monitor how opinions change. Simply examining whether responses are generally favourable or unfavourable will give some indication of users' attitudes towards the software.

Secondly, careful consideration should be given to the users motivation for answering the questionnaire. Data obtained from the survey described in section 3.5 indicates that the use of the FEMVIEW software has increased since its installation in the majority of companies. This observation is then used to argue that the software can be considered successful in fulfilling the needs of the user population. However, this assumes a representative sample of users have responded to the questionnaire. To be satisfied that this is the case, the motives for completing the questionnaire must be more closely examined. The most important question to ask is:-

> 'Do users who are unhappy with, or no longer use, the software have the same incentive to reply to the questionnaire as the user who is generally content with the package?'

If the answer to this question is no, then the results are biased and no conclusions can be drawn. In the survey conducted for this research it is most likely that data is biased in this way. This is illustrated by the fact that opinions expressed by the students did not correspond to the results obtained from the questionnaire, although this, in part, may be influenced by other considerations, such as an unreliable prototype version of the software. Also, the perspective of the students was the examination and evaluation of the software interface while the engineers viewed the system as a tool for accomplishing specific tasks.

5.4. Other Tools for Evaluation.

5.4.1. User Modelling.

Both designers and evaluators would find it highly desirable to be able to analyse an interface while it is still in the conceptual stage of design. Towards this end, user and interface models have developed. Models form the basis from which system performance can be predicted before development of even a prototype system. Although user models have shown some success based on the error-free performance, the development of these techniques is still very much in its infancy. Interface models have been used to form the bases for questioning future interface requirements and have been used in the construction of embedded user models within the interface. Again, much further research is required before future systems can benefit from these techniques. Until simple methods for constructing models have been developed and clear benefits, deriving from the models, established, user and interface models will remain a topic of little practical use to evaluators.

5.4.2. Prototyping.

In greater use in current design and evaluation procedures are techniques for prototyping (Shackel, 1981). Several, so called fourth generation programming languages (e.g. FOCUS, RAMIS, NOMAD, etc.) have been produced to help develop prototype systems quickly and effectively (Connor,1985). Prototyping allows the design team (and often customers) to use the interface to get hands-on experience, this in turn, permits obvious design flaws to be corrected (Savage and Habinek, 1984). It can take many forms and the advantages of this technique are well documented (Sommerville, 1985).

5.4.3. Evaluation Benchmarking.

Having created a prototype system for evaluation, formal methods for evaluating the system include simulation (Eason, 1983) which for software products can be achieved by evaluation benchmarking (see section 3.4 - Benchmarking). Tasks in the evaluation benchmark should be based on the knowledge and skill required by the user to effectively use the software. Benchmarks can be used for many different facets of evaluation. By specifying tasks in broad general terms benchmarks can be used to compare different packages. In the experiments conducted for this research, the aim of the benchmark was to evaluate how well the mental functions of the user are represented by the commands of the machine, thus satisfying the goal of usability. Benchmarks can also be used to confirm predictive results from user models. The disadvantage of the evaluation benchmark is that it generates another artificial dimension to the evaluation such that before the user can effectively use the system he/she must first understand the intention of the benchmark.

5.4.4. Protocol Analysis.

Protocol analysis or terminal protocols have been used with much success in these experiments. The method produces qualitative data that can provide a broad survey of phenomena and problems within the task domain. This contrasts to other controlled experimental studies, which typically examine the influences on behaviour of only a few factors in terms of some quantitative measure of performance (Mack, Lewis & Carroll, 1983).

Although the process appears to have been used with considerable success in a number of evaluations (Newell & Simon, 1972, Carroll & Mack, 1984), it still remains a technique little used. Certainly the interpretation of even short conversations involves many hours of analysis. What appears to be lacking, at present, are clear techniques for analysing the transcripts. Hammond et al (1981) used a method of analysis whereby views expressed by participants were classified with respect to a general set of variables underlying them. This method became the broad basis for the approach developed by the author. This procedure was as follows:

(1) transcription of the recordings from tape;

(2) insertion of key motor processes (from video analysis), timing details and information required to interpret the sequence of events;

(3) the segmentation of the manuscript into statements representing points made in the conversation;

(4) highlighting of the critical section or 'gist' of the statement;

(5) extraction of the nub or 'core proposition' of the statement;

(6) insertion of event categorisations (prefixed by £) and explanatory comments;

(7) determination of the underlying variables from the core propositions.

Figure 3.4 shows a typical transcript after this process has been applied. In section 3.8.4 a review of the conclusions drawn from the analysis of terminal protocols is provided suggesting how the FEMVIEW system may be improved.

When listening to stories on the radio our minds conjure up images which help us to picture the scenes. This is achieved by the careful description of the environment on the part of the reader. However, when watching television the scene is before us and so no explanation is required. Take away the picture from the screen and our image of what is happening becomes distorted. The same is true in evaluation when protocol analysis is performed by sound recordings alone. While the on-line monitor can describe the inputs from the user, without video feedback it is very difficult to observe how the manual is used, the reasons for long delays, or the meaning behind why a statement has been made. The analyser is left to fill in the missing gaps with the end result being influenced by the interpretation and views of both subject and evaluator. Therefore, it can be argued that for a complete picture of the interaction process, video recording should always be included with protocol analysis. Also, different personalities express themselves in different ways. For example, subject 3 indicated anxiety and relief when problem solving (by distinct body language) which were only noticeable during the video playback.

The advantages and disadvantages of video and talk-through techniques are suitably summarised in Lund (1985). Disadvantages include:-

(1) the analysis is time consuming;

(2) no meaningful timing data can be collected;

(3) the situation is stressful to the participants (although Lund suggests that users soon relax).

The advantages are:-

(1) the observer is placed with the user so the cause of errors are easily identified;

(2) early problems that are often later forgotten are retained;

(3) a lot of data is provided by just a few subjects;

(4) detailed analysis can be performed later without interrupting the users (e.g. by stopping the tape).

5.4.5. Physiological Measures.

It would have been desirable to have included physiological measures during the evaluation but for a lack of appropriate equipment. While it is recognised that some useful information has been obtained by these methods, both, the conditions of experimentation, and the analysis of results will require specialist knowledge. Work by Thomas

(1987) found no evidence to support any relation between physiological effects and computer use. In the experiments conducted for this research project no physiological measures were taken other than a single question being asked during the debriefing of the subjects, the question being whether they found the benchmark test stressful. Of the five subjects who took part in the evaluation, only subject 3 reported having been stressed during the benchmark exercise.

5.4.6. Debriefs.

Debriefing of subjects after the benchmark exercise proved useful in obtaining supportive evidence of users' attitudes. The debrief questionnaire successfully highlighted some of the major problems with the software, in particular the lack of training given to users and the poor referencing in the manual. In some respects using a questionnaire tended to make the debrief more formal. Users concentrated on the subjects covered by the questionnaire when perhaps in a more informal situation other, more pointed, criticisms might have been raised.

In the favourability questionnaire (Monk, 1985) used during the debrief, users were asked to give a rating between total disagreement (-3) and total agreement (+3) to statements such as:-

> "I feel satisfied with FEMVIEW."
> "FEMVIEW is easy to use."
> "I do not feel in control of FEMVIEW."

In many cases, the users found that these questions were difficult to answer (mainly because they were too general). Although the designer of the debrief questionnaire (Poulson, 1987(a), 1987(b)) specifies that the questions only form the outline for further discussion. In the experience of the author, this rarely occurred.

5.4.7. On-line Logging.

On-line logging has been described in some detail in section 3.6. Since this technique requires little programming effort and provides an unobtrusive means of observing interaction, it is an extremely useful evaluation tool. One simple piece of information on-line logs provide is which commands are and are not being used. This information can be useful when considering shortcuts for selecting frequently used commands. Also, concern should be given to commands supplied with the software that are not used since an unknown command is of no practical use in an interface (Root & Draper, 1983).

5.5. Considering User Experience: Expert or Novice.

Expert users, by definition, form strong links between the functions of a system and their mental representations of the task. In this way they form solution-oriented goals - they look at tasks in terms of what they can achieve using a particular piece of software. For example, given the task of producing a top quality curriculum vitae, the expert in traditional word processors (such as Wordstar or Spellbinder) may place little priority on what character font is to be used. In contrast, the expert of say, an Apple Mackintosh word processor (e.g. MacWrite) is likely to place a higher priority on the character font since it is an integral part of the system's functionality.

In evaluation, the distinction between the naive and expert user is very important. This is particularly true when evaluating existing systems. The fundamental difference between the two groups are the requirements of each. The naive user is interested in using the system to achieve a specific goal. To accomplish this he must be able to learn and understand the commands of the software. The expert user already knows to what extent the software can help him to achieve his goals and he understands the functions sufficiently to do this. What the expert wants from an updated system are possibly, shortcuts through the dialogue, amendments to increase functionality and increased flexibility in usage.

In this research project, both the needs of the naive user by the unobtrusive monitoring of system use by M.Sc. students (section 3.7) and also the needs of the end-users by laboratory experimentation (section 3.8) have been addressed. For usable systems the user must be supported from his first experiences using a new system right through to becoming an expert, regular user. In much of the software used on a day-to-day basis, there exists facilities of considerable benefit that are never utilised because the user simply doesn't know they exist. An example can be found in the Propascal compiler. The initial installation (as provided to students at Leicester polytechnic) has four compiler options set by the installation procedure. These options are maintaining source line numbers in the object code and performing range checks on index bounds, assignments and pointers. While these are helpful to the inexperienced user, the expert user can manage adequately without them. Once removed, the compiler will take less than half the time to compile. Reducing compilation time is very important for most people who regularly write programs and yet, without being told, it is unlikely that users will ever reinstall the software without these default options. This is by no means an isolated example. Users having learnt a method for achieving a specific task, rarely look for alternative approaches. Therefore when alternative methods are available it is necessary for the software to supply the user with this information within the dialogue. One method by which this may be achieved is by the software adopting the role of coach. (This is described in more detail in section 6.3.3.2.) In this way the computer initiates information and encourages users to take a more 'active' role in learning to use the system.

5.6. Considering Interface Characteristics.

5.6.1. Input/Output Devices.

In Chapter 2 voice recognition was proposed for the entry of commands within the FEMVIEW CAD package. A prototype system was developed and an informal evaluation conducted. While the system showed that the application could be used quite efficiently by spoken commands, there were no particular advantages for its use. It merely provided the user with an additional medium for communicating with the system. It appeared through numerous demonstrations that some individuals would welcome this additional medium while others would not. In this research project the system was never intended to become a marketable product since voice technologies are still relatively expensive. In future systems, it is possible that voice recognition technologies will become an integral part of computer hardware (as is now the the case with the mouse). The same may also be true for other input devices such as tablets and pens, roller balls and joysticks.

One can clearly see how the whole ethos of interfaces has changed as a result of widespread use of mice. Would icon based technologies ever have become popular with only keyboard operation? Different users prefer different input devices and choice is not always consistent with the most efficient means of interaction. In the initial survey of FEMVIEW users some 24% of respondents to the questionnaire stated that they would like to be able to operate the software using the mouse. Where possible, a choice of input devices should be available. Re-evaluation of an interface may be necessary as a result of the availability of new input devices. While mouse entry might suggest one type of interface, other devices suggest alternative designs. For example, the MEDUCA XP system developed at Cambridge Interactive Systems Ltd. uses a tablet and pen with six overlay sheets. All inputs (including ASCII text) can be entered by pointing to the appropriate place on one of the overlay sheets (a pictorial representation of the 'qwerty' keyboard is used for the entry of text).

As graphic systems have developed, so too has the need for better printers with facilities for producing graphical pictures and high quality print fonts. However, hardware concerns are outside the scope of this research.

5.6.2. Dialogue Design.

There has been much written about dialogue design and it is not the intention of the author to reproduce this work. However, appropriate dialogue design is critical to usability. Requirements of dialogue necessary for more acceptable systems are proposed by Hayes, Ball & Reddy (1981) and are summarised in Kidd (1982). Capabilities include:-

(1) Flexible parsing that allows for small mistakes in syntax and spelling.

(2) Robust communication, keeping the user informed of assumptions made and allowing him to make changes where necessary.

(3) Identification from description - the computer's ability to identify objects known to it internally by the user's description of them.

(4) Focus tracking, allowing search and excursions through the dialogue.

(5) Natural outputs appropriate to the current context.

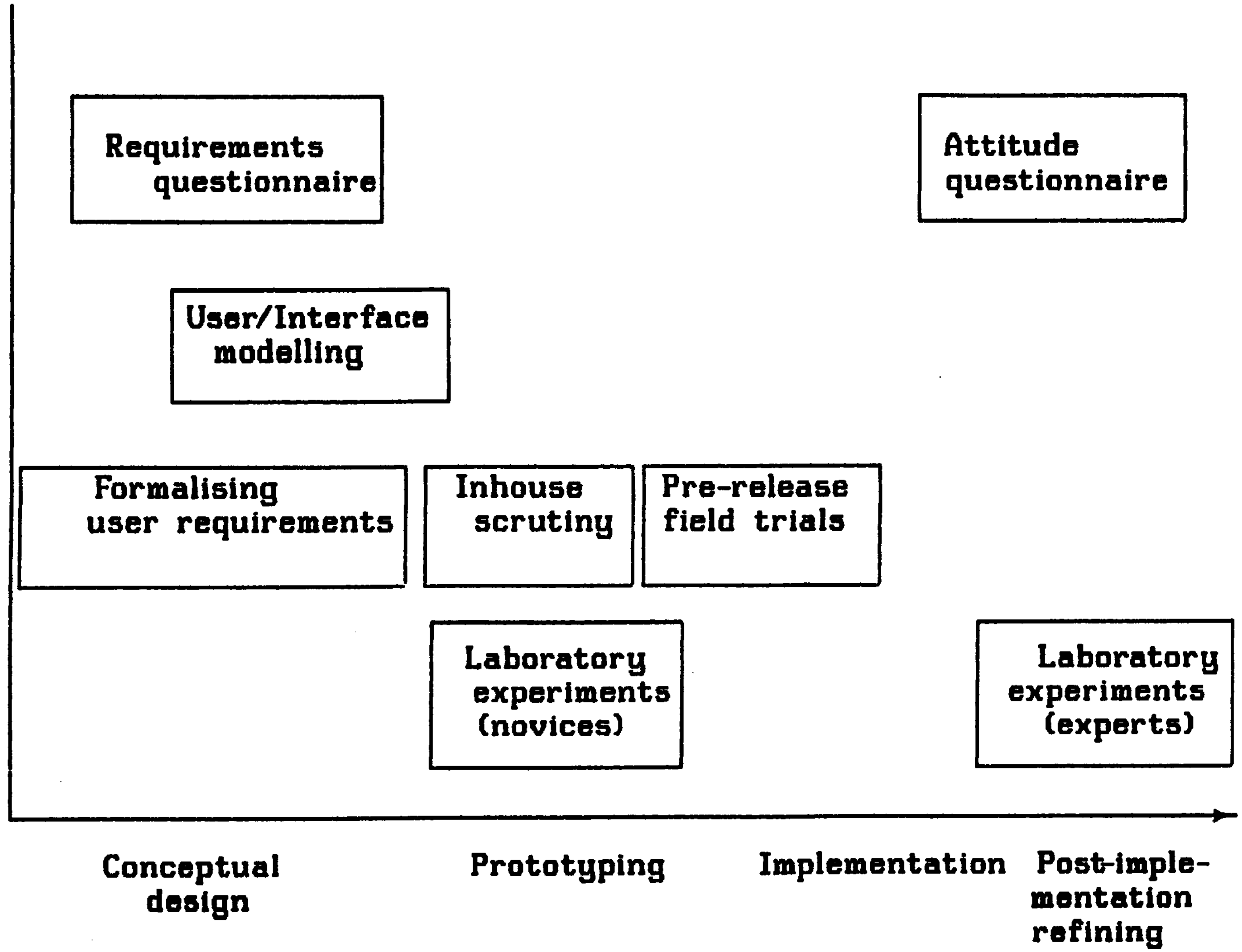
(6) Explanation facilities of both static (what the system can do) and dynamic (what the system is doing) nature to the user.

(7) Personalization, adjusting to the idiosyncrasies and preferences of the user.

5.7. The Integrated Use of Simple Techniques for System Evaluation.

Design and evaluation of human-computer interfaces is an iterative process in a system's development. Evaluation is seen by the author as an integral part of any system design and as such, the process of evaluation should start while a system is still in its conceptual stage and continue through to post-implementation. The stage in the project when the evaluation is conducted will tend to constrain the range of techniques which can be realistically considered, and the solution which can be taken to resolve the design issues being dealt with. For example, during the formalization of a product's requirements, simple questionnaires can be used to identify how to improve the ability of the product to perform the range of tasks required by the users. Once a formal representation of the dialogue has been established, interface characteristics and user models can be employed to identify and deal with any potential difficulties in the proposed design. Then, at the development stage, prototyping and both objective and subjective data can be collected. Leiser and Alberdi (1987) suggest that in-house use and scrutiny by work colleagues provides a useful informal test procedure. While field trials using pre-releases of the software can provide early end-user feedback, post-implementation evaluations fine tune the system for effective use within the working environment. These evaluations include attitude questionnaires (Furner,1987) and re-evaluations using expert users (see Figure 5.1). Although what is proposed here involves costs in terms of both manpower and time, justification for evaluation efforts such as these should be judged in terms of the ability of the software to fulfil the goals of the users in a usable way and to improve customer relations.

Figure 5.1. Evaluation effort during system life-cycle.



5.8. Summary and Conclusion.

The objective of the evaluation undertaken in this research has not been that of software preferences or performance but rather it has examined how closely the software has fulfilled the goals of the proposed user population and how well the machine functions match the mental functions of the human users. In this respect, fulfilling the goal of usability as proposed in Chapter 1.

Evaluation effort is seen by the author as an iterative process that runs in parallel to system design. The methods for evaluation will vary according to the stage of system development. At the conceptual stage, analysis should concern itself with the macro issues of matching the first level of the interface model between the goals of the user and the function of the system. The fourth evaluation principle ('design, test, measure, redesign and repeat') will ensure that by the post-implementation stage of analysis only micro alterations will be necessary to enhance usability. While laboratory experiments provide a means of evaluating software, it is easy to lose sight of the 'real-world' and so all designs should be tested by field trials. Field trials not only provide valuable feedback about operational performance but also provides a way of applying principle two ('work closely with the user population'). Therefore users feel they have some control and responsibility over the system design.

Many different evaluation techniques exist that provide a variety of data for analysis. Selecting suitable evaluation methods depends on the objectives of the evaluation. By far the most important single consideration is the end-users. While objective performance measures may indicate that system A is faster than system B at performing a particular benchmark, if users prefer to use system B then it is possible that overall performance will be better when system B is used. To ensure accurate results that can be validated and supported by statistical measures, evaluations should use as many different techniques as practically possible within the limitations of available resources. This will obviously require much effort on the part of the evaluating team.

Evaluations should consider novice and experienced software users as separate populations with different requirements. The expert moulds his mental functions to those of the machine, while the novice looks at the machine functions and tries to match them to his perceived method of solution. To develop usable systems, it is therefore necessary to design for the needs of the novice and then to refine designs for the efficient use by experts.

Chapter 6

Project Review and Proposals for Future Research.

Chapter 6.

Project Review and Proposals for Future Research.

6.1. Introduction.

In the final chapter the implications of this research project are examined and it is suggested how our knowledge of the areas discussed can be expanded by further research. To remain consistent throughout the thesis the three major topics of discussion (namely voice recognition, the HBP model and evaluation procedures) are documented in separate sections. The thesis title encompasses all three areas of this project in that:-

(1) Voice recognition was examined to predict whether its application would enhance usability in existing commercial software packages;

(2) user and interface models have provided a means of matching between system functionality and the mental functions of the users;

(3) evaluation has provided a tool for assessing the usability of human-computer interfaces.

6.2. Voice Recognition: A Question of Viability.

Having discussed both the benefits and the problems of implementing voice recognition systems, the lessons learnt as a result of this research program are discussed together with a look at the role of voice input in future interface development.

6.2.1. The Analysis of the Prototype System.

The prototype voice system provided a means of assessing the application of voice technologies in commercial systems. The system was successful in that it allowed the FEMVIEW package to be run almost totally by spoken commands. Although an evaluation of the system was made, this evaluation is by no means complete and further research is suggested. However, this study has highlighted some of the problems associated with both the technology, and also in implementing voice in existing commercial applications. During the course of this research, solutions have been found to some of the problems initially encountered. Such as a simple method for maintaining application/voicecard synchronisation, and the development of an application independent voice program for simple hierarchical dialogues. Within this particular application, the prototype system also offered a partial solution to the speaker dependency problem associated with current recognition systems. Because the number of word choices at any point in the dialogue was small, it was discovered that most users were capable of obtaining some recognition success irrespective of who had initially trained the system. By adding an on-line template training option to the software that allowed users to train the words that could not otherwise be recognised, the system became usable without the laborious half hour training procedure. At the end of each session the user was given the option to save the modified set of word templates. This meant that it was possible to effectively use the software by voice without ever needing to go through an initial training session. It could be argued therefore that the prototype system was, in part, speaker independent.

6.2.2. The Role of Voice Input in Interface Development.

In Chapter 3 a predictive forecast of the next generation of voice recognition systems is given. A word of caution is noted at this point. It is suitably illustrated by one respondent of the questionnaire that examined attitudes towards voice recognition:-

> 'We can make ourselves as unintelligible to computers as we are to our colleagues.'

The complexity of human-to-human communication is awesome. Even if voice systems can be developed to the point where recognition is as good as its human counterparts, there will still remain many problems to overcome, such as non-verbal protocols and the computers adopted role. This research suggests that speech input (even with perfect recognition) will be an imperfect method of data entry in many task domains. Booth (1982) outlines just some of the ambiguities in speech, such as homophonic and word context ambiguities.

6.2.3. Further Proposals for Evaluating Voice Systems.

Further research work is proposed to continue the survey of end-users attitudes towards voice technologies. Chapter 5 provides the framework for evaluating the prototype system in an objective way using benchmarking, attitude surveys, debriefs and so on. Practical problems associated with this evaluation include:-

(1) The sheer bulk of the equipment makes transportation to sites a problem.

(2) Using end-users (as opposed to students) is extremely difficult, since their time is very valuable.

(3) Evaluations of this nature take time to organise and conduct. This is compounded by having only one voice recognition system and training the voice card can take over half-an-hour per person.

(4) High subsistence and transport costs will be incurred.

Although benchmark tests in the laboratory can provide a good simulation of task environments, as Eason (1983) points out, field trials are important since they test predictions from simulations in a much harsher world. With a product such as a voice recognition system, the most important and fundamental test to its success in industry is its use in the field. It is in this area that future research activity using the prototype system should be directed.

6.2.4. Next Generation Voice Technologies.

The original aims of this research program in 1984 were to discover <u>if</u> voice technologies were viable in existing commercial products, <u>how</u> they could be implemented, and what, if any, were the <u>advantages</u>. This thesis has attempted to answer these questions, but at the same time it has generated many more, perhaps the most important being:-

> 'How close are we to the next generation of speech technologies?'

There is no simple answer to this question, but we have shown that the individual technologies already exist.

6.2.5. Findings from the Study of Voice Systems.

The results from this analysis suggest a rather pessimistic future for voice applications. Current voice recognition technologies will remain of little importance in the development of usable user interfaces over the next few years. Even looking towards the long term development of voice recognition, it is arguable whether it can ever totally replace the keyboard. The successful implementations are tolerant of the technology limitations, show clear benefits from the additional medium, and are accepted by the users.

Voice recognition and its associated technologies is a subject that has received much publicity in the past and, at least, over the next few years it will remain a topic of much research and development. To date, voice system development has been dictated by technology advancements. However, if voice systems are to be 'usable' a more application-oriented approach must be considered.

6.3. Developing the HBP Model and Applying it to Interface Development.

Chapter 5 offers only a taste of issues concerning the application of user models. A new model is presented based on Habitual Behaviour Patterns (HBP). Experimentation did not confirm or refute the hypothesis that user performance can be predicted by studies of personality traits and the author suggests the need for more detailed analysis. The following sections discuss methods of observing personality during interaction, the issues raised by the model and how the HBP model can be used in future self-adaptive personalised interfaces.

6.3.1. Manwatching: Looking for Personality Characteristics amongst Computer Users.

In looking for evidence of personality traits in human-computer interaction personality traits obtained by the 16 PF questionnaire were compared with users performance characteristics using the FEMVIEW software. To administer this type of experiment is difficult and time consuming. Some initiative for this approach came from simple observations of people using computers. It may well be possible to build on our, as yet, limited knowledge of personality related interaction by simply observing people using computers - a computer manwatch. One way of achieving this would be to take two classes of students that have never met. One class is then given the task of solving a simple benchmark test using a familiar software package (such as a word processor) while members of the second class selected one of the computer users and record all aspects of their interaction relating to personality. (It would be important to the success of the experiment that the class of computer users have no idea of the true purpose of the exercise.) Recorded observations would include speed of interaction, approach to problem solving, number of references to the manual, verbal comments reflecting attitude and so on. At the end of the session each observer would be asked to try to describe the personality of the user based on the observations made using the same vocabulary as used by the 16 PF (i.e. outgoing, experimenting, expedient and so on). Since no other knowledge of the users is known, the descriptions could only be based on the model built up from studying human-computer interaction. If these descriptions are accurate, then support is given to the hypothesis that human-computer interaction is affected by personality.

6.3.2. Issues Raised by the HBP Model.

Issues raised by the investigation based on the hypothesis include:-

(1) What traits are most important in human-computer interaction?

(2) Can traits be identified from interaction?

(3) How can the interface be altered to reflect the style of different personalities?

(4) Is a persons' personality in human-to-computer interaction the same as his/her human-to-human personality?

(5) How can interaction be analysed in order to develop a HBP model?

(6) How much control is the user given in determining his/her own user model?

6.3.3. Applying the HBP Model to Future Self-adaptive Personalised Interfaces.

The overall aim of this study is to recognise the key traits of an individual and relate these to the individual's interaction with the computer. If a correlation can be shown to exist, then the next and far more important step, in terms of interface evolution, is the virtual representation of an individual's personality within the interface and the development of self-adaptive interfaces based on these virtual representations. In this respect, the user model, as represented by the machine (or embedded user model), is a personalised one which describes the user in terms of preferred methods of interaction, learning strategies, expected response times, display preferences, used operations, method/detail of help and so on. For example, the stability/instability trait can be used in the interface to determine the length of time a user is left in a particular situation before assistance or further prompting is given. In the same way, the more impulsive, experimenting individual, having made an error, may be provided with an error message followed by a prompt to try something else, while the less impulsive (desurgent) individual may prefer a reference to a point in the manual, so as to make more calculated decisions. The interface will update the virtual model as more data is collected. The analogy here is with the use of the exponential method for sales forecasting. The method estimates future sales by the weighted examination of the previous sales figures. In a similar way, an estimate of how the user will act using the computer based on previous interactions can be made.

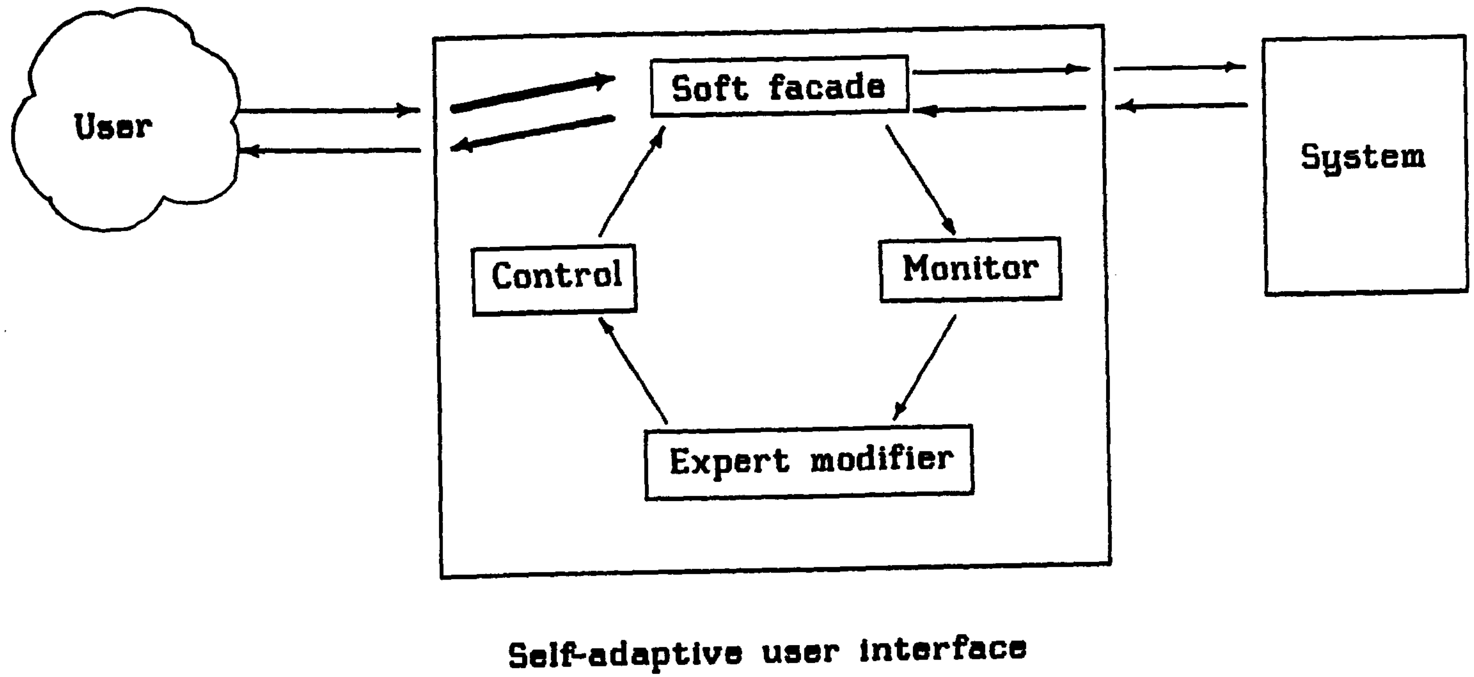
6.3.3.1. The Doors Metaphor for Personalising Interfaces.

Because users are reluctant to increase search effort and knowledge (Eason, 1983) and personalities rarely change, it may be possible to provide a self-adaptive, personalised interface. Although it is outside the area of this research to pursue this further, the methodology is described in Innocent (1982). The basic architecture of a self-adapting user interface is shown in Figure 6.1. One method of personalising the interface is presented by Sasso (1984). Here he uses a concept based around the metaphor of opening and closing doors. The user is provided with a primitive set of interactive processes and the facilities for creating, changing, deleting, opening and closing doors. In Sasso's system, the doors represent the combinations of primitive operations created by the user to simplify the interface for his individual needs. Applying the doors metaphor concept to the FEMVIEW software the current command keywords could be regarded as the primitive operations and doors could be visualised as icons representing commands or command sequences. Users are then able to generate command sequences and save them as icons. In a self-adaptive system, these doors would be generated automatically by the expert modifier (Figure 6.1) as a result of repeated recurrence, a repeated error, or a shortcut suggestion.

6.3.3.2. Self-Adaptive Interfaces Adopting the Role of Coach.

In practice, the way computer initiated information is presented to the users is a critical factor which will be very much influenced by

Figure 6.1. Architecture of a self-adapting interface.



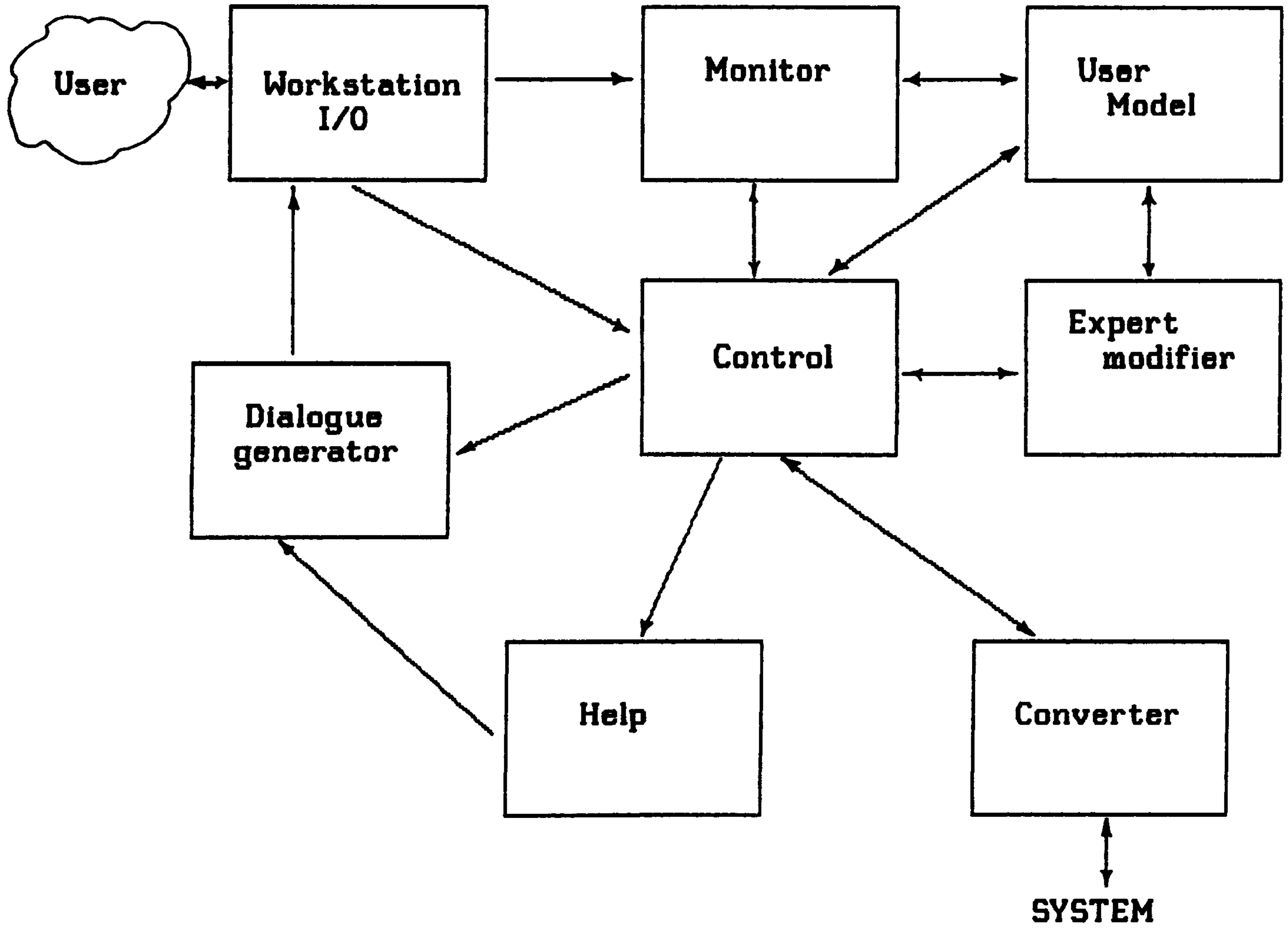
From Innocent (1982).

the personality of the individual. Card (1984) points out that the role the computer takes may influence the acceptance of the program by the user quite independently of the program's actual performance expertise. He cites two examples of systems where the computers takes on different roles in their method of presenting information, namely, the MYCIN system for diagnosis and treatment of blood diseases (which is strongly committed to being the partner who actually makes the decisions) and the WEST system for helping teach children arithmetic (which adopts the role of coach). In the latter system, advice is only given when the system is sure that the user is deficient in one particular area of knowledge (which includes using shortcuts) and then will only illustrate the point when a distinct advantage can be seen (Burton & Brown, 1982). Advanced interface systems using a knowledge-based approach are being developed towards this end with more complex components (see Figure 6.2). For more detail on adaptive and intelligent interfaces see Benyon (1984), Innocent (1982), Edmonds (1981), Edmonds & Guest (1984), Rissland (1984).

6.3.4. The Use of Personalised User Models for Occupational Assessment.

The HBP model provides one means of building knowledge of users by studying their interaction with the computer. As techniques are developed for building personalised user models it is possible to perceive future computer interfaces taking on the role of a psychologist. While interacting with the computer in order to achieve some specified task, the user is being analysed by the computer and a model is built of the user's aptitude and personality. This information could then be used as the basis for occupational

Figure 6.2. Components of a knowledge-based interface system.



From Benyon (1984).

assessment in the same way as occupational psychology tests are currently used for selection in some large companies (SHL, 1987, ASE, 1987). This may even raise the question of confidentiality in future, computer-initiated user models.

6.3.5. Applied Psychology in Interface Design.

Personality affects the way we as individuals act, how we handle different situations and how we respond to different stimuli. Much development work by psychologists has been conducted to identify personality features and use them for selection purposes. The aim of this research has been to extend personality assessment into interface design. Further work will require close links between psychologists and computer scientists before effective working systems can be developed. The HBP model has merely scratched the surface of what is, an exciting, new approach to interface design and evaluation.

6.4. Providing Evaluation Techniques for Assessing the Usability of Human-Computer Interfaces.

While the need for evaluation in the development of better systems has been repeated time and time again in recent literature, there still remains very little evidence to suggest that formal evaluations are being carried out to any great extent in developing software. The author's research has attempted to use many of the available evaluation techniques to assess their usefulness for aiding system design and assessing the usability of human-computer interfaces.

6.4.1. Recommendations for Producing Usable Systems.

In Chapter 1 usability was defined as the fulfilment of user requirements and matching between functionality and cognitive ability. Usability can be achieved by matching object manipulation by the system with real world tasks, machine functions with the mental functions most directly involved in human-computer interfacing and the physical devices with the sensori-motor processes of the human.

Evaluation provides the tools for developing usable systems and is seen by the author as an iterative process running in parallel with design. In the study of usability, five key recommendations have emerged:-

(1) Provide continuous evaluation and reassessment.

(2) Design for the inexperienced user but refine for the expert.

(3) Evaluate using both laboratory experimentation and field trials (since one without the other leaves gaps in our knowledge of usability).

(4) Cross validate results from different evaluation techniques to ensures that the true performance characteristics are reflected.

(5) Understand the users, as individuals, and adapt to their preferred methods of interaction.

6.4.2. Developing General Techniques for Evaluation.

During the course of this research program a variety of techniques for collecting data for evaluation have been used. Much time and effort has been spent in the development and administration of these techniques. Even a seemingly straight forward questionnaire will require a lot of work to ensure that users are not led by the questions and the relevant questions are presented. While designers and software engineers are eager to produce usable systems, an approach to system design, such as described in this thesis, may be difficult to carry out in practice. This is because time is usually a limited resource, and because designers often feel that they haven't the expertise necessary to conduct extensive evaluations. No instant solutions to these two problems are offered although efforts can be made to provide generalised tools to help make conducting evaluations easier. One such attempt is presented by the debrief questionnaire developed by Poulson (1987(a), 1987(b)) (and used with some success in this research for debriefing users after interacting with the software). The questionnaire provides simple indices for the perceived quality of software interfaces using a favourability scale. In a similar way, section 5.4.4 has documented a method of objectively studying protocol analysis and in section 3.6 key operators in on-line logging are presented. Other similar generalised tools could include simple frameworks for developing evaluation benchmark tests and questionnaires. It may also be possible to develop computer programs for conducting software evaluations working in a similar way to current debugging software. It is only with the development and general use of such tools that future systems may benefit from the current research in evaluation techniques and methodologies.

6.5. Conclusion.

As we strive towards the 'quality' interfaces of the future that fulfil the goal of usability, we must look beyond the current limitations of existing technologies and design philosophies. This research suggests that computers should treat users as individuals. Therefore, computers must adjust to our individual idiosyncrasies and shortcomings. If this is possible, then it will not only enhance the human-computer interface but may also change our perceived view of computers from their current role as servants to our needs to mentors in our quest for knowledge.

oooo0000oooo

References.

References.

A.S.E., 1987, Assessment and Selection for Employment Catalogue. NFER-Nelson, Windsor, Berkshire.

Allen, R. B. & Scerbo, M. W., 1983, Details of Command Language Keystrokes. ACM Trans. on Office Information Systems, vol. 1, pp. 159-178.

Alty, J. L., 1984, The Application of Path Algebras to Interactive Dialogue Design. Behaviour and Information Technology, vol.3, no. 2, pp. 119-132.

Anderson, T., Moore, T. & McKinley, R., 1985, Issues in the Development and Use of a Speech Recognition Data Base for Military Cockpit Environments. Proceedings of Speech Technology '85. Media Dimension Inc., New York.

Anmed, W., 1985, Computer Recognition of Cerebral Palsy Speech. Proceedings of Speech Technology '85. Media Dimension Inc., New York.

Ashton, R., 1985, Voice Input of Warehouse Inventory. Proceedings of Speech Technology '85. Media Dimension Inc., New York.

Barnard, P. J., Hammond, N. V., Morton, J. & Long, J. B., 1981, Consistency & Compatibility in Human-computer Dialogue. International Journal of Man-machine Studies, vol. 15, pp. 87-134.

Beail, N., 1985, Repertory Grid Technique and Personal Constructs. Croom Helm publishers, Beckenham, Kent.

Benyon, D., 1984, Monitor. A Self-Adaptive Interface. Proceedings of Interact'84, Conference on Man-Machine Interfaces. pp. 307-313.

Biermann, A. W., Rodman, R. D., Rubin D. C., & Heidlage, J. F., 1985, Natural Language with Discrete Speech as a Mode to Human-to-machine Communication. Communications of the ACM, no. 28, pp. 628-636.

Booth, A. W., 1982, The Resolution of Ambiguities and the Correction of Errors in the Automatic Transcription of Palantype. Ph.D. Thesis, Leicester Polytechnic.

Brady, J., 1986, A Theory of Productivity in the Creative Process. IEEE Computer Graphics & Automation, May, pp. 25-33.

Bramer, B., 1983, Fortran Based Machine Independant User Input Processor. Proceedings of The User Interface: The Ergonomics of Interactive Computing. Leicester Polytechnic, 14-16th. September.

Bramer, B, 1984, Fortran Based Command Processor User Manual: Version 4. Internal document. Human Computer Interface Research Unit, Leicester Polytechnic. 19th. October.

Bramer, B., & Rickett J. D., 1984, A Report on the Visit to Lloyds Register of Shipping & Their Voice Recognition Systems. Internal document, Human Computer Interface Research Unit, Leicester Polytechnic.

Broad, D. J.& Shoup, J. E., 1974, Concepts for Acoustic Phonetic Recognition. In Speech Recognition. Reddy, D. R (Ed.), Academic Press, London, pp.243-274.

Brundage, W. J., 1985, Consumer Applications of Speech Recognition. Proceedings of Speech Technology '85. Media Dimension Inc., New York.

Burger, R. M., 1984, The Impact of ICs on Computer Technology., IEEE Computer, vol. 17, no. 10, October, pp. 88-95.

Burton, R. R. & Brown, J. S., 1982, An Investigation of Computer Coaching for Informal Learning Activities. In Intelligent Tutoring Systems. Sleeman D. & Brown S. J.(Eds.), Academic Press, New York.

Card, S. K., 1984, Human Factors and the Intelligent Interface. Presented at Combining Human and Artyificial Intelligence: A New Frontier for Human Factors, Symposium for the Metropolitan Chapter of the Human Factors Society, New York, 15th. November.

Card, S. K., Moran, T. P. & Newell A., 1980, The Keystroke-Level Model for User Performance Time with Interactive Systems. Comms. of the ACM, vol 23, no. 7, July, pp. 396-410.

Card, S. K., Moran, T. P. & Newell A., 1983, The Psychology of Human-Computer Interaction. LEA publications, London.

Card, S. K. & Young, R., 1985, Predictive Models of the User: A selective review. In preparation.

Carroll, J. M. & Mack, R. L., 1984, Learning to Use a Word Processor: By Doing, by Thinking, and by Knowing. In Human Factors in Computer Science. Thomas & Shneider (Eds.), Ablex Pub.

Cattell, R. B., Elser, H. W. & Tatsuoka, M. M., 1970, The 16PF Handbook. Institute for Personality and Ability Testing, Inc. Champaign, Illinois.

Clarke, A. A., 1986, A Three-Level Human-computer Interface Model. International Journal of Man-machine Studies, vol. 24, pp. 503-517.

Cochran, D. J., Riley, M. W. & Stewart L. A., 1980, An Evaluation of the Strengths, Weaknesses and Uses of Voice Input Devices. Proceedings of the Human-Factors Society - 24th. Annual Meeting. Los Angeles.

Cohen, P. S. & Mercer, R L, 1974, The Phonological Component of an Automatic Speech-recognition System. In Speech Recognition. Reddy, D. R. (Ed.), Academic Press, London. pp.243-274.

Connor, D., 1985, Information System Specification & Design Road Map. Prentice Hall, New Jersey.

Cooke, N., 1984, The Flight Testing and System Integration of Direct Voice Input Devices in the Flight Deck. Royal Aircraft Est. Bedford.

Eason, K. D., 1983, Towards the Experimental Study of Usability. Proceedings of The User Interface: The Ergonomics of Interactive Computing. Leicester Polytechnic, 14-16th. September.

Eason, K. D., 1984, Towards the Experimental Study of Usability. Behaviour and Information Technology, vol. 3, no. 2, pp. 133-143.

Edmonds, E. A., 1981, Adaptive Man-Computer Interfaces. In Computing Skills and the User Interface. Coombs, M. J. & Alty, J. L. (Eds.), Academic Press, London.

Edmonds, E. A., 1982, The Man-Computer Interface: A Note on Concepts and Design. International Journal of Man-machine Studies, vol. 16, pp. 231-236.

Edmonds, E. A. & Guest, S. P., 1984, The Synics User Interface Manager. Presented at Interact,'84, IEE, Savoy Place, London.

Ellis, A., 1946, The Validity of Personality Questionnaires. Psychological Bulletin, no. 43, pp. 385-426.

Erman, L. D., Hayes-Roth, F., Lesser, V. R. & Reddy, D. R., 1980, The Hearsay-II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty. Computer Surveys, vol. 12 no. 2, June.

Femview, 1982, FEMVIEW User Manual. Femview Ltd., 4 West Cottages, Stretton Lane, Houghton-on-the-Hill, Leicester.

Furner, S. M., 1987, Practical Information about Interface Operating Characteristics for Engineering Design. Proceedings of the colloquium on 'Evaluation Techniques for Interactive Systems Design: 1' IEE Digest no. 1987/38. 27th. March.

Gentner, D. & Stevens, A. L., 1983, Mental Models. Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Goldhor, R. S., 1985, A Large Vocabulary Highly Programmable Speech Recognition System. Proceedings of Speech Technology '85. Media Dimension Inc., New York.

Gould, J. D. & Lewis, C., 1983, Designing for Usability -- Key Principles and What Designers Think. In Human Factors in Computing Systems. A. Janda (Ed.), North-Holland, Oxford.

Hammond, N. V., Long, J. B., Morton, J., Barnard, P. J. & Clarke, I. A., 1981, Documenting Human-Computer Mismatch at the Individual and Organisational Levels. Hursley Human Factors Laboratory, HF040, March. (Hammond is a member of the MRC Applied Psychology Unit, Cambridge.)

Harris, L. R., 1978, The ROBOT System: Natural language Processing Applied to Database Query. In Proceedings to ACM 78.

Hayes, P., Ball, E. & Reddy, R., 1981, Breaking the Man-machine Communication Barrier. Computer, vol. 14, no 3, pp. 19-30.

Howard, S. & Murray, D. M., 1987, Experiences in Evaluating the Human-computer Interface. Proceedings to the colloquium on 'Evaluation Techniques for Interactive Systems Design:1' IEE Digest no. 1987/38 27th. March.

Innocent, P. R., 1982, Towards self-adaptive Interface Systems. International Journal of Man-machine Studies, vol. 16, pp. 287-299.

Itakura, F., 1975, Minimum Prediction Residual Principle Applied to Speech Recognition. IEEE Trans. Acoustic, Speech, Signal Process. ASSP-23, pp. 67-72.

Johnson, S., Connolly, J. & Edmonds, E. A., 1984, Spectrogram Analysis: A Knowledge Based Approach to Automatic Speech Recognition. LUTCHI Research Centre, University of Technology, Loughborough.

Joost, M. G. & Petry, F. E., 1979, Design Factors in the Use of Isolated Word Recognition for Speech Therapy. Proceedings, Human Factors Society, pp. 30-34.

Joslin, E. O., 1965, Application Benchmarks: The Key to Meaningful Computer Evaluations. Proc. of the National Conference. ACM, vol. 20, Cleveland, Ohio, August 24-26th., pp. 27-37.

Kelly, G. A., 1955, The Psychology of Personal Constructs. vols. 1 & 2, Norton, New York.

Kidd, A. L., 1982, Man-Machine Dialogue Design. British Telecom Research Laboratory, Ipswich (Pub.).

Killmon, P., 1986, Speech Recognition Advancements Bring Cybernetics to the Boardroom. Computer Design, July.

Krech, D., Crutchfield, R. S. & Livson, N., 1969, Elements of Psychology. 2nd. Ed., Knopf, New York.

Krug, S., 1981, Interpreting 16PF Profile Patterns. Institute for Personality and Ability Testing, Inc. Champaign, Illinois.

Kurzweil, R., 1985 The Kurzweil Voice Writer. A Large Vocabulary Voice Activated Word Processor. Proceedings of Speech Technology '85. Media Dimension Inc., New York.

Laporte, C., 1985, A Description of the Architecture of a Voice I/O Evaluation System and its Application in Military Aircraft. Proceedings of Speech Technology '85. Media Dimension Inc., New York.

Leggett, J. & Williams, G., 1984, An Empirical Investigation of Voice as an Input Modality for Computer Programming. International Journal of Man-Machine Studies, vol. 21, pp. 493-520.

Leiser, R. G. & Alberdi, M., de, 1987, Developing Voice Control Dialogues for Multi-feature PBX. Proceedings to the colloquium on 'Evaluation Techniques for Interactive Systems Design:1' IEE Digest no. 1987/38 27th. March.

Lewis, N. C. & Crews, A. E., 1985, The Evolution of Benchmarking as a Computer Performance Evaluation Technique. MIS Quarterly, March, pp. 7-16.

Logica, 1984, LOGOS Direct Input in the Cockpit. Logica Space and Defence Systems, Cobham, Surrey.

Lund, M. A., 1985, Evaluating the User Interface: The Candid Camera Approach. In Human Factors in Computing Systems-II. Borman L. & Curtis B. (Eds.), North-Holland, Oxford.

Mac an Archinnigh, M., 1985, A Model of a User's Conceptual Model of... in Pfaff, G. E. (Ed,): User Interface Management Systems. Springer, New York.

Mack, R. L., Lewis, C. H. & Carroll, J. M., 1983, Learning to Use Word Processors: Problems and Prospects. ACM Trans. of Office Information Systems, vol. 1 no. 3, July.

Maguire, M., 1984, An Introduction to Dialogue Design. Seminar given at Leicester Polytechnic. Martin Maguire is a member of HUSAT, Dept. of Human Sciences, University of Technology, Loughborough.

Majchrzak, A., Chang, T-C., Barfield, W., Eberts, R. & Salvendy, G., 1986, Human Aspects of CAD. Taylor Francis Pub. London.

Marsh, G., 1984, Don't Just Sit There - Talk to Me. Defence Helicopter World (June-August).

Martin, G. L., 1987, The Utility of Speech Input in User-Computer Interfaces. To be published in International Journal Man-Machine Studies.

Miller, R. B., 1968, Response Time in Man-computer Conversational Transactions. Proceedings Spring joint Computer conference, 33, AFIPS Press, Montvale, New Jersey, pp. 267-277.

Monk, A., 1985, Fundamentals of Human-Computer Interaction. Academic Press, London, pp. 69-79.

Morris, M. F. & Roth, P. F., 1982, Computer Performance Evaluation: Tools and Techniques for Effective Analysis. Van Nostrand Reinhold Company, London.

Morton, J., Barnard, P. J., Hammond, N. V. & Long, J. B., 1979, Interacting with the Computer: A Framework. In Teleinformatics'79. Boutmy E. J. & Danthine A. (Eds.), pp. 201-208.

Myers, I. B., 1962, The Myers-Briggs Type Indicator. Consulting Psychologists Press, Inc. Palo Alto, California.

Neal, A. S. & Simons, R. M., 1983, Playback: A Method for Evaluating the Usability of Software and its Documentation. In Human Factors in Computer Systems. Janda A. (Ed.), North-Holland, Oxford, pp. 78-82.

Newell, A. & Simon, H. A., 1972, Human Problem Solving. Prentice-Hall, New Jersey.

Norman, D. A., 1983, Design Rules Based on Analysis of Human Error. Communications of the ACM, vol. 26 no. 4 April, pp. 254-258.

Nye, J. M., 1982, Voice Integration: The Critical Mass. In Proceedings of the Voice Data Entry Systems Applications Conference. AVOIS. September.

Palomo, J. M., 1987, Computer Programmer Aptitude Battery. In Assessment & Selection for Employment (ASE) Catalogue, Windsor, Berkshire.

Pancas, D. L., 1969, On the Use of Transition Diagrams in the Design of a User Interface for an Interactive Computer System. Proc. of the National ACM conference, pp. 379-385.

Penniman, W. D. & Dominick, W. D., 1980, Monitoring and Evaluation of On-line Information System Usage. Information Processing & Management, vol 16, pp. 17-35.

Petry, F. E.& Joost, M, G,, 1981, Computerized Speech Training Aids. Proceedings, IEEE Southeastcon, pp. 248-250.

Pols, L. C. W., 1971, Real time Recognition of Spoken Words. IEEE Trans. Computing. C-20, pp. 972-978.

Poock, G. K., 1982, Voice Recognition Boosts Command Terminal Throughput. Speech Technology, vol. 1, pp. 50-57.

Poulson, D. F., 1987(a), Towards Simple Perceived Indices of the Perceived Quality of Software Interfaces. Proceedings to the colloquium on 'Evaluation Techniques for Interactive Systems Design: 1', IEE Digest no. 1987/38 27th. March.

Poulson, D. F., 1987(b), Towards Simple Measures of the Perceived Quality of Software Interfaces. (Internal document) HUSAT Research Centre, Loughborough.

Ramsey, H. R. & Atwood M. E., 1979, Human Factors in Computer Systems: A Review of the Literature. Tech. report SA1-79-111-DEN. Science Applications Inc. 7935E Prentice Ave. Englewood, Colorado.

Rich, E., 1983, Artificial Intelligence. McGraw-Hill, London.

Rickett, J. D. & Bramer, B., 1986, Problems with Adapting Commercial Software Packages to Use Voice Recognition for Command Input. Centre for Applied Research in Information Systems (CARIS), MCS Research Unit, Leicester Polytechnic.

Rissland, E. L., 1984, Guest Editorial: Intelligent User Interfaces. International Journal of Man-machine Studies, vol. 21, pp. 279-282.

Roberts, M., Kirby, D. & Candy, L., 1983, Evaluation of the Human-computer Interface. Proceedings of The User Interface: The Ergonomics of Interactive Computing. Leicester Polytechnic, 14-16th. September.

Roberts, T. L. & Moran, T. P., 1984, The Evaluation of Text Editors: Methodology and Empirical Results. Comms. of the ACM, vol 27, pp. 265-283.

Rodman, R., 1985, Speech Recognizer Performance with Dysarthric Speakers : A Comparison of Two Training Procedures. Proceedings of Speech Technology '85. Media Dimension Inc., New York.

Root, R. W. & Draper, S., 1983, Questionnaires as a Software Evaluation Tool. Proceedings of the CHI conference on Human Factors in Computing Systems, pp. 83-87.

S.H.L., 1987, Occupational Tests and Training Courses. Saville & Holdsworth Ltd., Occupational Psychologists, Esher, Surrey.

Sasso, R., 1983, Personalising the Software Interface. In Draft. Oxford University Computing Laboratory Group. Programming Research Group.

Savage, R. E. & Habinek, J. K., 1984, A Multi-level Menu-driven Interface: Design & Evaluation Through Simulation. In Human Factors in Computer Systems. Thomas & Shneider (Eds.), Ablex Pub., London, pp. 165-186.

Self, J., 1974, Student Models in CAI. International Journal of Man-machine Studies,, vol. 6, pp. 261-276.

Shackel, B., 1981, The Challenge of the Non-expert User. In User Friendly Systems. Pergamon Info, (pub.).

Shaw, M. L. G., 1980, On Becoming a Personal Scientist. Academic Press, London.

Shneiderman, B., 1980, Software Psychology. Winthrop London.

Shortliffe, E. H., 1976, Computer-based Medical Consultations: MYCIN. Elsevier, New York.

Siegal, S. S., 1956, Nonparametric Statistics for the Behavioral Sciences. McGraw-hill, London.

Singh, N., 1987, Voice Recognition for Dental Surgeons M.Sc. Man-computer Interaction, Project Report. School of Maths, computing and Statistics, Leicester Polytechnic.

Sommerville, I., 1985, Software Engineering. 2nd. Ed., Addison-Wesley Pub., Wokingham, England.

Stevens, G. C., 1983, User-friendly Systems? A Critical Examination of the Concept. Behaviour & Information Technology, vol. 2, no. 1, pp. 3-16.

Tauber, M. J., 1987, Top-Down Design of Human-Computer Interfaces. To be published in Visual Languages. Chomp, Ichikawa, Lipomenides (Eds.). Plenum Press, New York.

Thimbleby, H., 1985, User Interface Design: Generative User Engineering Principles. In Fundamentals of Human-Computer Interaction. Monk A. (Ed.). Academic Press, London

Thomas, C., 1987, Evaluation of a Technique for Usability Assessment in Interactive Systems. Proceedings of the colloquium on 'Evaluation Techniques for Interactive Systems Design: 1'. IEE Digest no. 1987/38. 27th. March.

Velichko, V. M. & Zagoruyko, N. G., 1970, Automatic Recognition of 200 Words. International Journal of Man-Machine Studies, vol. 2, pp. 223-234.

Vernon, P. E., 1966, Personality Assessment. A Critical Survey. Methuen & Co., London EC4.

Visick, D., Johnson, P., & Long, J., 1984, The Use of Simple Speech Recognisers in Industrial Applications. Proceedings of INTERACT'84, First IFIP Conference on Human-Computer Interaction, London.

V.S.I., 1986, Data Vox Auditor: A Major Saving in Stock Auditing Cost. Voice Systems International Ltd., (product information), The Science Park, Cambridge.

Waterman, J., 1985, Speech Communication: How to Use It. In Fundamentals of Human-Computer Interaction. Monk A. (Ed.). Academic Press, London, pp. 221-236.

Weinberg, G. M., 1971, The Psychology of Computer Programming. Van Nostrand Reinhold Co, Berkshire, England.

Weizenbaum, J., 1966, ELIZA - A Computer Program for the Study of Natural Language Communication between Man and Machine. Communications of the ACM, vol. 9, no. 1, January, pp. 36-44.

Wetherill, G. B., 1982, Elementary Statistical Methods. Chapman & Hall, London, pp. 129-161.

Wilensky, R., 1982, Talking to UNIX in English: An Overview of UC. In Proceedings AAAI2.

Young, R. M., 1985, User Models as Design Tools for Software Engineers. Handout for the Alvey Workshop on MMI/SE, September.

Appendices.

A1

User attitude scale to voice recognition.

User Attitude Scale to Voice Recognition.

In order to represent some quantification of computer users' attitudes towards voice recognition products and the software which uses these systems the following simple questionnaire has been constructed.

Your co-operation in answering these questions is requested, it should only take a few minutes of your time and will be of great value to an ongoing research project.

Room is available beside each question for you to add your own comments if you wish.

Section A - Your Experience.

Give each of the following questions a rating between 1 and 5.

5 = Very experienced.
4 = Experienced.
3 = Some experience.
2 = A little experience.
1 = No experience.

A1/ How experienced are you with using computers?
[comments]

☐

A2/ How familiar are you with the FEMVIEW product?
[comments]

☐

A3/ How experienced are you with voice recognition systems?
[comments]

☐

Section B - Favourability Scales.

Give each statement a rating between 1 and 5.

5 = I strongly agree.
4 = I agree with reservations.
3 = I do not agree or disagree.
2 = I disagree to some extent.
1 = I strongly disagree.

B1/ 'Using the FEMVIEW system by spoken commands is very easy.'
[comments]

B2/ 'Using the FEMVIEW system by voice requires a lot of concentration.'
[comments]

B3/ 'Voice recognition is a feature that would enhance most systems.'
[comments]

B4/ 'Voice input is a fast, effective mode of input.'
[comments]

B5/ 'The percentage of words that were recognised correctly is high enough for voice input to be a viable method of data entry in most general application.'
[comments]

Section C - Applications and Future.

(Answer the following questions in the format requested.)

C1/ Would you use voice facilities if it were made available on commonly used software such as Word processors?

Always,
Occasionally,
Never. (tick as applicable)

If your answer to this question is occasionally state instances when you feel you would use voice input rather than some other mode of input.

C2/ What advantages, if any, do you see voice entry providing?

a/ In this application (FEMVIEW):-

b/ In general applications:-

C3/ What drawbacks, if any, do you see the current voice system having?

a/ In this application:-

b/ In general applications:-

C4/ Rank the following input devices in order of speed of operation. If you are unfamiliar with any of these devices leave the box blank. Give devices of equal speed the same value.

1 = fastest device ... & ... 5 = slowest device.

		[comments]
Keyboard entry		
Mouse entry		
Voice entry		
Touch pad entry		
Tablet & pen (with overlays)		

C5/ Please add a short statement expressing your attitude towards voice input in general. Is it a good or bad method of data entry? Should we continue to concentrate our research in this field?.....and so on...

C6/ Can we ever achieve the so called '5th. Generation' of computers where we talk to computers in the same manner as we would talk to a friend or work colleague?

YES/NO

(tick as applicable)

Add comments on this point if you wish.

Thank-you for completing this questionnaire. Your opinions on this subject are of great importance to those of us interested in this area of research.

A2

Summary of data collected by the user attitude questionnaire towards voice recognition.

Summary of data collected by the user attitude questionnaire towards voice recognition.

A/ Your experience.

5 = very experienced
4 = experienced
3 = some experience
2 = little experience
1 = no experience.

	5	4	3	2	1
With computers.	5	2	-	-	-
With FEMVIEW.	-	1	1	1	4
With voice input.	-	1	3	1	2

B/ Attitude towards voice (favourability scale).

5 = Strongly agree
4 = agree with reservations
3 = do not agree or disagree
2 = disagree to some extent
1 = strongly disagree.

	5	4	3	2	1
FEMVIEW is easy to use by voice.	1	5	1	-	-
FEVIEW by voice requires a lot of concentration.	-	3	2	1	1
Voice input enhances most software.	-	4	-	2	1
Voice is a fast effective mode of input.	-	2	2	2	1
Voice error rates are acceptable.	-	2	3	-	1

C1/ Would voice systems be used.

Always	0
Occasionally	6
Never	1

When would voice be applicable:-

Demos or evaluations.

Searching and replacing text but cursor control might be awkward using voice.

When speed of input is quicker than the keyboard.

When hands are busy (I find typing easy).

Easy work e.g. Word processing.

If application not confidential and it won't disturb others.

C2/ Advantages of voice.

In FEMVIEW	In general
Management acceptance	Disabled gain access to computer
Demos and evaluations	Allows communication over telephone.
More choice	Hands busy, non critical applications.
Quicker than keyboard for inexperienced	Quicker for inexperienced keyboard users.
More natural interface	More natural interface.
Simplifies task of repeatedly entering the same command over and over again.	Allows hands-off data entry.
	Good for industrial/secretarial work (e.g. telephone etc.)

C3/ Disadvantages of voice.

In FEMVIEW	In general
Discrete word, speaker dependant.	Poor recognition accuracy.
Slower than keyboard.	Difficulties lead to dissatisfaction.
Slow because of incorrect parsing (poor recognition).	High error rates.
Difficulties lead to dissatifaction.	Lack of generality between users.
Similar commands get confused.	Disrupted by sore throat, background noise.
Complicated commands needed.	Trained to single persons voice.

C4/ Perceived Speed of input.

Keyboard	1	2	2	1!1	2!2	5	3	16/7	2.3
Mouse	3	1	1	2	1	4	2	14/7	2.0 fastest
Voice	3	3	3	3	3	3	2	20/7	2.9
Touch Pad	3	5	-	2	4	-	3	17/5	3.4 slowest
Tablet & Pen	2	4	-	-	2	-	1	9/4	2.3

Comments.

!1 Application dependant.
!2 User dependant.

C5/ Attitudes towards voice input (general comments).

Present technologies make voice restrictive, until speaker independant systems that use large vocabularies with natural language processing are available few applications will benefit greatly from voice.

For applications such as word processing the ambiguity of sentences plus the difficulty of translation of voice to machine language makes it impractical. For applications using a limited subset of spoken English with little or no semantic ambiguity voice input becomes viable.

Not good for computer people but o.k. for general public.

Good for non-critical users and/or nonexpert users.

Whichever way you look at it voice input is very useful and everyday it is becoming more advanced.

It is potentially a good method - but has a way to go. It should seem "natural" to "talk" to the computer if it is to be successful.

Voice input is useful for certain applications (i.e. aircraft). We need better technology until it can be widely implemented in the office, i.e. more research needed.

C6/ The future of voice.

To be acceptable, computers will have to be better at understanding than humans. This will require a lot of work and will probably make end systems ask a lot of questions.

We can make ourselves as unintelligible to computers as we are to our colleagues.

Too difficult.

Don't Know.

Who can say? In this field we seem to perform more & more incredible tasks: there is no real justification for dismissing the 5th. generation concept.

A long way off. Artificial Intelligence problem, vast amounts of knowledge needed. We do not understand fully yet the cognitive processes in communication.

A3

Papers.

Centre for Applied Research in Information Systems

School of Mathematics Computing & Statistics

Leicester Polytechnic

P O Box 143

Leicester LE1 9BH

WORKING PAPER NUMBER 11

Problems with adapting commercial software packages to use voice recognition for command input

by

J Rickett and Dr B Bramer

September 1986

CARIS WORKING PAPERS

Copies of this and other CARIS working papers may be obtained from:-

The MCS Research Unit
School of Mathematics
Computing & Statistics
Leicester LE1 9BH

OR, on loan from:-

Kimberlin Library
Leicester Polytechnic

Titles in the series to date are:-

1. Parkin A "The System Generator as a Research Tool for Methodology"

2. Parkin A "System Development Tools: Their Nature and Usefulness"

3. Parkin A "Induction of the Decision Rules of an Expert System by Means of Decision Tables"

4. Parkin A "Addition of Probabilistic Inferencing to an Expert System Shell Based on Decision Tables"

5. Benyon D "Justification and Outline of a Knowledge-based Interface System"

6. Skidmore S & Mills G "Computer Aided System Design - the CARISmatic Framework"

7. Wroe B "A Chasm Model for the Successful Design and Implementation of Information Systems in Small Companies"

8. Walmsley D "A Computer Based Training System for a Course in Systems Analysis & Design"

9. Benyon D & Skidmore S R "Towards a Toolkit for the Systems "

10. Parkin A "Towards the data-types needed in a system generator"

11. Rickett J & Dr Bramer B "Problems with adapting commercial software packages to use voice recognition for command input"

Problems with adapting commercial software packages to use voice recognition for command input

John Rickett and Dr. Brian Bramer,
School of Mathematics, Computing and Statistics,
Leicester Polytechnic.
March 1986

Abstract. Commercial voice recognition systems have been available since the mid-1960's as plug-in cards for microcomputers or as add-on black boxes connected to existing computer systems . To date, many of the commercial voice systems have had limited vocabularies, poor recognition rates, and have been very expensive. As a result few commercial organisations have considered the use of voice systems as an alterative mode of data entry except in very simple, or highly specialised applications.

As part of a Ph.D research program, a commercial CAD package was adapted to use several command input techniques including voice. The package, as supplied, makes use of a powerful keyboard driven command processor, which is simple enough to be used by inexperienced users and still allows expert users to enter commands in a very concise form for fast interaction. This paper describes some of the problems encountered in implementing the voice command input and the techniques evolved in overcoming them. Although the software was a CAD engineering package, many of the problems discussed will apply to other, more general, application areas.

1. Introduction.

The use of voice technologies in simple applications include dialless telephones, speech operated robots, speech controlled hi-fi systems and other household machines, such as voice operated home sewing machines (Brundage,85, Yoshimura,85). Specialised applications include cockpit management in helicopters and aircraft (Marsh,84, Logica,84, Cooke,84, Anderson et al,85), although still at an experimental stage (Laporte,85) and in areas of health and medicine (Anmed,85, Rodman,85). In Japan, speech products are developing along similar lines (Uenohara,85).

With advancing chip technology (Burger,84), the performance/price ratio of voice systems is improving and increasingly vendors are offering plug-in card options (Votan,84, Viglione,85), speech recogniser add-on peripherals (Marconi,84, Kode,82), or even a standard voice facility built into the keyboard of existing computers. For example, speaker dependant, discrete word voice recognition facilities are already included as standard in several home computers such as the Apricot Portable and Epsom PC computers. Companies have also been formed to provide alternative voice activated keyboards for existing computers (Caratech,84).

The development of systems with improved template matching algorithms (Johnson et al,84) and far greater vocabularies than in the past (Goldhor,85) are making this form of input device increasingly attractive to industry.

Commercial software houses have a massive investment in existing software packages and may well be considering adapting their software to incorporate voice input into the 'user interface' (Kidd,82) in addition to their current modes of input. For the adaptation to be viable:

1 The user interface must be at least as good as the original.

2 The command format and information displayed on the screen must not differ radically from the original.

3 The cost of such adaptation must be reasonable, i.e. a small percentage of the original package implementation costs.

If the adaptation fails to meet one or more of the above criteria a total redesign and rewrite of the package could well be required, making the addition of voice input not commercially viable.

2. The prototype system.

With the assistance of Femview Ltd, the industrial co-operating body of this research program, a prototype system has been developed which incorporates voice input into a commercially available CAD software package. The CAD package, FEMVIEW, is an interactive graphics finite element post-processor which can be used to:

(a) check that finite element models have been generated correctly

and (b) enable the display of the results from many finite element analysis packages, including CADAM & MEDUSA (Femview,82).

The purpose of the research program is to evaluate alternative command input techniques in applications software, using the reactions of FEMVIEW users as a means to assess the suitability of the various techniques.

To enable its use on many different computer systems and graphics displays, FEMVIEW uses keyboard entry of commands which follows a strict command syntax. The user is prompted for command input by the display of a menu and a message (Bramer,83 & Bramer,84). Commands from the menu can then be entered singly, or as a command sequence, (i.e. command words from a sequence of menus separated by spaces), the command line terminator being the Carriage Return key. To simplify user interaction and enable experienced users to enter command sequences very quickly some specific techniques are available:-

1 The entry of command words using a minimum or shorthand typing facility enables the command processor to uniquely identify a single command from the menu. For example, to select STOP from the menu:-

STOP
CATALOGUE
CATEGORISE

only the character S need be entered. However, to select CATALOGUE would require, for example, CL to differentiate it from CATEGORISE (e.g. CE).

2 Once a command sequence has been entered, default paths through the command structure are set up enabling the user to enter only the command words that change in successive sequences. For example, the user enters the command (command words are shown in full):

DISPLAY NODES FROM 50 TO 100

Bramer B, & Rickett J D, 1984, A Report on the visit to Lloyds Registar of Shipping & Their Voice Recognition Systems. Leicester Polytechnic, School of Maths, Stats and Computing.

Brundage, 1985, Consumer Applications of Speech Recognition. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Burger, R M, 1984, The Impact of ICs on Computer Technology., IEEE Computer, Vol. 17, No. 10, October, pp. 88-95.

Caratech, 1984, Keytronic : Speech Recognition in a keyboard Peripheral. Caratech Electronics Ltd., Alton, Herts.

Cooke, N, 1984, The Flight Testing and System Integration of Direct Voice Input Devices in the Flight Deck. Royal Aircraft Est. Bedford.

Femview, 1982, FEMVIEW User Manual. Femview Ltd., 4 West Cottages, Stretton Lane, Houghton-on-the-Hill, Leicester.

Goldhor, 1985, A Large Vocabulary Highly Programable Speech Recognition System. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Johnson, S, Connolly, J and Edmonds, E A, 1984, Spectrogram Analysis : A Knowledge Based Approach to Automatic Speech Recognition. Dept. of Computer Studies, Loughborough University.

Kidd, A L, 1982, Man-Machine Dialogue Design. British Telecom Research Labs. Ipswich Pub.

Kode, 1982, Voice Products. Kode Ltd., Calne, Wiltshire.

Logica, 1984, LOGOS Direct Input in the Cockpit. Logica Space and Defence Systems, Cobham, Surrey.

Laporte, C, 1985, A Description of the architecture of a voice I/O Evaluation system and its application in military aircraft. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Marconi, 1984, Speech Recogniser SR128XX. Marconi Space and Defence Systems, Stanmore, Middlesex.

Marsh, G, 1984, Don't Just Sit There - Talk to Me. Defense Helicopter World (June-August).

Rickett, J, 1986, An Investigation into the 'User Acceptability' of Human Computer Interfaces in CAD Systems. Leicester Polytechnic, School of Maths, Stats and Computing.

Rodman, R, 1985, Speech Recogniser Performance with Dysarthric Speakers : A Comparison of two training Procedures. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Uenohara M, 1985, Speech Products and Their Applications in Japan. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Viglione, 1985, Integrating Voice Into Personal Computers. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

VOTAN, 1984, Votan VPC 2000 Users Manual. Copyright Voice Systems International, Cambridge.

to display on the screen a sequence of nodes from the model. To change the upper end of the node display the following could be entered:

TO 200

In this case the sequence 'DISPLAY NODES FROM 50' would be assumed. Care must be taken to ensure the command entered does not appear in an earlier menu in the default sequence.

Facilities which could confuse a new user, such as the default searching technique of 2 above, are normally switched off. Experienced users can switch these on to improve interaction.

In the prototype evaluation system the FEMVIEW command processor, which was implemented on an Apollo Domain DN300 workstation, was upgraded to accept commands not only from the keyboard, but also:

1 by positioning a cursor over the required command word in the menu (moving a tablet pen, mouse or touch pad) and hitting the button to select the command;

2 by typing an integer number corresponding to the position of the required command word in the menu e.g. to select CATALOGUE in the above menu the number 2 would be entered (if the menu included integer or real number input this facility was automatically disabled);

3 by accepting character command strings, identical to those from the keyboard, from an RS232C serial line.

The latter enabled a voice input device to accept spoken commands, convert these to the equivalent character string (corresponding to a keyboard entry) and then transmit these over the serial line to then be processed by the command processor. The voice input system used was a VOTAN VPC 2000 IBM PC compatible bus plug-in card attached to an Olivetti M24 microcomputer. A program in the Olivetti controlled the VOTAN card and handled the RS232 serial link. This mechanism was selected as the simplest way to attach a voice input system to the Apollo Domain DN300. The advent of the Apollo Domain DN3000 with its integral IBM PC/AT compatible bus could enable a suitable plug-in card to be connected directly. In addition, the modified command processor included a built-in monitor giving statistical feedback on objective measures of users performance during an interactive session. This will be used in the research program as part of the evaluation process (Rickett,86).

3. Selection criteria for a voice recognition system.

When examining commercially available voice recognisers, selection was based on the following criteria which were felt necessary to fulfil the overall objective of the research program :-

1 continuous (or connected) speech recognition;

2 good performance/price ratio;

3 software flexibility (i.e. the programmer must have software control over the active set of voice templates);

4 a total vocabulary size sufficient to hold the full range of commands used within FEMVIEW (i.e. approximately 200 words);

5 an ACTIVE vocabulary size (i.e. the number of word choices available at any one time) greater than the maximum number of options available at any point in the FEMVIEW dialogue (i.e. approximately 20 words);

6 a good interface between the hardware and software.

4. Voice systems evaluated for the prototype system.

The commercially available voice recognisers examined in detail for this prototype system are listed below with a brief overview of each. Other systems were also examined but were rejected either because they were not continuous recognisers or they were not available within the U.K.. All costs shown are based on 1984 figures.

Votan VTR 6000. This is a general purpose voice terminal that connects to any computer supporting an RS232 interface with XON/XOFF protocol. It provides continuous speaker dependant voice recognition. It can be configured either as the primary I/O device (replacing the keyboard) or incorporated into an existing computer configuration. Voice key firmware provides a method by which existing programs can be driven by verbal commands without the need for software modifications. It has a maximum vocabulary size of approximately 200 words, with up to 75 active at one time. (cost £4500).

Votan VPC 2000. This device provides similar capabilities to the VTR 6000 but is an IBM PC compatible bus plug-in card. (Cost £3100).

Marconi SR-128XX Speech Recogniser. This system provides connected speaker dependant recognition for up to 240 words. It is a stand-alone system that connects to the host in the same manner as the VTR 6000. Loading of voice templates is achieved via a built-in mini-cassette recorder and recognised words are indicated on a 40 character plasma display mounted on the front panel. (The retail price of this system was approximately £10000 although an educational discount was discussed).

Logica's LOGOS I, II and III. LOGOS provides continuous speaker dependant recognition with a user-programmable word syntax. The system is controlled by a VDU or host computer. LOGOS offers a maximum vocabulary of between 120 and 600 words, with between 20 and 240 words active at any one time. Since each system is custom made for the buyer the system is very expensive. (LOGOS I costs between £18000 and £50000 and LOGOS II and III were approximately £10000 each).

Having evaluated each of these systems based on the above selection criteria, the Votan VPC 2000 plug-in card was selected to be fitted into an Olivetti M24 personal computer. Figure 1 shows the system configuration.

5. General problems with voice input.

When looking at voice recognition systems it is extremely easy to believe that it will enhance system performance and improve its usability. Many of the demonstrations presented are very simple applications with small, specially selected template sets. This section summarises some of the problems encountered when attempting to use voice input within the FEMVIEW system.

5.1. Suitability of command syntax.

The command syntax of existing applications is developed for keyboard or similar 'unnatural' input devices. Therefore the command structure is not designed in a form compatible with spoken input. For example, a command in FEMVIEW to rotate the displayed model could be:-

EYE ROTATE LEFT 90 <return>

When speaking this command in its present form it sounds disjointed. The same command based on spoken input may be :-

ROTATE the model LEFT through 90 degrees.

Obviously, this is not appropiate for keyboard entry since it introduces redundant typing (which would lead inevitably to regular users becoming frustrated with the system). Since keyboard based data entry and spoken input are in direct conflict here, this problem has no easy solution. The object would be to develop commands that are syntactically and grammatically correct and yet do not introduce redundancy into the dialogue. To achieve this may require extensive dialogue modification and much time and effort.

5.2. Distinguishing between linguistically similar words.

Existing recognition systems have extreme difficulty in distinguishing words that are linguistically similar. Therefore, all words in an active set must sound different to avoid this system limitation. This will often require altering the vocabulary and sometimes the command syntax of the application.

5.3. Large Vocabularies.

Most applications software makes use of a large total vocabulary. For example, FEMVIEW uses 197 different words plus integer and real number input, filename input, and special control characters and commands. For the system to be totally keyboard independant it is estimated that around 300 word templates are required with over 100 different recognisable sets (i.e. command menus). It was found that although the manufacturers of the selected voice recognition system claimed that the total vocabulary was only limited by the computer's memory capacity, in practice, there was an upper limit of around 200 words (depending on the number of sets used). The result of this has been that some commands must still be entered using the keyboard.

To train the system with such a large number of words is extremely time consuming. This was found to be a major factor in discouraging people from using the system. However, because of the nature of the FEMVIEW command syntax a method of operation is available in the prototype system that avoids prior training of the system (see section 6.2.).

5.4. Software modification for the adaption to voice.

Manufacturers of voice recognition systems often tempt would-be buyers by offering keyboard emulation software which makes voiced commands appear to the host to be sent from the keyboard, thus 'removing the need for application software modifications'. At first this may appear to be a very useful feature. However, if the end-product is to become commercially viable it must be suitable for the naive user. Keyboard emulation is fine until the user decides to use an alternative form of input or the recognition is incorrect and erroneous data is sent to the application software. At this point the commands that the application is expecting are likely to differ from the recognisable words within the currently active set of voice templates. Once the synchronisation between the command processor and voice recogniser is lost recovery can become very difficult, if not impossible. It has been found that for all but the simplest applications, programs will require source code modifications to enable them to use voice recognition devices effectively.

5.5. Naive user misinterpretation.

To naive users, the prototype system's ability to recognise voices (without prior training) was, on occasion, incorrectly associated with the ability to perform natural language processing and reasoning. Therefore, it is felt that voice recognition devices are not the solution to encouraging 'technology-shy' employees to use computer systems as advocated by many voice recognition manufacturers, even though it is a more natural form of communication.

5.6. Ergonomic considerations for the use of voice systems.

When using spoken input commands it is important to consider the working environment. Clearly, it is not suitable to use a voice recogniser in a location where other people are trying to work, since it will be both distracting and annoying to them. Most recognisers are susceptible to background noise and better recognition accuracy can be obtained by minimising this. If this isn't possible then training should be done in the environment where the voice system is to be finally used and using a sound proof microphone such as the type used by commentators.

6. Problems using voice input with the prototype system.

Using the prototype system, commands issued to the command processor of FEMVIEW could be:-

1 typed from the keyboard;

2 selected from the menu list using a mouse driven cursor;

3 spoken (with the appropiate command string sent to the command processor via the serial line);

4 selected by typing or speaking the integer number corresponding to the position of the command in the menu.

The voice input system could be used in parallel with the others, as the user required. The following are some of the specific problems that were encountered when adapting our prototype system to incorporate voice input.

6.1. Maintaining synchronisation

When a command is spoken, the VOTAN system examines the vocabulary in the active template for a match. It is very important that the VOTAN software maintains synchronisation with the current menu displayed by the command processor on the Apollo. Initially, both the command processor and the voice software start at the root of the menu structure with the initial menu displayed on the screen. As a valid command is spoken the equivalent characters are sent to the command processor and the set of active templates in the voice recogniser are altered to reflect the vocabulary of the next menu in the sequence. The selection of a command from the next menu can then be made. A problem can arise, for example, if the user types commands at the keyboard and then uses the voice input system. The voice input software can become out of step with the current command processor state. To ensure that synchronisation is maintained, the voice input software periodically sends an interrupt over the serial line to the Apollo for information on the present state of the dialogue. The command processor replies with the current menu number and the voice software sets up the Votan system with the set of active words corresponding to that menu (see figure 2).

6.2. Training the system

In general, the voice system has to be trained to accept the spoken commands of a particular user. This can mean running training software that goes through the complete vocabulary of the commands with the user speaking each word in turn and then checking it. The resultant information is then saved onto disk and loaded into the voice system before the application program is started. It has been found from experience that if other users then use the package, certain words can often be understood, but in general each user would wish to train the system for their own voice.

The vocabulary of a program such as FEMVIEW can contain 200 words making training a very long process. Problems can occur due to day to day variations in the user's voice, for example, if they have a cold.

Often the number of words that a user would use in a day can be a small percentage of the complete vocabulary of a complex program and because at any point in the FEMVIEW dialogue the number of word options is small, by reducing the acceptance threshold for selecting a word the voice system can appear to be speaker independant for a large percentage of the words. To take advantage of this, the voice software was amended to allow the user to train the system interactively while the application program is running. The voice information of the user, or even of another user, would be loaded from disk and the application program started. If problems occurred during the interaction, with the voice system being unable to accept a certain word, it would then be trained for that word, and the user would continue to use the system. At the end of the session the new voice information could be stored to disk. This process removes the need for any prior training of the voice templates.

6.3. Command confusion.

If a system was being designed for voice input, care would be taken to ensure that the words in a menu could be easily differentiated by the voice system. When adapting an existing program to use voice input there is likely to be several menus where the voice system has difficultly differentiating between certain words within that menu. For example, in FEMVIEW a menu that caused such problems was:-

G-STRESS
P-STRESS
D-STRESS
E-STRAIN

SYSTEM FLOW

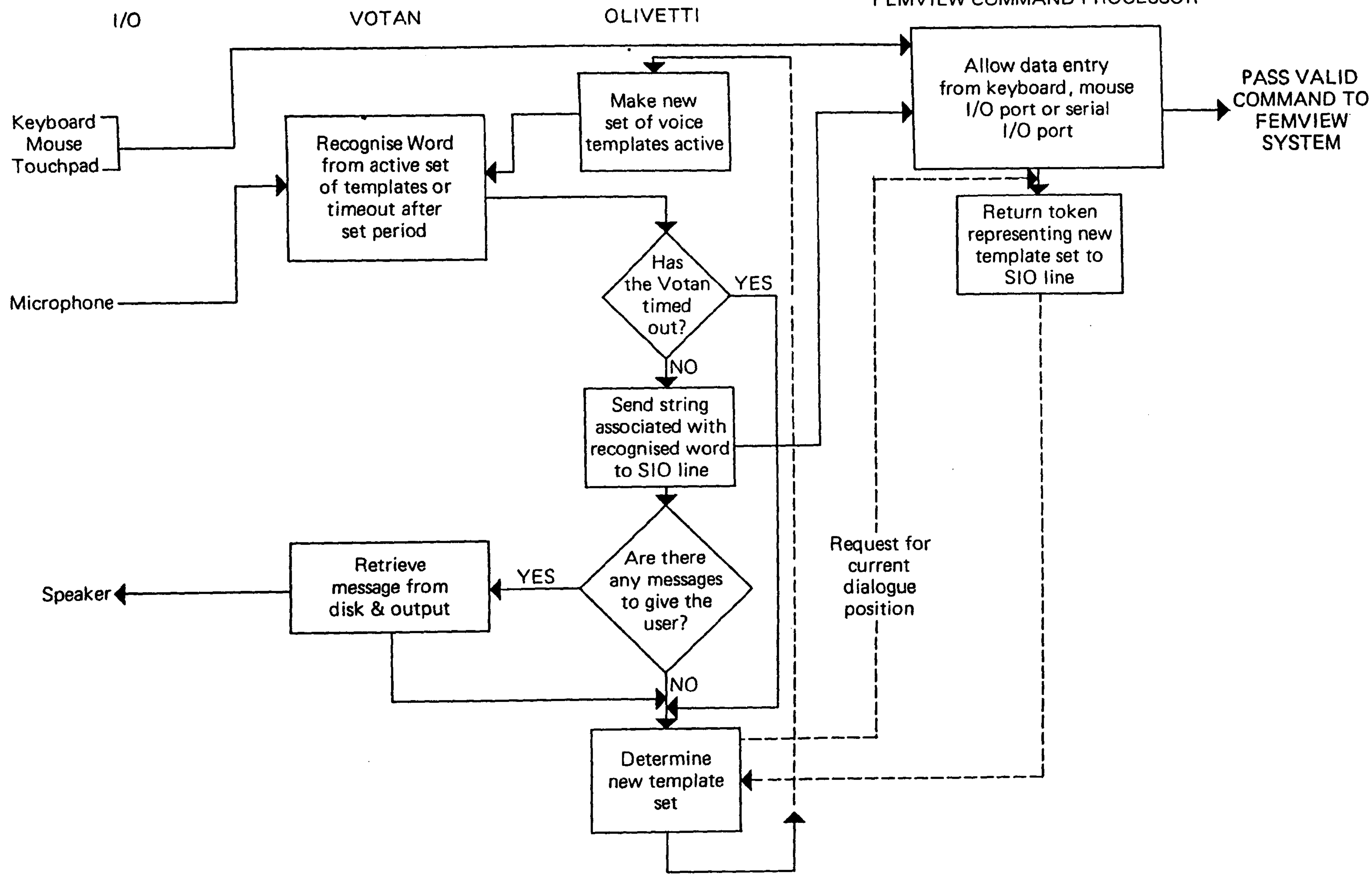


Figure 2

T-STRAIN

(These options specify which attribute of analysis results is required for elementwise result assessment. See Femview,82)

One of the criteria for the viable adaptation of a package to use voice is that the command format and information displayed on the screen must not differ radically from the original. It is therefore not possible to change the words of the menu as this would change the format of the keyboard interface. One of the extensions to the command processor was to enable selection by typing or speaking the integer number corresponding to the position of the command in the menu. In situations such as the above, when the voice input system was having problems in differentiating between the words of the menu, it was possible for the user to speak the number corresponding to its position.

6.4. Continuous speech.

Although the VOTAN VPC 2000 voice card is a continuous speech recogniser, it was discovered that when an active set of templates was replaced for a new set, the microphone was switched off until the new set was loaded. Therefore, the commands had to be entered as a string of discrete words as opposed to a continuous sentence, thus invalidating one of the selection criteria. This was not identified during our initial evaluation since the system was continuous 'within' each set and subsets of a set, but not 'between' different sets.

In practice, users of the voice recognition system found that a higher degree of recognition accuracy can be obtained by developing their own voices to match the recogniser. For example, even with the most advanced continuous speech recognisers, users of the system must have a definite pause between the end of one word and the start of another (unlike normal conversations between humans where the end of one word often follows into the start of the next). Users must also remember to avoid context dependant inflection in their voices since this too, will affect the recognition accuracy.

6.5. Software limitations.

With our selected recognition system each word template is described within a program by a unique template name, an Input/Output buffer string, a program string and a training prompt string. For example, a typical template used in FEMVIEW is:-

```
1.  £def   template    EYE
2.  £text  EYE comm     = "EYE"<10><13>
3.  £text   EYE  response= "EYE    001 003   056 078   088 001   "
4.  £text   EYE  prompt  = "say EYE"
```

Line 1 declares the template name EYE. Line 2 indicates the characters to be sent down the serial line on recognition (i.e. the word EYE followed by line feed [ASCII 10] and carriage return [ASCII 13]). The numbers within the response string on line 3 represent source menus that contain the word EYE and destination menus after recognition. In this case the word EYE is found in menus 1, 56 and 88 and control after recognition of this word goes to menu 3, 78 or 1 depending on the respective source menu. Line 4 is used only for training the word templates.

With the prototype system, the voice program is generated automatically from the FEMVIEW vocabulary and command syntax which is held in a data file. This avoids human errors and allows easy program update should the vocabulary or syntax be altered. One problem encountered is that template names must be unique within the first eight characters and that the template name must not be any of the reserved program words. This can cause extreme difficulty to the software generating the voice program.

7. Advantages of voice input.

Although in the above sections many problems have been highlighted which may discourage a would-be buyer from considering voice recognition, there are still many advantages over other forms of input device.

7.1. Suitability to 'Non-Critical' busy-hands applications.

For 'Hands-busy' applications e.g. piloting an aircraft or many engineering jobs (see Marsh,84, Logica,84, Cooke,84, Anderson et al,85 and Bramer & Rickett,84), voice recognition offers an alternative medium for communicating with the computer. However, it must be emphasized that the applications using voice control MUST be non-critical. It is acceptable to use voiced commands to operate the radio in an aircraft but obviously it is not a good idea to operate the aircraft controls by voice. In engineering applications voice control can be used when either extra time or extra personnel are required to transfer information from its source to the computer, e.g. one person reading data from a technical diagram and another typing it into a computer.

7.2. Novelty value of voice recognisers.

The prototype system, in its present form, is still limited in its capabilities. It has, however, already attracted considerable attention, leading to magazine articles, several internal demonstrations and a showing at a recent computer software exhibition. This 'novelty' value to a commercial organisation is of considerable benefit for sales and promotion of their products. It could therefore be argued that the cost and effort involved in setting up a voice operated system could be justified on these grounds alone.

7.3. Frees users from the keyboard.

One limitation of most current input devices is that they require the user to sit in close proximity to the screen to operate the system. With voice recognition the only physical device the user needs is the microphone, thereby giving the user much more freedom to move around. This has proved useful for demonstration purposes, where a demonstrator can stand well clear of the display, speak directly to clients and issue spoken commands to the software package.

7.4. User controls the speed of operation.

Some devices such as light pens, mice, etc., require selections to be made from menu lists or icons displayed on the screen. Many novice users find this form of input very useful but as experience is gained it can become frustrating since the user's speed is dictated by the speed at which new menus or icons are displayed. In our prototype system both keyboard entry and voice entry allows experienced users to enter commands in advance of the screen display giving the user control over his own working pace.

7.5. Provides a preferable user interface for casual users.

Users of FEMVIEW can be broadly split into two categories: frequent users and casual users. For the frequent user the poor recognition rates for spoken commands make it an inpractical mode of input. However, for the casual (or infrequent) user, the error rate may be acceptable and the user may find this interface device preferable to the alternatives.

8. Summary.

Voice recognition systems are becoming more and more widely available. As a result, commercial software companies are being forced to consider recognition systems as an alternative and/or additional mode of input device. Currently there is little practical documentation on the state of today's voice technologies.

From experience in developing a prototype system that includes voice input as an integral part of the user interface, this paper has described:-

a/ general problems that can arise in adapting existing software to incorporate voice input;

b/ the advantages of voice input;

c/ selection criteria for choosing between different commercially available devices;

d/ specific problems associated with our selected voice recogniser.

Although much investment has been put into the manufacture of speech recognisers, there remains very little evaluation into the suitability of such devices in existing commercial software. It is towards this end that the ongoing research of the authors is directed. Much work still remains and it would be unwise at this stage to speculate about the long term commercial viability of speech recognition in existing software applications.

References.

Anderson T, Moore T & McKinley R, 1985, Issues in the Development and Use of a Speech Recognition Data Base for Military Cockpit Environments. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Anmed, W, 1985, Computer Recognition of Cerebral Palsy Speech. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Bramer, B, 1983, Fortran Based Machine Independant User Input Processor. Proceedings from "The User Interface : The Ergonomics of Interactive Computing" 14-16 Sept. Leicester Polytechnic, School of Maths, Stats and Computing.

Bramer, B, 1984, Fortran Based Command Processor User Manual. Copyright Leicester Polytechnic, School of Maths, Stats and Computing.

Bramer B, & Rickett J D, 1984, A Report on the visit to Lloyds Registar of Shipping & Their Voice Recognition Systems. Leicester Polytechnic, School of Maths, Stats and Computing.

Brundage, 1985, Consumer Applications of Speech Recognition. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Burger, R M, 1984, The Impact of ICs on Computer Technology., IEEE Computer, Vol. 17, No. 10, October, pp. 88-95.

Caratech, 1984, Keytronic : Speech Recognition in a keyboard Peripheral. Caratech Electronics Ltd., Alton, Herts.

Cooke, N, 1984, The Flight Testing and System Integration of Direct Voice Input Devices in the Flight Deck. Royal Aircraft Est. Bedford.

Femview, 1982, FEMVIEW User Manual. Femview Ltd., 4 West Cottages, Stretton Lane, Houghton-on-the-Hill, Leicester.

Goldhor, 1985, A Large Vocabulary Highly Programable Speech Recognition System. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Johnson, S, Connolly, J and Edmonds, E A, 1984, Spectrogram Analysis : A Knowledge Based Approach to Automatic Speech Recognition. Dept. of Computer Studies, Loughborough University.

Kidd, A L, 1982, Man-Machine Dialogue Design. British Telecom Research Labs. Ipswich Pub.

Kode, 1982, Voice Products. Kode Ltd., Calne, Wiltshire.

Logica, 1984, LOGOS Direct Input in the Cockpit. Logica Space and Defence Systems, Cobham, Surrey.

Laporte, C, 1985, A Description of the architecture of a voice I/O Evaluation system and its application in military aircraft. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Marconi, 1984, Speech Recogniser SR128XX. Marconi Space and Defence Systems, Stanmore, Middlesex.

Marsh, G, 1984, Don't Just Sit There - Talk to Me. Defense Helicopter World (June-August).

Rickett, J, 1986, An Investigation into the 'User Acceptability' of Human Computer Interfaces in CAD Systems. Leicester Polytechnic, School of Maths, Stats and Computing.

Rodman, R, 1985, Speech Recogniser Performance with Dysarthric Speakers : A Comparison of two training Procedures. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Uenohara M, 1985, Speech Products and Their Applications in Japan. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Viglione, 1985, Integrating Voice Into Personal Computers. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

VOTAN, 1984, Votan VPC 2000 Users Manual. Copyright Voice Systems International, Cambridge.

Yoshimura, M, 1985, Application of Voice I/O to Home Sewing Machine. Proceedings of Speech Technology '85. Media Dimension Inc, New York.

Voice Recognition: A Question of Viability in Commercial Software

By

John Desmond Rickett.

To be presented at the IEE colloquium on
"Emerging Office Technologies"
held at the Savoy Place, London.

School of Mathematics, Computing and Statistics
Leicester Polytechnic.

August 1987.

A4

FEMVIEW users questionnaire.

FEMVIEW USERS QUESTIONNAIRE

In an attempt to identify users opinions about particular aspects of the FEMVIEW Finite element post-processor system, the following questionnaire has been produced.

Its main concern is the User Interface of the FEMVIEW system (i.e. the interaction between you, the user, and the displays and controls of the system).

Results from this questionnaire will be analysed by a student at Leicester Polytechnic who is evaluating Human-Computer Interfaces as part of his Ph.D. thesis.

Features of this research relating to the FEMVIEW system include: the investigation of various alternative forms of input device based on new technology, for example, the use of cursor control devices and voice controlled data entry. Also display designs and useability of the system.

Your co-operation in completing this questionnaire is requested. It should only take a few minutes. Any information received as a result of the following questions will be treated in strict confidence.

Note : You may not be in a position to answer the questions in section 1 about your company use. If so, don't worry, just start at section 2.

1. Your COMPANY Use.

1.1. When did your company first start to use FEMVIEW ?

______(month) ______(year)

1.2. How many people in your company used FEMVIEW when it was first available to you ?

______

How many of your people use it now ?

______

1.3. Has the regular use of FEMVIEW in your company :-

Increased
Remained unchanged
Decreased

(tick as applicable)

1.4. If the use of FEMVIEW has changed, please give the reasons.

_ _
_ _
_ _
_ _

1.5. How many hours per week, in total, is FEMVIEW currently used in your company ?

_ _ _ (hours/week)

1.6. How would you describe the users of this system ?
(please indicate the number of people within each category)

Engineers	_ _
Systems Support	_ _
Computer Programmers	_ _
Management	_ _
Computer Operators	_ _
Students	_ _
Others, Please state	_ _
_ _ _ _ _ _ _	_ _ _
_ _	_ _

2.Your INDIVIDUAL Use.

2.1. Under which of the above categories would you place yourself?

_ _ _ _ _

2.2. How would you describe yourself in your use of FEMVIEW ?

i) Experienced / Novice
(delete as applicable).

ii) Regular / Occasional
(delete as applicable).

2.3. Has your use of FEMVIEW :-

Increased
Remained unchanged
Decreased
(tick as applicable)

2.4. If your use of FEMVIEW has changed, please give the reasons.

_ _

2.5. How many hours per week do you use FEMVIEW ?

__ ______(hours/week)

3. The COMMANDS

3.1. Command Syntax and Semantics.

3.1.1. 'EYE', 'VIEW' and 'PRESENT' are examples of FEMVIEW commands. How well do you remember the meaning of these, and other commands when using FEMVIEW ?

very easy to remember
easy to remember
adequate
difficult to remember
extremely difficult to remember
(tick as applicable)

3.1.2. How easily can you remember the syntax of the FEMVIEW commands ?

Do you :-

- Press the return key after most words and use the menu prompts to determine the command syntax

- Press the return key occasionally when you are unsure of a commands syntax and/or you wish to follow the menu prompts

- Very rarely press the return key except at the end of the full command

(tick as applicable)

3.1.3. Some operations involve a sequence of commands, for example :-

> RESULTS LOADCASE CASE1
> RESULTS NODAL DISPLACE ALL
> PRESENT CONTOUR LEVELS 9

Do you find sequences of commands such as these, and others :-

very easy to remember
easy to remember
adequate
difficult to remember
extremely difficult to remember

(tick as applicable)

3.1.4. Below is an area for any suggestions, likes and dislikes of the FEMVIEW commands in the present system.

3.2. Help Facilities

3.2.1. Help facilities to FEMVIEW are provided in two forms :-

i) a help option built into the system providing guidance on the syntax and semantics of each command in the menu, and

ii) the user manual

How informative do you find these help facilities?

Built-in Help	**Manual**
Very informative	Very informative
good	good
Adequate	Adequate
poor	poor
very poor	very poor

(tick as applicable)

3.2.2. If you have any comments or suggestions relating to these help facilities please place them below.

_ _
_ _
_ _
_ _

3.3. Keyword Entry

3.3.1. Keywords can be entered in full or in shortened form, e.g. 'DATA DISPLAY' or 'D D'.

Do you make use of this minimal tvping feature?

Always
Usually
Only occasionally
Never

(tick as applicable)

3.3.2. In FEMVIEW, keywords are entered via the keyboard. Would you like to see additional forms of inout implemented within this software, such as :-

cursor pointing devices :-
- mouse
- light pen
- touch pad
- tablet and pen,

icon based menus,
voice recognition,
Other, please specify _ _ _ _ _ _ _ _ _ _

(tick as applicable)

3.3.3. If your choice of additional forms of input device is based on seeing or using other systems incorporating these devices. Please state (if possible) the name of the systems and computers on which they were used.

_ _
_ _
_ _
_ _

3.3.4. What advantages do you see from the use of additional forms of input device ?

_ _

3.4. Menu Prompts

The keyword options for commands are available in menu form. These are called the 'menu prompts'.

3.4.1. Do you rely on the menu prompts ?

Always
Usually
Only occasionally
Never

(tick as applicable)

3.4.2. Do you find the menu prompts informative ?

YES / NO

(delete as applicable)

3.4.3. Could the menu prompting be made to be more informative. If so, how ?

_ _

3.5. Error Messages

3.5.1. FEMVIEW will display an error message, where for instance, the user enters a command out of sequence or mis-spells a command.

In general, do you find the FEMVIEW error messages ?

Verv informative
good
Adequate
poor
very poor

(tick as applicable)

4. Additional Information.

4.1. If you have any comments, observations or suggestions not covered elsewhere in this questionnaire, please enter them below.

_ _
_ _
_ _
_ _
_ _
_ _

5. PRACTICAL EVALUATION.

5.1. In the next stage of the research Project, the results from this questionnaire will be used to improve certain aspects of the FEMVIEW User Interface.

Are you willing to participate in a practical evaluation of these improvements ?

YES / NO

(delete as applicable)

5.2. If your answer to the above question is YES or you have any queries concerning this research, please give the following information :-

Name _ _ _ _ _ _ _ _ _ _ _ _

Company Name _ _ _ _ _ _ _ _ _ _ _ _

Address _ _ _ _ _ _ _ _ _ _ _ _
_ _ _ _ _ _ _ _ _ _ _ _
_ _ _ _ _ _ _ _ _ _ _ _

Telephone No. _ _ _ _ _ _ _ _

Thank-you for your time and co-operation in completing this questionnaire.

Yours faithfullv,

John Rickett,
Research Student,
School of Maths, Stats and Computing,
Leicester Polytechnic,
P.O. Box 143,
Leicester. LE1 9BH.

Tel. (0533) 551551.

A5

Summary of results obtained from multi–user survey.

Summary of results obtained from multi-user survey

	Ricardo		AWRE		Rolls Royce			
1.1	1978	-	-	-	1981	-	-	June 1981
1.2	3/8	-/4	4/10	-	4/8	-	-	10-15/ 15-30
1.3	↑ FE ↑ workload →	↑	↓ Other S/W	-	↑Staff ↑	-	-	↑Number
1.5	50	10-20	2	-	30	-	-	20-40
1.6	8xEngrs	Engrs	10xEngrs 1xEngAss	-	Engrs ?Managers	-	-	15-50xEng 1xSys Sup
2.1	Engrs	Eng	Eng Ass	Eng	Eng	?	Eng	Support
2.2	Exp Reg	Exp Reg	Exp Occ	Exp Occ	Exp Reg	Exp Reg	Exp Occ	Exp Occ
2.3	↑Mgmt role ↓	↑	0	↓ Change in duties	o	↑	0	↓
2.5	3	2-5	15hrs 8 wks/yr (x2hrs/wk)	0-2	4	2	2	1
3.1.1	V easy	Easy	V easy	V easy	Easy	Easy	Adeq	Adeq
3.1.2	V rarely	V rare	V rare	V rare	V rare	Occ/Vrare	V rare	Most words

3.13	V Easy	Easy	Easy	Easy	Easy	Easy	Adeq	Easy
3.21 (a)	Adeq	Adeq	Good	Good	Good	Not used	Adeq	Good
3.21 (b)	Good	Good	Good	Good	Good	Good	Good	Adeq
3.31	Always	Always	Always	Always	Always	Always	Always	Adeq
3.32	None	None	Light pen	Mouse L pen Voice	Any	Tables & pen-CAD appln	?	-
3.41	Only Occ	Only Occ	Only Occ	Only occ	Only Occ	Only Occ	Usually	Usually
3.42	?	Yes	Yes	Yes	Yes	Yes	Yes	Yes
3.56	Good	Good	Adeq	Good	Adeq	Adeq	Adeq	Adeq
5.1	Yes	Yes (on site only)	No	No	No	No	Yes	Yes

	Babcock Power		MOD RARDE		Unknown	LCMS	Lucas Aerospace	Lucas Gisling
1.1	Feb 1982		-	March1983	1981	June 1984	Sept 1984	1985
1.2	5/40	6/6	-	2/5	3/3	1/6	2/2	1/2
1.3	↑	0	-	↑Use PAFEC ↑	0	↑ FE use ↑	↑ Femview in design process ↑	↑ New site now using Femview
1.5	30-90	15	-	-	4	10	6	< 3
1.6	40 x Eng 1 x sys sup 2 x prog 1 x	5 x Eng	-	1 x Eng	4 x Engrs	10xEng	6.Eng	2 x Eng
2.1	Support	Engineer	Engineer	Eng	Eng	Eng	Eng	Eng
2.2	Exp occ	Exp reg	Exp reg	Novice occ	Exp occ	Exp reg	Exp reg	Exp reg
2.3	↑ Verification testing ↑	0	↑ ↑ FE jobs + ↑ terminals	0	↑Responsibilities ↓	0	As above ↑	↑ initially problems with Femgase
2.5	15	5	Up to 10	Up to 10	2hrs/mth (0.5hrs/wk)	2	4	< 2
3.11	V Easy	Easy	V easy	Easy	Easy	V easy	Easy	Easy
3.12	Occass	V rare	V rare	Occass	Occass	V rare	V rare	Occass

3.13	V easy	Easy	V easy	Easy	Difficult	Easy	Easy	Easy
3.21 (a)	V informative	Adeq	Adeq	Rarely used	Good	Adeq	Good	Poor
3.21 (b)	V informative	Poor	V informative	Good	Good	Good	Adeq	Adeq
3.31	Usually	Always	Always	Always	Usually	Always	Always	Always
3.32	Input file for usual seq like FV	None	None	-	L pen speed	Mouse L pen Tablet	None	Mouse Tablet & pen Icons
3.41	Only occ	Never	Only occ	Only occ	Only occ	Never	Only occ	Only occ
3.42	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
3.51	Good	Adeq	Poor	Adeq	Adeq	Adeq	Good	Poor

↑ - Increased
0 - Remained unchanged
↓ - Decreased

	A E D							
1.1	Feb1985	-	-	-	-	-	-	-
1.2	/10	-	-	-	-	-	-	-
1.3	↑ ↑ applns.	-	-	-	-	-	-	-
1.5	35	-	-	-	-	-	-	-
1.6	8xEng 1xGen 1xStud	-	-	-	-	-	-	-
2.1	Eng	Eng	Eng	Eng	Eng	Eng	Eng	Eng
2.2	Exp Reg	Nov Occ	Regular	Nov Occ	Nov Occ	Exp Reg	Exp Reg	Exp Reg
2.3	↑	Less FE work ↓	0	0	0	↓ Non-linear problem solving	0	0
2.5		Less FE work	1-2	2	0-20hrs	-	4	5
3.11	V easy	Easy	Easy	Adeq	Adeq	Easy	Easy	Easy
3.1.2	Occass	Occass	Occass	After most words	Occass	Occass	Occass	Easy

3.1.3	Easy	Adeq	Easy	Easy	Adeq	Easy	Adeq	Easy
3.2.1 (a)	Adeq	Adeq	Adeq	Adeq	Adeq	-	Adeq	Adeq
3.2.1 (b)	Good	Adeq	Poor	Good	Adeq	-	Adeq	Adeq
3.3.1	Always	Always	Always	Usually	Always	Always	Always	Always
3.3.2	-	-	Mouse	Mouse fewer errors faster output	Mouse MassComp faster control	-	Light pen quicker	Tables + pen speed
3.4.1	Only occ	Usually	Usually	Usually	Only occ	-	Only occ	Only occ
3.4.2	Yes	Yes	Yes	No	No	-	Yes	Yes
3.5.1	Good	Adeq	Adeq	Adeq	Adeq	-	Adeq	Adeq
5.1	Yes	No	No	No	No	-	Yes	No

	Gullick Dobson		Femview		GEC		Leicester Poly	
1.1	1984	1984	June 1983	1978	Feb1984	-	1982	Sept 1984
1.2	3/2	3/2	2/4	1/5	6/1-2	-	-	1/4
1.3	0	0	↑ more analysis runs	↑ No of staff ↑	varies with workload	-	0 demos only	0
1.5	2-4	2	10	20	0-10	-	1	1
1.6	2x Stress Eng	3xEng	2xEng 1xSys Sup 2xProg	2xEng 1xSysSup 2xProg	1xEng 1x FE Specialist	-	2xStudents	2xProg 2xStuds
2.1	StressEng	StressEng	Support	Prog	FE Spec	Eng	lecturer	Stud
2.2	Exp Reg	Exp Reg	Exp Occ	Exp Reg	Exp Reg	Novice Occ	Novice Occ	Novice Reg
2.3	0	0	↑ More analysis	New ↓ version	varies with workload	0	0	↓ Change in work
2.5	2	2	1	5	1½	N/A	2	1
3.11	Difficult	Adeq	Easy	Adeq	V eaxy	Easy	Adeq	Easy
3.12	V rare	Occass	Occass	V rare	V rare	Most Words	Occass	Occass

3.13	Adeq	Adeq	Easy	Easy	V easy	Easy	Adeq	Difficult
3.21 (a)	V Poor	Poor	Poor	Poor	Adeq	Adeq	Adeq	V Poor
3.21 (b)	V Poor	Good	Adeq	Good	Good	Good	Adeq	Poor
3.31	Always	Always	Usually	Always	Always	Usually	Always	Usually
3.32	None	Mouse	Mouse Icons Mac more fun	All cursors Icons less errors	Mouse touchpad	Touch pad	-	Mouse Icon Voice
3.41	Only occ	Usually	-	Only occ	Only occ	Usually	Usually	Usually
3.42	No	No	Yes	Yes	Yes	Yes	Yes	Yes
3.51	Adeq	Adeq	Poor	Poor	Good	Adeq	Adeq	V Poor
5.1	No	Yes	Yes	-	Yes	Yes	Yes	Yes

	Scicon			Knoemer (Norway)		BP research centre		
1.1	-	1981	1981	Dec 1981	1982		-	1982
1.2	/4	4/4	1-2/4-5	4/8	1/8	1/8	4/8	3/8
1.3	0	↑ use of graphics FE analysis	↑ comp. facilities + new designs/y	↑ more FE work each yr.	↑ more people use it	↑		↑ FV now linked to most FE s/w
1.5	20	5-30	15	20	25	28	-	40
1.6	3xEng 1xS.S	3xEng 1xS.S	3xEng 1xS.S	4.5 Eng -	5xEngrs [illegible] 1xMaths		-	-
2.1	Engineers	Eng	-	Eng	Eng	Maths	Eng	Eng
2.2	Exp Reg	Exp Reg	Exp Reg	Exp Reg	Reg	Exp Reg	Reg	Exp Reg
2.3	↑ new uses for s/w	↑ as above + better h/w	↑ reliance on FV	↑ More FE work	0	0	↑	↑ as above + own MicroVAX
2.5	8	6-8 As proj requires	5	5-7	5	4	15	2
3.1.1	Easy	Easy	Easy	Adeq	Adeq	V easy	Adeq	Adeq
3.1.2	V rare	Vrare	Occas	V rare	Occas	V rare	Occas	V rare
3.1.3	Easy	Easy	Easy	Adeq/Diff	Adeq	V easy	Easy	Easy
3.2.1 (a)	Good	Good	Poor	V Poor	Adeq	Good	Good	Poor
3.2.1 (b)	Good	Good	Adeq	Good	Adeq	Good	Good	Adeq
3.3.1	Always	Always	Always	Always	Always	Always	Usually	Always
3.3.2	None adv.only to novice	None	Only if quicker than typing	Mouse, maybe problem with cursor hits	None relevant	-	Tablet & pen	Voice speed & convenience
3.4.1	Only occ	Only occ	Only occ	Only occ	Usually	Never	Only occ	Only occ
3.4.2	Yes	Yes	Yes	Yes	Yes	-	Yes	Yes
3.5.1	Adeq	V informative	Adeq	Adeq	Poor	Poor	Poor	Adeq
5.1	Yes	Yes	Yes	No	No	Yes	Yes	Yes

A6

Suggestions concerning the FEMVIEW software as a result of the questionnaire.

Suggestions concerning the FEMVIEW software as a result of the FEMVIEW USERS QUESTIONNAIRE.

Input features.

Allow the enter of several commands without continuously redrawing the model after each command e.g. HOLD = no redraw; HOLD OFF for redraw.
(R.R. & Ass.)

Option to stop model from being drawn when entering Data Base would be useful.
(R.R.& Ass.)

Output features.

Ability to add comments to plot/screen.
(Lucas Girling)

Display attribute load case, options ,etc currently in use. This would save time in plotting something which you did not want. Also a display of viewing angles chosen would aid repeatability.
(A.E.D.)

Dialogue design.

Suppress menus option.
(L. CMS, Lucas Aerospace)

Menu prompts: add option to expand a command in full (i.e. expand prompt).
(SCICON)

Show menu sublevels.
(Gullick Dobson)

Should be a better relationship between single words & their function. Also less double negatives e.g. OPTION NOMESH OFF.
(Gullick Dobson)

DELETE & FINISH should not be abbreviated to single keys ('D' & 'F'), these can be easily pressed by mistake.
(Gullick Dobson)

Loadcase option from main menu rather than through RESULTS.
(Babcock Power)

Rotate option from main menu.
(Kvaemer)

Give abreviations for STRESS & STRAIN e.g. ST & SN.
(GEC)

Error messages.

Audible warning when command error (with facility to switch it off).
(GEC)

Expand error messages (not beyond 2 lines).
(MOD RARDE (FH))

Interface design.

Cancel the drawing of a model after it has been started
(Gullick Dobson, AWRE)

FEMGEN/FEMVIEW interface consistancy.
(RARDE (FH), R.R.& Ass., MOD RARDE (FH))

For verification purposes would like to see a visualisation mode command which will sequence thru. each model, loadcase, results output & display type automatically thus testing all interfaces.
(Babcock Power Ltd)

Provide help on taking shortcuts.
(CMS)

If only one loadcase, project, &/or model pick it up automatically.
(MOD RARDE (FH))

'Eye Rotate' confusing for novices.
(R.R.& Ass.)

Longer names for models, i.e. 10 - 12 letters.
(Lucas Girling)

List possible 'strings', 'integers', 'reals'.
(FEMVIEW)

Abort a command without breaking from the program if the wrong option or attribute has been chosen.
(A.E.D.)

Application related features.

Add the facility to superimpose graphs
(R.R. & Ass.)

To display surface results the component has to be viewed in VHF before OPTIONS OUTLINE can be selected, this is very slow.
(L. CMS, Lucas Aerospace)

Add PRESENT TABLE NODE 123 <u>ALL</u> as in GRAPH ALL loadcases for a single node.
(R.R. & Ass.)

Difficulties at present erasing line.
(AWRE)

More facilities are needed for PRESENT CONTOURS eg:-
P C VALUE V1 V2 V3 V4 etc. to draw contour values -

P C range V1 TO V2 STEPS V3
(A.E.D.)

Improvements needed for structures with different element types eg. solid elements mixed with beam elements.
(A.E.D.)

The whole mesh has to be drawn before zooming. This is very time consuming.
(A.E.D.)

Help facilities.

Provide a manual Index.
(AWRE)

Add help facilities like FEMGEN.
(GEC)

Help should give explanations not only commands available.
(Kvaemer)

Training.

Include an example Data Base with an input file which steps new users through s/w.
(MOD RARDE (FH))

More explicit information regarding how to use the various options (after a break of several weeks or months.
(A.E.D.)

A7

Key to the on–line monitor.

Key to On-line Monitor.

OPERATOR		ACTION
N	-	Mouse Movement
Z	-	Mouse Stopped
M	-	Mouse Button (Left,Centre,Right)
K	-	Keystroke
B	-	Backspace
R	-	Reject Line
V	-	Voice Entry
?	-	Voicecard Synchronisation
S	-	Enter Command Processor
C	-	Request for Menu Prompt (i.e. Return key pressed before completion of the command string
F	-	Valid Command accepted (exit command processor)

Monitor File Format

OPERATOR	TIME	ACTION string/key/operation
[.]	[000.00]	[...............]

A8

Transcripts from interactions.

Subject 1.

So this is a 300; they are all 300's
FEMVIEW

2 If I can give you your task
2 That's your bench exercise
2 Could I start FV up & put in your monitor name

Yes

[long pause]

2 Would you like to talk your way thru'

Shall I read this 1st.

[reads out quietly]

Log into Fv....

2 If you'd like to do ex 1 & talk your way thru' as you go with your ideas..

Right
Logged into FV
User key is user
Right, now for some reason I'm going to put it in upper-case
Because I'm more comfortable using upper case.
I'm going to select bench which is in lower case as it happen.
I'm going to cont. in upper-case & select the model housin.
Show the model in a broken mesh format
Which on FV version 4, so it's mesh broken
OK
Define the temp. subset model consisting of the following elements in the range between 1 & 50. Display the defined subset with hidden lines removed.
OK
So I'll do set append elements 1 to 50
Set show defined
I say V H F
OK
So it hasn't said otherwise so I'm leaving it in broken mesh
OK
Locate. Right, because its a locate I know it's going to be in dotted line so I'm going to say Mesh Broken Off colon eye locate
Ok so that has given me a solid line view of the subset & a dotted line outline. This is probably the icon as it was. Right.
If it was 3.5 I was carrying around in my head it would be Options Locate.
Select loadcase case1
S L case1
Displace nodal elements in all directions. [laughs]
& present 10 contour levels
So Results nodal displacements all

Displace...So I guess that means mesh Deform all
& present contours to 10 levels
Present contour level 10
Ok
Exit the FV s/w back to the o/s
Which means finish

2 OK. If I just take you back into FV & put in the monitor.

[long pause]

2 Right. If you'd like to continue on

Right
Log back into Fv. So I remember my user key is user.
OK
I notice that I can type minimal typing. I have to type bench in full & I can select HOUSIN with H
Oh right. I'm not reading the instructions.
Index Models
Index Projects
I just wasn't reading it properly
Right
I'm not very good at reading written instructions which doesn't help.
TTBO
Right
Rotate into a plan elevation. The initial offset of all models is a 20 degree rotation in both the x & y directions.
So plan?
I don't know what a plan is here
Prob. in the x, y plain so
EyE rotate to 90
Lets have a look at that.
No I don't call that a plan
Eye rotate to 0. What is a plan? That looks like a plan to me.
OK
Show a mesh construct, in full perspective with hidden lines removed.
OK. So
Full perspective so what are we going to say
Is it EYE perspective colon v H F. Something like that?
Put on perspective & put on a hidden line view
Redisplay the model with the lines in a dashed format & enlarge to fill the whole screen.
OK. This is a little bit confusing.
I could type ahead.
So I could say Mesh broken colon eye frame
Copy this into viewports 1,2,3 & 4.
Drawing viewport use 1
Drawing viewport copy 1 to 2.
Either these are right or not. I'm typing way ahead of them.
Drawing viewport copy 2 to 3
Drawing viewport copy 3 to 4.
Now...

2 What exactly have you done here?

OK Copy this view into viewport 1, 2 3 & 4.

I said use viewport 1, which transfer the contents of viewport 0, which is the whole thing. Made it use viewports & put the information in vp. 1. & it then says copy this into vp. 1 2 3 & 4. So copy from 1 to 2, & it drew in 2. 2 to 3, 3 to 4 & it's now drawn in all 4 viewports
Right
Select all loadcases, & make a nodal displacement in the x-axis. Present the deformed shape with a scaling factor of 50.
OK so using viewport 1. Drawing Viewport use 1
OK. Control goes to viewport 1.
Select all loadcases. I can't believe. I have to say index loads.
Ah. Right.
R L L1
Selects loadcase 1 of which there is only 1

2 What was that there?

I've had a look at the loads index because I was confused by the fact that it said Select all loadcases. There is only 1. It would be very nice if that was a default selection
So there was only 1 selected.
Make nodal displacements in the x-axis. Present the deformed shape with a scaling factor of 50.
Ok so results nodal displacement x.
& Mesh Deform x 50
I didn't have to say results nodal displacement x, I only needed to say mesh deform x 50 because I'm reading instructions line by line. If I read ahead I'd probably do something a bit different.
I think this is measuring a lot to do with how I read a written instruction.

2 There is nothing stopping you reading ahead if you feel more comfortable doing that.

OK. I'm Trying too hard.
Present the deformed shape with a scaling factor of 50.
OK
I can't tell any difference & it hasn't said anything on the monitor & I don't know whether I've done that or not.
I'm just going to have faith that that's what I've done.
Which is a bit silly
In viewports 2 3 & 4 show similar diagrams with nodal displacements in the y-axis, z-axis & all axis respectively.
OK I think I need to say
Drawing Viewport use 2
Then I need to select the results loadcase which is R L L1
& then I can mesh deform y 50
I'm trying to stare hard at this to see whether I can detect any difference since its drawn it.
I don't know the model well enough to know if the deformation's in x, y or z of big. Or at a factor of 50 I going to be able to make sense of it.
OK I just imagine that that has moved.
Yes there's a slight difference between that picture in viewport 2 & the picture in vp. 1.
OK. drawing vp. use 3.
Select the results on loadcase L1.

Now. here when I displace it in z
Its going to be a boring picture because I'm actually looking at it in z so there won't be any different unless they.
Mesh Deform z 50
Except it has perspective on. It might make a difference. I just don't know.
OK. We're going to vp. 4 & we're going to Apply results to the displacement. Once this has drawn what it's drawing.
In vp. 4 under the diagram add the following text.
Plan view of model name. By your name on date
Add the appropriate information
Ok. Save the screen using Drawing Save with the title what's it
OK Drawing vp. use 4
Right I'm asked here to add 2 pieces of information & to save the view & I know it is only going to save 1 of these.
So I have a dilemma.
It already says in the monitor what the model name is so plan view has the axis on it so I'm going to ignore the 1st piece of information & just put the last.
drawing contents text
So I say
I'm going to say
Plan view By . I have to make it short otherwise I won't get it in the viewport.
Derek on. Have a look at my watch. 10th of the 4th.
10 dash 0 4.
dash 87
& I get the chance to move this with the cursor.
Where am I going to put it!
At the bottom
I did what i thought was a cursor hit which was the space bar & I'm now confused so I..

2 There we are...

OK.
Save this screen using draw save with the title view2.
Drawing save.
I'm not saving the screen I'm saving a drawing
Drawing Save.
I'm a bit confused in that it's accessing file. *Never mind.*
Drawing Save .
with the title view2. Tell the observer when you are ready to proceed to ex 3.
oK

2 So let's go on to 3 now

Yes
In fig. 1 you are given 2 models.
Try to reproduce this diagram in exactly the same format as that shown.
Once you are happy with your reproduction save it . *Use the* Drawing save command & give it the name view3.
OK
I've got to say I'm a little bit confused by the wording here. It seems to say save this 'screen'. Using drawing save & I know that I'm only saving, or I believe I'm only saving

individual drawings.

2 Shall we say that it saves the whole screen just to clarify.

Yes
Right

2 Just as a particular screen rather than saving them all to save time.

I'm trying to.
Have a look at this.
For no reason I'm going to start at the top left hand.
So I'm going to put.
Drawing viewport use 1.
& I'm going to index models because I have a vague glimpse that. Ok
Either the photocopy deliberately missed off the model name in the top left.
But it looks like the one in the bottom right. So I guess that's BTLH. I guess it must be that, so I select BTLH
OK the viewport representation of it & it looks like what the picture is here. You can see the long view.
So I just say View hidden fill & I think...
Am I right.
I'm just wandering here exactly where we are going for like Umm, this actually says accessing file & the only way I'd get accessing file & then go on to a new vp. is by saying V H

2 Shall we say that that's not really worth it. Because I think that's not intended.

OK
Drawing viewport use 2
So this looks like the right hand side
So index models
Select BTRH
Do I go for this.
Looks to me like a vp. representation
& it is saved unfortunately
I can go mesh shrink, V H F
It's the picture
Ok
I'm going to do this
Looking at the instructions again I might as well save 1
Once you are happy with the glimpse, save it. Use the drawing save command & give it the name view3.
OK
It makes me wonder if I don't know how the program works & when I say data save whether I save the whole thing.
I'm going to use 1
& I'm going to save this.
drawing save
I hit the keys in the reverse order which I sometimes do & said save.
I'm going to call this VPT1 OK
Drawing vp. use 2
& attempt to save this with drawing save VPT2
Ok

Just to check that.
I'm going to ignore what it says in the top one
Drawing vp. use 3
& then
I'm just going to say data display
Cause I don't know where I am now.
Oh. right. I do know where I am
That's another model there.
A completely different model
So I'm going to say drawing vp. copy 2 to 3.
Can I do better than that colon mesh broken colon results loadcase 11 colon results nodal displacements all colon present contour level 5
Right, I don't know whether there are ways to reduce that it looks like it might
[long pause]
Ok It's taking a while to process that
I'm still confused by the instruction of save this screen using drawing save with the title view2.
& I still think that I'm only saving individual viewports so I'm going to, I can't save them all, I could save them all with the same title.
But I don't think it's going to be very informative so I'm going to ignore the instruction that it has to be view 2 & call this VPT3.
It looks like it
Ok
So I was right in that it looks like it was drawing save VPT3 & I've got to go onto the last one. It looks a little bit like no. 1 So I can say drawing vp. copy 1 to 4.

[pause]

right. Then move onto ex. 4

2 Yes ex 4 & then a little questionnaire afterwards.

Ok
It's hidden line view. Looks like I need to apply 1st of all Ah! R L L1 colon R N D A results displacements all then Mesh outline & I reckon we need colon.
1 2 3 4 5 6 7 8 9 10. Present contour levels 10.
Is that what it is?
Looks a bit hopeful
Ok. Looks very like it. So I'm going to say drawing save VPT4.
Ok. now as far as my interpretation of saving the screen, I believe I can dump the metafile of this so I'm going to say shift dollar. Oh. now I'm surprised by that because in the installation I'm used to that would have dumped a metafile. So I'm going to try data plot. Hardcopy. No I can't do it. I'm going to say data plot because I know that is intended to plot all 4 & I'm going to call those view 2. Ok?

2 Yes

Right.
Ok let's go onto 4.It is nearly right.
It is thought that project test model hanger has not been generated correctly. Examine the model carefully & record

any faults in the mesh on the form provided. Table 4.1. Do not spend more than 5 mins. on this ex. All faults are connectivity problems.
When you are happy that all errors have been detected exit from the FV system back to the o/s of your computer.
Ok.
It hasn't come back yet because it's doing this Fv tedious plot. But when it does.

2 Do you mind if we leave ex 4. Then i can ask you some questions & then we can go to lunch.

I tell you what I can do ex. 4 while we are answering questions Ok?

2 Fair enough, & it won't interfere with you.

No it won't.

2 Ok. Great.

[System had crashed at this point & no further work completed]

Subject 2.

right
is this colour?

2 I'm afraid this is black & white

black & white,right

2 there is your ex.
2 if you'd like to start FV up

right
do we just type end of text now or

2 just type FV will do

fine

2 just type FV now
2 I didn't notice where the cursor was

right
I've just typed Fv

2 Can I just enter the monitor for you

Urh! hum

2 so we... just find the monitor name
2 right if you'd just like to try exercise 1 & talk thru your ideas as you go

right
the user key!

2 What is your user key?

sorry

2 Ah!

for the dbase
is it Hope1?

2 yes Hope1

[from screen]

incorrect user key

2 Ah! that's interesting

there's a security key on the dbase I can't get in without knowing what the sec. key is.

2 Ah! really

Unless it's [types in silence]

2 Ah! that's interesting

Ah!
no that's 3 attempts so it kicks me out

2 If I put the monitor name in again
2 we'll have another go at that
2 i might of put the incorrect monitor name
2 it shouldn't.....
2 I'll just see if [goes to find supervisor].....

[return not found him]

2 Shall we try the database name

what's that?

2 dot dot comma monitor directory slash name

[super. returns]

2 John we have a type your user name here

what's the user name for the dbase?

3 User

[points to place on Benchmark with the instructions]

Ah! sorry
right

2 Ah! sorry
2 I'll put you back in again
2 Ok.
2 Sorry about that

It's my fault. I wasn't reading carefully enough
Right. User key user

2 Ok

We've been asked to select the project bench
so we just enter the letter b
in fact we have to go the whole way because it is not unique
select H
That will give us the exact model we've been asked for
On the display
Show the model in a broken mesh format
Mesh

[looks at manual]

In the manual under mesh
Mesh broken
Fine
Define a temp. subset model consisting of the finite elements in the range between 1 & 50.
So I shall do SeT append elements 1 to 50
I will then ask to define the set

Set show defined
with the hidden lines removed so I do view
View hidden line
this should then just give us a frontal view of the model with no back lines visible
which is clearly what it has indeed done
So far so good
we now require to look at the rest of the model with respect to the subset we have now defined
That's the EYE command and the option is
EYE LOCATE
& the rest of the model has been drawn in
Results loadcase case1
results loadcase C A S E 1
There is a no. of loadcases with the model
I'm now going to select a loadcase that will have results attributes in it
Displace nodal elements in all directions & present 10 contour levels
Results Nodal Displacement all
Present CONTOUR LEVEL 10
Displacement nodal all is the attribute that will give me displacements in axis x, y, & z
1st. task completed exit from the FV s/w back to the o/s.
Which means you'll have to type that thing again presumably
Fin

2 OK. if I just do this again

I'm being monitored all the time by the microphone
Not that it matters

2 Yes you are
2 It's being videoed all the time as well. But the 1st. one's just a test to get you warmed up.

So the next one's going to be much more difficult

2 yes this is the proper one now the real mcKoy.
2 If you'd like to go ahead again. Same sort of thing.

Entering user key
turning page
select project bench
select model TTBO
Oh

2 Can you explain what happened exactly there

Urh! We do slash & go back to index models
select projects

[types in silence]

2 What went wrong?

I in fact then didn't see the little m on the end & selected bench instead of benchmark & therefore I was thrown but the next command involved TTBO which in fact was not in that part. project Therefore one had to return to the project

level selecting TTBO
Fine
Rotate into plan elevation, the initial offset of all models is a 20 degree rotation in both the x & y.
Into a plan elevation. The initial offset of all models is a 20 degree rotation in both the x & y directions.
Well I guess that must be an EYE ROTATE to 0 0 0.
Into a plan elevation
Well that could be a plan elevation
I'm not sure exactly what they mean by a plan elevation
2 OK

[long pause]

2 Should we assume that that's what they mean & continue

Well I don't think so because it says show mesh construct in full perspective. With hidden lines removed.
Well they are hidden lines.
Rotate into a plan elevation
I don't understand
Perhaps if we proceed with eye rotate to 45 45 45.
I'll proceed with that
Show a mesh construct. So I'm viewing the mesh
This is the mesh that we have here
Show a mesh construct in full perspective
Full perspective comes under the eye command
I believe it is in fact EYE pERSPECTIVE
But I believe there is an option at the end of the EYE PERSPECTIVE which will give a degree of the perspective that one can in fact place on a given model
So eye perspective
Well lets try eye perspective 3

[long pause]

Redisplay the model with lines in a dashed format
So we will do
with hidden lines removed so
VIEW HIDDEN LINE
I shall use the semicolon option here
With lines in a dashed format
So I shall do view edges
I'm, guessing it's a view edges command

[checks in manual]

view edges
It is not view edges

[pause]

2 So what is it if it's not view edges?

I've just tried, just entered mesh broken
I'm having a think about that. I don't think that's lets just look at this.
Well it's obviously given it quite a problem here because it's thinking about it.

[long pause]

Right this is clearly not what we wanted. What we wanted here was an outline 7f the model
So in fact I should have used the eye command & that is EYE OUTLINE
So enter the eye outline command after this picture has been completed
Enlarge to fill the whole screen

[Pause - waiting for picture to draw]

This is obviously a bad move

2 You picked a hum dinger of a command their

Well it obviously requires a lot of computation

2 Would you normally be able to do something about this

Unfortunately not this is one of the problems with FV
Is that once you've started on a command you can't stop. When it gets going.
Right
I shall now enter eye outline
EYE OUTLINE
look back at the manual

[long pause]

Cancelling that command
I'm passing on. I'm not sure how to get a dotted line but, I've got them. I've succeeded. Presumably that's because I had the mesh command on.
I've been asked now to fill the whole screen.
Drawing contents command off
Colon eye frame
I'm switching off the commands on the right hand side & filling so I get a nice blown up picture the biggest picture In get on the screen of the model.
I've now been asked to copy the model to a viewport
So drawing copy viewport 1
slash
Drawing
Slash
Drawing viewport use 1
It seems to have done the trick
Results nodal. I've been asked to select all the loadcases
Results loadcase all.
Present the deformed shape with a scaling factor of
The next command is results nodal displacement x
For the x displacement.
Present the deformed shape with a scaling factor of 50.
Present shape 50.
We have now nice & neatly done.
In viewports 2 , 3, & 4 show similar diagrams with nodal displacements in the y axis, z-axis & all axis respectively.
Right. So
Now copying viewport 1 to 2
Drawing viewport copy 1.... to 2.

Learning a little bit now
& now doing results nodal displacement x semicolon present shape 50. results nodal displace y colon present shape 50. I'm now going to copy across but to save time we'll now use the semicolon command on the drawing viewport use.
So I shall now drawing viewport use 3 semicolon results nodal displacements z semicolon present shape 50
I shall now go on & use the 4 th. one where all the attributes are used.
Drawing viewport use 4 semicolon results nodal displace all semicolon present shape 50.
In the viewport add the following text. Text required in viewport 4.
Drawing content. drawing contents text
Add monitor text
Text required P L A N B Y top right TTBO
Cursor keys, I do not know where they are
Right space bar
No its normally a space bar for a hit
Hitting. Trying to get the key
Do not know which key to activate the text command
Space bar. It doesn't like space bar. Pass

2 Shall we skip that one & go on

I don't know how to. OK. well again that's it
Slash
Doesn't like it

2 OK so we're stuck at the moment are we?

I'm afraid so
I don't know its the space bar back at --- so
slash
Is John there

2 I'll just see if I can find him
2 You can't actually get out of that you don't think.

Well I could try hitting something. Well I could

2 I'll just have a quick look to see if John's around.
2 Well what we can do is quit out of that with control Q
2 Right & we'll put in a new monitor for you.
2 go on
Right a slight problem there. The last command was just drawing save with the title. That would then complete the ex. 2.

2 We have a problem because the keys are different on here to the one you have.

You don't use FV at all.

2 No. I am a totally naive user here.

It looks quite impressive in colour.
So user key again
Right. This is figure
We require model BTLH

BTLH. Select BTLH
Top right BTLH
Right
We've been asked to reproduce that as closely as we can.
4 viewports
Drawing viewport use 1.
Model has now been constructed that is in fact the. In fact we have the mesh on so we just view the mesh.
& of course it's mesh outline.
It's a fairly course mesh. It hasn't taken too long.
Right View hidden line
View mesh
We should now have the figure that's Urm.
I think I've given it a good one again
We don't seem to be having too much success with this one

[pause]

Most of the meshes we've been working with have been 2-d so by 3-d a little resting.

[pause]

In fact what I've done is I've done View Hidden line which is now given me a view hidden line. I've asked it to give me the mesh so it will redraw the whole thing again with the mesh superimposed on it.
That is my belief anyway.
Had I used view hidden line semicolon view mesh it would have ignored the 1st command & I'd be advancing a bit quicker.
Almost there

[long pause]

Lets just have a look at the view mesh command

[looks in manual]

View mesh
Mesh outline
View mesh
OK. Lets just move on
The model has clearly been rotated round about the y-axis through.
Right hand cork screw rule
minus 180
So let me do Drawing viewport use 2
Drawing viewport copy 1 to 2 semicolon eye rotate to 0 -180 0.
See if that is in fact any use to us.
Does that look about right we ask ourselves
Well of course it has worked but the original model was not in fact rotated.
We in fact require something like Drawing.
Eye rotate to 20 -160 20.
See if that's given us a bit better
Well it's almost there
The x doesn't require quite so much rotation.

2 Shall we call it that

We'll call it that. I can play around all day & still not get it right. Again I am now struggling because I merely have to do mesh shrink.
& I've shrunken the mesh
But I'm getting all the mesh on the
I'll just have a quick look at the view hidden line command.

[looks at manual]

View ...hidden
Right. Well I've got the same problem as I had in the 1st. one.
I'm not sure how to get the mesh just on the front.
Let me do
Drawing Viewport copy 2 to 3 semicolon
I want to do ... Mesh Broken semicolon Results nodal displacement all. Present contour level 4.
Just checking in fact how far the datum are apart.
& in fact we can see that the datum are point 4 8.
Right
so the contour levels are. Let me just do that & see what I get 1st. of all.
Results Loadcase not selected.
Well in fact it has not put me into the.
Drawing viewport copy 2 to 3 semicolon I missed out. Mesh broken semicolon Results nodal displacement all.
It's the wrong command. I can't see now. In fact there we go.
Results loadcase L1.
L1 is the loadcase in top left.
Semicolon Results nodal displacement all semicolon
Present contour level datum 1.22 steps .48
Steps .48
Perhaps I'm being to ambitious. Let's just move into.
Drawing viewport copy
Copy 2 to 3.
It's now drawing it in the mesh broken & the command seems to have in fact worked.
I want Results loadcase L1
Results nodal displacement all
Present Contour datum 1.22 steps .48

2 & that will?

& it's now drawing on the contour levels
It looks not to be what is required. Probably because I haven't in fact specified the range of results required on the model
But the steps should be in steps of .48 beginning at 1.22

2 Shall we call that one OK & move on

Ok
I've got my min & max & it's going up to e. I've got an extra contour there.
So I'll just move on to the last one.
One I should be good at because it just looks like a view hidden line.
So let me do Drawing

Viewport use 1
It says in fact
Drawing viewport use 4. Drawing copy viewport 1 to 4. semicolon results loadcase L1 semicolon results nodal displacement all.
Drawing viewport copy 1 to 4
Results loadcase L1
results nodal displacement all
& I'm now required to do view hidden line
That's what it says I believe

2 Can you talk me thru' the problem you were having earlier with your incorrect command. Is that just a syntax error or...

Urm Drawing viewport here?

2 Yes

Well I had in fact entered the command correctly
Perhaps trying to be too clever & I could probably benefited from in fact using the command in the FV manual which I don't have to use a semicolon I can suppress all drawings & enter the commands one at a time & in that way I don't have to string the commands in a long line & if I get 1 wrong in a long line, it throws me & I'm wasting time entering the line after.
So that would in fact be useful had I used this other command where I could type in the commands one at a time & the computer would tell me if they are right or wrong.

2 Which do you prefer

In fact I should use the other one. It's something I didn't know existed actually until this morning when I was just brushing up on my FV coming up on the train. I don't know if I need it or not but I thought I'd use the opportunity to read the manual a bit better & pick up a few useful commands.
Well this is looking good. This is clearly what is required in the last one. I'm looking now at the contour levels & seeing that they are in fact spaced out by .26 so in fact I shall enter a similar command to the command I did previously. I'm not sure how to terminate a level 3.37 without using the manual but we'll have a stab & then proceed accordingly. Although I doubt whether the contour level will come out correctly.
Right Present contour datum 1.04 steps of .026.
We'll see now what that give us. & in fact I think we've cracked it.
It looks as though it is exactly what's on the. So we've cracked it. Success.

2 Great.

100% right except interesting enough we've got 3.38 compared with 3.37 and 1.04.
I'd be interested to learn why that is the case. Perhaps someone could tell me that later on.
That completes the 3 rd. test.

2 Can I ask you now to finish. We're running a little bit short of time.

Fin

2 & if I could ask some questions of you.

Surely.

Subject 3.

Right
So I just log onto FV
I guess that's just user [authorisation code]
password
Right, user
login incorrect

½it's FV to start¼

Enter monitor filename?

{holds up sheet with monitor name}
{reads out monitor name while typing}
{silence while display is drawn}

User key?

{reads BENCHMARK}

Ah!! User
So... I want to select bench
& I want to....
Select housin

{Reads instructions quietly to himself}

Set
APPEND
ELEMENTS
So we want 1
to 50

2 So what have you done there?

Defined a temp. subset [points to request on sheet]

[reads from benchmark] Ah!!

Range between 1 & 50.

[to keyboard]

Set [to himself]
Is it off? I think
Cancel
Append elements 1 to 50
Right. Reads next part of BENCHMARK

2 So if you talk thru' what you're doing while you're doing

it.
Right.

[looks at screen]

So it's going to be show

[looks at screen]

Append
I'm getting lost here

{types in silence}

2 So why did you get lost?

[looks around the screen]

Wait a minute.

[looks at BENCHMARK]
[reads exercise out loud]

I'm trying to find ; display the defined subset.
I thought it was set show

[looks at keyboard]

Defined is it!

[leans back & scratches head]

Right.

[looks at BENCHMARK]

View with hidden line.

[starts to type]

Ah... fill...that should be format

[waits for display]

2 What was that then?

[points to screen]

I was doing a view

[points to BENCHMARK]
[reads out]

View subset. I was viewing it with hidden lines removed

[looks at screen]

& I thought the command was full.

{the command in 3.5}

[looks at BENCHMARK]

Hidden line so I messed it up

[looks at BENCHMARK & reads]

...select filename superimposing the whole model.
I'm not sure how to do this

[rubs hands on hips]

Do an eye & locate
& that's that

[reads from BENCHMARK]

...select loadcase...

[looks at screen]
[types in silence]

I'm just selecting a loadcase &....

[looks at BENCHMARK]

Results...we want nodal do we? Displace nodal

[checks BENCHMARK]

& displacements & I want the ....

[looks at BENCHMARK]

....all

[looks at screen]

2 So what have you done now?

[looks at BENCHMARK & reads]
Select loadcase case1 &...sorry...displace nodal elements in all directions
Ah!! I've displaced 'nodes' in all directions.

[looks at screen]
[rubs hands]

So I wanted elements.

[back to keyboard]

I'm not doing very well here
Ah!![looks at BENCHMARK & reads]...displace...
Results....[points to screen] is that just a cursor then 2 dashes is it...Yeh.
Right....element

2 So what happened?

Results elements [reads from screen]
Results loadcase not selected ½This is an error message¼
not selected?
It's not taping is it? Now?

[looks at observer]

2 Yes

Oh no!

[looks at screen & gestures with arms open]

Right...Results
Element
Results not selected.½again¼ Loadcase

[while typing]

Results loadcase case1
Results elements

[taps the table]
[looks at BENCHMARK]
[sits back in chair]

I've selected loadcase 1
I'm missing something obvious here

[leans forward]

Results loadcase 1

[holds bridge of nose]
[as he types]

Results Nodal Displace all
I'll take it at that being correct

[goes to BENCHMARK]

Displace elements nodal all. Present contour levels
Right. Present contour levels [looks at BENCHMARK]
10.

2 So how did you get 'round the previous problem.

I must say I might be interpreting this wrong.

[looks at BENCHMARK]

My interpretation was to present the nodal displacements
then generate the contour which I've done there.
But I was then, I was mislead somewhere here [points to BENCHMARK]
Displace Nodal Elements
They want the elements displaced
I tried to select that. I'm not sure about that.
So...We've done that to an extent. Whether it's right.
Lets have a look at what else. So we are going to go on to
ex 2. now.

2 Well lets finish ex. 1 first.

Sorry

[turns back the page]

Right

[types finish]

2 go on now.

Sorry

2 Go on to ex 2

Can I just have a quick look at it first?

[Reads BENCHMARK]

log back into FV....[reads quietly to himself]

[shakes his head & puts paper down]

I'm not sure I can do all this

[laughs]

We can only try can't we

2 Of course.

Right. Log back into FV.
& we want user, sorry monitor filename

[looks at observer]

2 It's the same as before except 2

[points to previous filename on screen]
[while typing]

User key, User.
Right, so we want to select bench &
select TTBO
Right,. Rotate into a plan elevation.
Urh!!

2 If you talk your way thru' what you're doing.

'Trying to rotate it to a plan elevation
All I can think to do is to do an eye rotate

[looks at BENCHMARK]

20 degrees in both the x & y directions.

[looks at screen]

I'll try to see what that tells us
I'm not sure...20
I'm not sure what that's doing

[reads from BENCHMARK]

The initial offset of the model is 20 degrees in both the x & y directions.

[looks at screen]

I'll try another eye
Rotate

2 So what are you doing now?

I've done it 20 degrees but I'm not convinced I did that right.
So I'm doing it again.
Eye rotate left, right?

[gestures with arms]

Rotate about the z clockwise
to somebody help, it's a real sort of...
eye rotate to 20
hum. 20 space 20.
I'm not sure that that's right but we'll go on saying it's right.
Redisplay the model showing in a mesh construct in full perspective.
View. Just check that
Mesh

2 So what have you done now?

So we've viewed the mesh now I'm going to view it in hidden line.

[looks carefully at screen]

view backspace hidden full

[looks at BENCHMARK]

Right
It's going to take a while to do this. I should have done a quick hidden line

[looks at BENCHMARK]

Redisplay model with lines in a dashed format.
I'm not sure how to get a dashed format for lines

2 There's a manual there

Sorry. I'm forgetting that the manuals there

[picks up manual & searches in silence]
[long pause]

I've got to find this

[looks at BENCHMARK]

Redisplay model with lines in a dashed format

[looks at screen]

2 So what have you done now?

Got to display that with dashed lines so I would image it's
view line
Lets try that
Back to the original
No..I'm not sure how to do that so we'll...
Enlarge

[looks at screen]

zoom
Out

[reaches for mouse & moves]

2 So what are you doing now?

[moves keyboard to make room for mouse movements]

That should take p the maximum area.

[moves hands back to keyboard]

So that should do it

[Reads from screen] Accessing the file
[looks at BENCHMARK]

Copy this view into vp. 4

[goes to manual, turning away from observer]

I haven't used viewports before so that's going to be new to me.

[looks at manual]
[long pause]

viewport drawing

[back to keyboard]

D

[looks at menu on screen & reads list]

Display, plot, save, viewport. viewport.

{types v into computer}

[back to manual]

{turns to page on viewports in manual}

2 So what are you looking for in the manual?

viewport

[looks at BENCHMARK]

So if we say [goes to keyboard] Use
viewport
& we generate the first viewport.
So copy
one to 4. Let's see if that
No [scratches head]

{In fact the default vp. is 0 so copy 0 to 4 would have been correct}

[looks at manual]

Let's try use zero

2 So what happened there. What did you do & what happened?

[looks at manual the immediately at screen]

I'm trying to put the model in 4 viewports

2 & what did you use to try to do it?

I tried to copy 1 to 4 which is not right & i've got to sit down & try to think about it.

[looks at manual]

Lets try: Use space 2

2 So what are you doing now?

Trying to put that model into vp. 2 & not having much luck.

[back to manual]

2 So why did you use view 2.

No. I was trying to Urh!..

[reads manual]

Try that. Copy vp. 1 to 2 .
Incorrect command. {missed out drawing viewport}

[looks at manual]

Not sure about that. Let's see what else I have to do.

[reads next part of the BENCHMARK]

Right. We'll try that.

2 So what are you doing now?

Select a loadcase.

2 So you're not trying to use viewports. You're carrying

on to
the next one?
Carrying on to the next one ... see what happens

[types in silence]

2 Tell us what you are doing.

Select loadcase... Ah!!

2 & why are you doing it?

Well we're trying to find

[looks at BENCHMARK]

Select all loadcases & make a nodal displacement in the x-axis
using viewport, select all loadcases
I'm not sure about that. I'd imagine ...
Well I'll try all but I'd thought we'd need to.
Oh!.. No... That's right.

[looks at BENCHMARK]

Select loadcase all
Results nodal displacement
& with displacement in the x direction.
backspace x
Right & ...Present
I want to present shape
'Not specified the scaling factor

[rubs ring on hand]

factors assumed to be 118

[looks at BENCHMARK]
[looks at manual]
[looks at BENCHMARK]
[looks at screen]

2 So what's....

I'm puzzled now
I've stepped one step but now we've Urh!!

[looks at BENCHMARK]

Using vp. 1 select loadcase & make a displacement in the x axis & present the deformed shape.

[looks at screen]

Ok. We've presented the deformed shape

[looks at BENCHMARK]

I've still got to get these viewports 2 & 3 to carry on.

[looks at manual]

2 So what are you doing now?

[while still looking at manual]

Looking to see how I get 4 viewports

[pause]
[looks at BENCHMARK]

Tell you what we'll...
Just wondered whether to go on to the next one.

[Turns page & looks at ex 3]

I'm struck on this viewports

[goes back to manual]

I want to know how to copy that view into the 3 other viewports

[points on the screen]
[back to the manual]
[reads from manual]

If going from a single vp. to multiple.....

[covers mouth with hand]
[long pause]

Doesn't make sense!

2 So what's the problem here?

Trying to put this model into 3 viewports

2 Why doesn't it make sense then?

[looks back at manual]

I've used...I don't know.
Maybe it's my fault.
I should have used zero for the full screen to single vp. supposedly.
Let's start again.
Redraw.

2 So what are you doing now?

Just clearing the screen to give us space for my commands..Do
Ah!!
Why did you do it?
Just to clear the commands off.
Right.

[looks at BENCHMARK]

I might go back to the beginning & try it.

2 So why is that?

Cause I'm not sure how to get these things from there

2 Just keep talking through it.

Right. I'm sorry.

2 & why you're doing it.

Yes!...Use...Well I'm trying to use the full screen. I'll try that.
Ah!! Incorrect command B 200

[looks at manual]

Wait a minute. We'll try drawing viewport use 0.

2 So what have you done?

I'm using drawing vp. use 0.

2 What's happened now?

Hopefully I was going to...[screen changes]
Yes!. I was going to put....
So that's putting the model onto the full screen.

2 So you're back to the full screen.

Back to the full screen.

[holds hands together]

2 Why have you done this?

To try & then figure out how to go back into 4.

[looks at manual]

Right.
Drawing
Viewport
Now we're going to...

2 What are you doing now?

I'm going to use 1 this time & try to get back to the view we had before.
Right. So then we're going to do...
Drawing Viewport copy
& we're going to copy 1 to 2
Right

2 So what...

So we've managed that. Now I just do the...
So drawing
viewport

copy 2 to 3

2 So why were you puzzled then?

Swapped windows. I'm not familiar with that.
Right, backspace copy 2 to 3.
We're getting round it now.

[gestures with hands]

My mistake before was that I wasn't going back into drawing viewport then doing the command.
I was just issuing a command at the main menu level. So it was incorrect.
Unless there's a faster way of doing it. I'm not sure.
viewport copy 3 to 4.
Right

[gestures with hands]

Now we want from these instructions.
Using viewport 1
so we should be back to this one now.

[looks at BENCHMARK]

Select all loadcases
Results
loadcase
all

3 When you've done this do you want to finish otherwise you'll run into the others time. We're a bit short of time.

[laughs]

when we've done this! [points to BENCHMARK]
So results loadcase all
Results nodal
So I've selected loadcase all & I'm now going to select the nodal displacements

[stars to type]

...displacements in the x direction & then we want to present shape
So.
Present
Shape

2 What's happened now?

Presented the displaced shape with a nodal displacement in x on that model in vp. 1
Right.
I'm not sure how to get the scaling factor, so just assume the max.

[looks at BENCHMARK]

So in viewports 2, 3 & 4 in y & z axis
Right so that's easy now.
So drawing
viewport use
and I want to go to 2 next, so the next vp.
So come back to that one.
& then we want to select a...
displacement in the y direction
Results
Displacements
y

2 So what's happened here?

I've got an incorrect command.
Results loadcase
nodal.

[gestures with hands]

I've got to reselect the loadcase for that vp.
Right
results loadcase all
& then I want Results
nodal
displacement

[looks at BENCHMARK]

y direction.

2 That's Ok. So if you'd just tell me what you're done.

So I've entered the...

2 What did you do first.

Sorry. I've selected loadcase in vp. 2. I've then selected nodal displacements in the y direction & now I'm going to present the shape...

[stops & stares at screen]
[looks at manual & starts turning the pages]

2 What are you doing now?

I'm just going to quickly spot if I can set the scaling factor.
No I don't know.
Right

[back to the screen]

Anyway we've done that one
So we're going to go to 3 & 4 now.
So, Drawing
viewport use 3
& then we want to
Select the loadcase
Results loadcase all

& then we want to display & we want to select nodal displacements in the z direction
Results Nodal Displacements in the z direction
& present shape.
The next thing is we go into the next one & do the same thing for nodal displacements in all directions.

2 So what are you doing now?

Right. So now I'm going to select vp. 4
Drawing viewport use 4.
& we want to select the loadcase again
select loadcase all
& then we want to select
Results nodal displacements in all directions
& then present shape
& the next thing we want to do is in vp. 4...following text.
Right. Adding text? I don't know how to do that.

[looks at manual]

text
That's interesting drawing

2 If you talk thru' what you're doing.

I'm just looking thru' the manual to try & find out how to add text to the vp. 4

[from the manual] using text....

Right
So we want to do Drawing
contents & we want to do
Text.
Drawing contents text

[reads from screen] add monitor, max. 80 characters.
[looks at manual]

2 What's the problem here?

I've been asked to enter text & I want to put it in
Plan view of TTBO
& where do we want to put that.

[looks at BENCHMARK]
[moves to mouse]

Put it there

[clicks mouse button]

& then we want to
add some more...
Drawing contents text

[looks at BENCHMARK]

By -------

& it's ....Date today?

2 [give date]

So that's that put on.
So we've done that

[looks at BENCHMARK]

& we want to do a drawing save
& we'll save it.
Title of drawing view 2
So we want to quit out of that {enters F}
That was abysmal.

Subject 4.

So how do we log into FV. Does it say.
login....

2 Right the monitor filename is ......

right.
I'm going to pick the...
bench project
'Forgot the select.
& I've got to be in upper-case.
Select housin
is it housin we want?

[checks BENCHMARK]

yes
Right. Ok. we're into FV.
Define a temporary model subset consisting of.....
I don't really know what that means?

[pause]

Don't know what that means at all

[long pause]

I'll try the options menu, see if that tells me anything.
No

[long pause]
[looks through manual]

I think I'll have to miss that thing cause I have no idea what it means.
None at all.
Ok.

[from BENCHMARK quietly to himself]

Locate the currently selected finite elements by superimposing an outline representation of the whole model.
Right.

2 So what are you doing now?

I'm viewing the mesh at the moment.
I don't know whether that's quite what we want.

[pause]

Let's have a look at the overview.
Oh crikey!
This really doesn't help at all

2 So what are you going to do?

Have another look at help.

Perhaps commands.

[quietly to himself]

J***s C****t
I wander if view section's what I want.

[long pause]
[types in silence]

2 So what have you done there?

I'm trying to figure out how to pick this model subset but I obviously haven't got it right at all.
I'll try help set.

[pause]

2 So what are you looking for?

I'm still trying to find out how to do this select the model subset.
Results possibly
Element
No

[pause]

Try help again

[pause]
[types in silence]

This system, unless you know exactly how to do things is hopeless.
The help as far as I can see is practically no use at all.

[looks at manual]

Possibly if you had a bit more experience of it you might know what help meant but it really doesn't tell you anything.
Help Commands.
Now previously that was showing me a list of commands. We still appear to have the set.
List up on there
Perhaps put something else up on the screen.
& then do help commands.
No, still doesn't do anything.
As a help system, it's a bit useless.

[looks thru' manual]

Mesh......[cont. turning pages]
Ah!, Display a line of nodes
No that isn't the same as what we're trying to do.

[Back to turning pages in manual]

2 What kind of thing are you looking for?

I'm still trying to figure out how to do this part one.
Which is to define a temporary model subset.
Which I still haven't been able to find out.

2 What are you actually looking for in the manual?

I'm trying to find the command to do that.
Which ....

[turns pages of manual]

Ah!! here we are [quietly to himself]
Try this rubbish
Mesh label
Right
Now perhaps it's this
Elements
Ah!!
No it's just labelling the model.

2 What did you try there?

I tried mesh label elements, thinking that it would actually enable me to label this mesh subset.

[Back to manual]

Results [continues to look thru]

[pause]

Try present. No

½At this point he is checking each command in the menu with it's description in the manual.¼

Ah!! Subsets.
So it's obviously the set command that I need to do the the subset.

[from the manual]

Append elements to the subset. Remove, show.
Append
elements
1.
Oh. Right.
I wonder if we can do append elements..
1 to 50
No.
It says error 200. Incorrect command.

[looks at manual]

What I was thinking, append elements 1 to 50 to my new...Oh.

½the problem at this point is that the command is SET append... and he has only put APPEND ...¼

That's what I said.

[pause]

Oh. I forgot the set.
set append elements 1 to 50
Ah! right. Now I appear to have done that.

[looks at BENCHMARK]

So I'm going to save the current thing.
I'll call it ----
No set active.
Oh g*d

[looks at manual]

Set show
defined.
ah. So that appears to be my subset.

[pause]

Right.
View edges

2 So what are you doing now?

I now appear to just have the subset rather than the whole model.
Assume what we want to do now is locate it so we'll use eye.
Ok.
So there we are.
So I've done parts 1 & 2.
Select loadcase.
Results
loadcase
& I want loadcase case1
Right
So I think I've done this bit £ machine fns. No feedback.

[from BENCHMARK]

Displace nodal elements in all directions.

[pause]

2 So what's the problem now?

I've no idea which command to use again.
Perhaps it's present.
I think it might be present.
Right present.
contour
levels 10

½This requires a results nodal displacements all before contours can be shown¼

Ah. I've got to say what I want to show.
So I've got to tell it that I want to do this displace nodal elements.
Perhaps it's the view command.
No.
Results.

[pause]

For some reason the help commands doesn't appear to...
Oh dear.

2 So what happened there?

I've no idea, I thought I was in help.
& I wanted overview.
We now appear to be in options.

{ In fact he had entered help commands overview when the correct command was help overview. It didn't recognise the O, he then re-enters help overview with the effect of giving him help commands help & then the O is recognised as Option from the main menu. He finally rejects this command & enters the correct commands }

Go back into help.
Overview
Ah!! right.
So I've picked the loadcase.
What I want to do is this displace nodal elements stuff.

[pause]

Hmmm.
Which I've no idea how to do.
No idea at all.
Scan thru the manual again.

[looks thru' manual]
[long pause]

Oh dear.
I don't seem to be able to find this instruction.

[while still looking thru' the manual]

No. we've gone thru' sets, that's not it.

[looks thru' manual]

Oh. I really haven't any idea how to do this last bit.
None at all.
It would be useful in the help system if you could have a more detailed version of each of these.
It really doesn't tell you anything.
I'll do help command view.

[reads from screen]

View edges.

One would assume an engineer who uses this every day would learn to use the system despite the system. Just by knowing certain commands did certain thing. But there is certainly no obvious linkage between the instructions given or the commands you can type in & what they can actually do.

[long pause]

2 So what are you doing now?

Just trying odd things.
Went into results there.
But that really didn't tell me anything.
Perhaps if I did help commands results
No it really doesn't help at all.

2 Perhaps you could explain what you're trying to do.

I'm trying to find the command to do this last part.
This displace nodal elements.
& I can see any command that does that.
I can do the 10 contour levels
I've found that bit. But there is no obvious way to do it either looking at the help facility or looking in the manual.

[looks thru' manual]

Present.
Possibility. No I've looked at present.

[long pause while looks thru' manual]

No. I think I've come to a dead halt.
Unless working with FV might give us a hint.
Ah. Perhaps accessing results 3 6.

[turns pages of manual]

So we've done the results loadcase. Yes
Ah.

[pause]

Still doesn't help

[cont. to look thru manual]

Try results
element
Give's us weird error message 450.
Which doesn't actually tell us anything.

3 I'm just a bit conscious of the time....

Ah. well I've come to a point where I am totally stuck John.

3 Ah. Well that's perhaps a good point to stop then.

I cannot see how to do the commands.

3 Just do the debrief and.....

Ok. So if I do finish.
Finish.
There we are.

A9

Defining the dimensions used to construct the scales.

Defining the dimensions used to construct the scales.

In table 3.2 the 23 original dimensions, as presented by Poulson, 1987(a), 1987(b), are given. The remaining 10 dimensions have been produced by the author and cover aspects of - the system, the benchmark test and user characteristics.

Level of Instruction

Refers to the quality of any instructions provided with the equipment.

Training

Refers to the quality of training given in using the equipment.

User Support

Refers to the availability of human help in using the equipment.

Experience of Usage

Refers to the degree of experience in using the equipment.

Availability

Refers to the extent to which the equipment is available when the user wants to operate it.

Discretion of Usage

Refers to the degree of control that the user has as to whether or not they use the equipment.

Concentration Required

Refers to the mental effort or concentration required in order to use the equipment.

Fatigue in Use

Refers to the extent that using the equipment is felt to be tiring.

Ease of Use

Refers to the lack of effort that was required in order for the user to learn to operate the equipment.

Simplicity of Use

Refers to the lack of complexity in operating the equipment.

Enjoyment

Refers to the degree to which using the equipment is perceived as a pleasant experience.

Satisfaction

Refers to the degree to which the user is satisfied with the system.

Frustration

Refers to the extent to which using the equipment is a source of irritation or frustration.

Flexibility of Usage

Refers to the extent to which the equipment allows the user to achieve the same objectives in different ways.

Range of Application

Refers to the power of the system, or the range of activities which can be carried out using the equipment.

Utility

Refers to the perceived value or use of the equipment to the user.

Need for Improvement

Refers to the degree that the user believes the equipment should be improved.

Perceived Control

Refers to the extent to which the user believes they have mastered or are in control of the equipment.

Transparency

Refers to the degree to which the equipment's operations are obvious or transparent to the user.

Ease of Error Correction

Refers to the ease with which users mistakes can be corrected when using the equipment.

Reliability of Operation

Refers to the user's confidence that the equipment will not go wrong.

Speed of Operation

Refers to the speed of operation of the equipment, and the perception that it takes too long to use.

Visual Appearance

Refers to the degree to which the equipment is aesthetically pleasing.

Usefulness of manual

Refers to the quality of the manual for problem solving.

Value for money

Refers to the percieved value for money of the software.

Usefulness for FE work

Refers to the preference of the software for displaying results of finite element analysis.

Design of benchmark

Refers to how realistically the benchmark exercise simulated real world tasks.

Difficulty of task

Refers to the level of difficulty of the benchmark exercises.

Protocol distraction

Refers to the degree to which talking through ideas was considered off-putting.

Stress of benchmark

Refers to the level of stress the evaluation procedures put on the users.

Level of critique

Refers to the general level of critism the user has towards computer systems generally.

Benefits of colour

Refers to the perceived benefits colour displays have over monochromatic displays (both in the use of FEMVIEW and generally).

Effects of frustration

Refers to how the users' consider their performance with the software will be affected by frustration.

A10

Femview users evaluation benchmark.

Femview Users Evaluation Benchmark.

The following series of exercises are designed not to test your ability but to test the Femview software. Relax, take as long as you want to carry out these exercises and if you wish to ask any questions please feel free to talk to the observer.

"Talk Through Your Ideas"

You will see a microphone in front of you. Please speak into it to explain the methods you are adopting and any questions or suggestions that come to mind during your use of the software. State whether the task you have been asked to perform seems trivial or complex, and verbalise on anything and everything that you are thinking about while you are carrying out the required exercises. The hope is that the recording will act as a verbal representation of the thinking that is required during an interactive session using the Femview software.

When you have read this note look through the exercises to familiarise yourself with the tasks, then indicate to the observer that you are ready to start. It is not neccessary to complete the commands in the sequence they are presented. However, please complete each exercise before moving on to the next....

EXERCISES.

Exercise 1.

This section will not be used in the evaluation. Its purpose is to give you a chance to become familiar with the type of tasks you are required to perform, and to allow you the opportunity to practise speaking, while carrying out a specified task.

> Log into FEMVIEW, (the observer will tell you the name to give to the monitor file) your authorisation code is ´USER´. Select project ´BENCH´ and model ´HOUSIN´. Show the model in a broken mesh format (dashed lines).
>
> Define a temporary model subset consisting of the finite elements in the range between 1 and 50. Display the defined subset with hidden lines removed.
>
> ´Locate´ the currently selected finite elements by superimposing an outline representation of the whole model. Hint: this is part of the EYE command.
>
> Select loadcase ´case1´, and select resultant nodal displacements. Present ten contour levels.
>
> Exit from the FEMVIEW software back to the operating system.

You have now completed the first exercise...I hope you managed to speak into the microphone while you were thinking about the tasks. Now lets start exercise 2. Tell the observer when you are ready to proceed....

Exercise 2.

Log back into FEMVIEW with the authorisation code 'USER'. Select project 'benchm' and model 'TTBO'. Rotate into a plan view (looking down the z axis. The initial offset of all models is a 20 degree rotation in both the X and Y directions). Show the model, in perspective, with hidden lines removed.

Redisplay the model with the lines in a dashed format.

Copy this view into viewports 1,2,3 and 4.

Using viewport 1, select the first loadcase and select nodal displacements in the x-axis. Present the deformed shape with a scaling factor of 50.

In viewports 2 and 3 show similar pictures with nodal displacements in the Y-axis and in all axes repectively.

In viewport 4 under the view of the model add the following text:

'Plan view of <model name>'
'By <your name> on <date>'

<...> means add appropiate information.

Save this screen using DRAWING PLOT with the title 'view2'.

You have now completed exercise 2.....Tell the observer when you are ready to proceed to exercise 3....

Exercise 3.

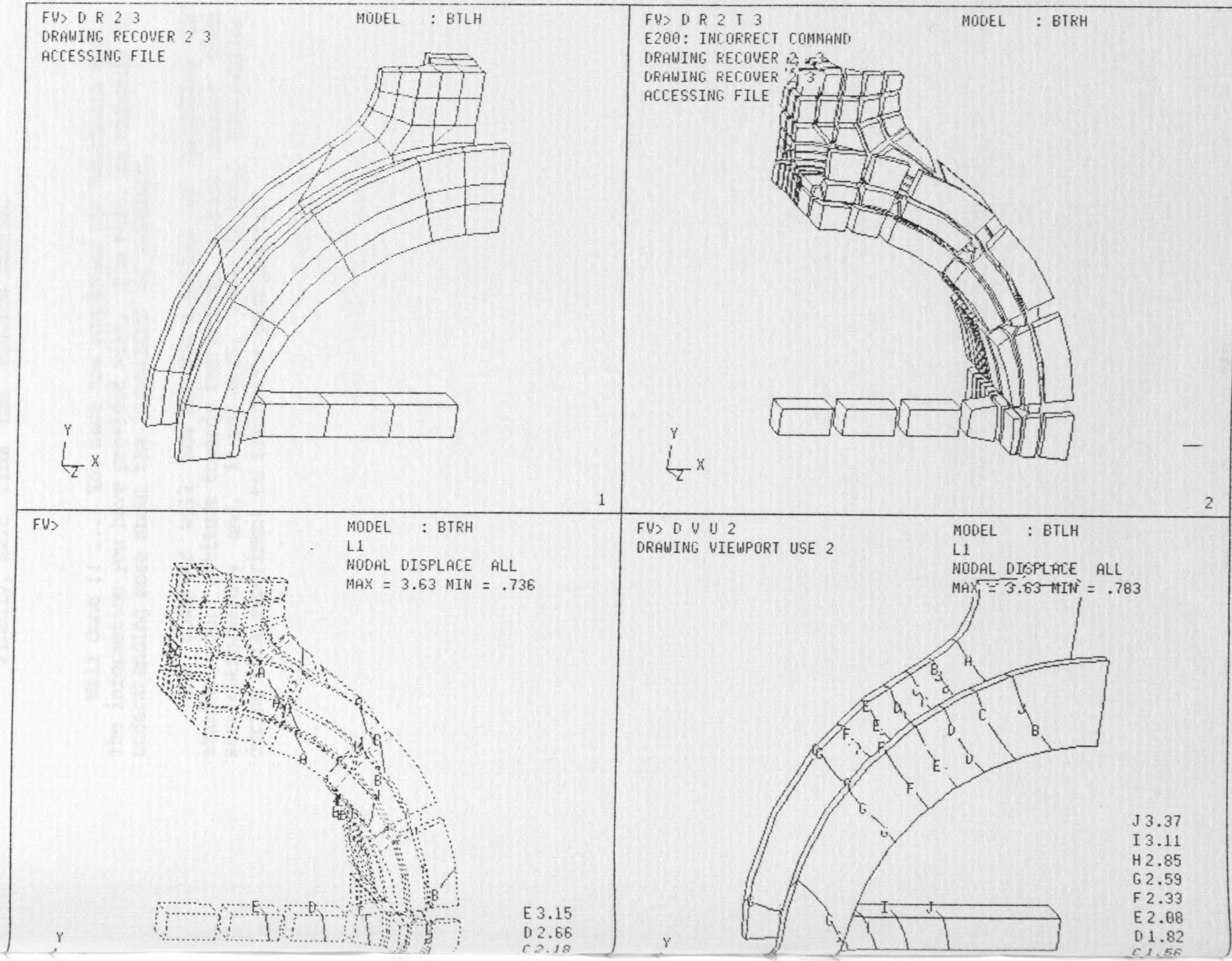
In figure 1 you are given views of 2 models. Try to reproduce these views in exactly the same format as that shown.

Once you are happy with your reproduction save it. Use the DRAWING PLOT command and give it the name 'view3'.

You have now completed exercise 3. Again, look through the forth and final exercise and let the observer know when you are ready to proceed....

Exercise 4.

TEXT BOUND
INTO
THE SPINE


FV> D R 2 3
DRAWING RECOVER 2 3
ACCESSING FILE
MODEL : BTLH
Y
Z X
1
FV> D R 2 T 3
E200: INCORRECT COMMAND
DRAWING RECOVER 2 3
DRAWING RECOVER 2 3
ACCESSING FILE
MODEL : BTRH
Y
Z X
2
FV>
MODEL : BTRH
L1
NODAL DISPLACE ALL
MAX = 3.63 MIN = .736
E 3.15
D 2.66
Y
FV> D V U 2
DRAWING VIEWPORT USE 2
MODEL : BTLH
L1
NODAL DISPLACE ALL
MAX = 3.63 MIN = .783
J 3.37
I 3.11
H 2.85
G 2.59
F 2.33
E 2.08
D 1.82
Y
C
V
M
R
P
O
E
S
U
D
I
H
F

It is thought that project ´TEST´ model ´HANGER´ has not been generated correctly. Examine the model carefully and record any faults in the mesh on the form provided (Table 4.1). Do not spend more than 5 minutes on this exercise. All faults are connectivity problems.

Finally, exit from the FEMVIEW system.

Well done !! .... You have now completed the benchmark test. The information you have provided will, I´m sure, be valuable in understanding more about the ´usability´ of software.

The observer will now ask you a series of questions to examine your attitude towards the system, please answer them straightforwardly and, if you wish, refer to your interaction during this experiment to illustrate your points.

<u>ERROR REPORT</u>

Use only with exercise 4.

Error No.	Nodes at fault	Probable fault
1.		
2.		
3.		
4.		
5.		
6.		

Table 4.1.

A11

Interpreting the 16 PF Profile Patterns.

Interpreting the 16 PF Profile Patterns.

Primary Source Traits.

A. Warmth.

High scores: Warm-hearted, easy going individuals, easy to get along with and trustful. Usually good in situations involving interpersonal relations. Will happily adapt to other peoples schedules. Prefers to work in a group.
Low scores: Individuals more oriented to things and ideas than to people. Stand by their own ideas, are critical and cautious in emotional expression. Prefer to work alone.

B. Intelligence.

High scores: Remember things more easily (high mental capacity), adaptable, fast learning and more persistent.
Low scores: Low mental capacity, are unable to handle abstract problems, are less well organised with a tendency to be quitters.

C. Emotional Stability.

High scores: Calm, stable individuals constant in their interests and less easily distracted. High scoring individuals more frequently become leaders and have generally good morale.
Low scores: These are emotional individuals, easily annoyed by things and people and generally dissatisfied with things. They are evasive of responsibilities, tending to give up quickly.

E. Dominance.

High scores: These are assertive dominant individuals, stubborn and competitive. They are independent, often stern and hostile and like things their own way.
Low scores: Found in more conforming, conventional, humble individuals. They tend to be more considerate of others, less assertive and diplomatic.

F. Impulsivity.

High scores: These are happy-go-lucky, cheerful individuals. They are often quick and alert to situations and talkative.
Low scores: Silent, introspectful individuals with a slow cautious manner.

G. Conformity.

High scores: More persistent individuals, more respectful of authority, and more conforming to standards and rules. They tend to express self-controlled behaviour and regard for others as opposed to emotional and impulsive behaviour. They are inflexible and as a result they are less able to cope with extreme stress.
Low scores: More usually associated with slack, self-indulgent, less responsible people. They often show little determination to do well.

H. Boldness.

High scores: Typically adventurous, bold, and energetic individuals who like being the focus of attention. Often express that taking part is more important than winning. They are quick, impulsive decision makers although not necessarily always making the correct choices.
Low scores: Shy, and tormented by unreasonable sense of inferiority, slow and poor at expressing themselves this person prefers being with

a few close friends to large groups. They have few interests and tend to be very deliberate in their actions.

I. Sensitivity.

High scores: Tender minded, clinging, insecure individuals. They report to prefer reason rather than force to get things done. Tend to act on sensitive intuition and are anxious about themselves in their actions and physical appearance.
Low scores: As individuals they expect little from others, tending to be self-reliant and responsible. They act on logical evidence rather than feelings, keeping to the point.

L. Suspiciousness.

High scores: Traits include suspecting, jealous, critical and irritable. They are particular about correcting errors of others and do not forget criticism easily. There is some relationship between this trait and paranoia.
Low scores: Easygoing, laid back individuals perhaps lacking ambition and drive.

M. Imagination.

High scores: Unconventional, impractical people unconcerned with everyday matters. They often forget trivial things and are easily seduced from practical judgements. They are generally enthused but on occasions give up.
Low scores: These are practical, objective people tending to be more concerned and worried than high scoring individuals but less susceptible to changes in enthusiasm.

N. Shrewdness.

High scores: Emotionally stable, ambitious people. Like to cut corners where possible but are considerate of others.
Low scores: Genuine but socially clumsy people with a blind trust in human nature. They have simple tastes and often content with what comes their way. They are warm-hearted and tend to have a lot of friends. They are more straightforward and less constrained by rules and standards.

O. Insecurity.

High scores: Worried, guilty, moody people who experience frequent episodes of depression.
Low scores: Self-confident, cheerful people insensitive to people's approval or disapproval (non-caring). They prefer to get on with things rather than worry about them.

Q1. Radicalism.

High scores: These are experimenting, liberal, analytical, free-thinking individuals. They are trusting of logic rather than feelings and will often break with the established way of doing things. They are often good problem solvers.
Low scores: Are more respectful of the established way of doing things and tolerant of traditional difficulties. They are more conservative in their approach to life than the higher scoring individuals.

Q2. Self-sufficiency.

High scores: Individuals with high score in this trait tend to be loners, preferring to make their own decisions than working with a group. They tend to be resourceful, adapting well to the problems they encounter.

Low scores: These are group dependent individuals depending on social approval. They are conventional and fashionable.

Q3. Self-discipline.

High scores: Generally high scoring people have strong control over their emotional life and behaviour. They prefer to get their thoughts organised before acting and are generally neat and tidy (well organised) rarely leaving things to chance.
Low scores: These are uncontrolled people who tend to follow their own urges rather than think carefully about them first. They are careless of social rules and slack.

Q4. Tension.

High scores: These are easily frustrated and irritated individuals. This trait is often associated with frustrated motivation and anxiety.
Low scores: These are relaxed, tranquil people, appearing composed and calm.

Second-order traits.

Qi. Extroversion.

Both extrovert and introvert behaviour is described in Section 4.6 of the main text.

Qii. Anxiety.

Anxiety needs little explanation, the main contributors to anxiety from the primary trait include emotional instability (low C score), threat sensitivity (low H score), suspiciousness (high L score), guilt (high O score), low integration (low Q3 score) and high tension (high Q4 score).

Qiii. Tough Poise.

High scores: Personality characteristic reflecting a cool, emotionally detached, and controlled individual. Very high scores are associated with individuals insensitive to the feelings of others.
Low scores: Sensitive but moody individuals who feel, rather than think their way through problems. They are often too attentive to the emotional aspects of situations to act clearly.

Qiv. Independence.

High scores: Individuals tend to show internal control of their actions and are not dependent on others.
Low scores: These are associated with people who are not dominant, radical or self-sufficient but subdued in their behaviour.