Order Number 8814285

Knowledge-based support for software selection in information centers: Design criteria, development issues, and empirical evaluation

Vinze, Ajay Shreekrishna, Ph.D.

The University of Arizona, 1988

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark __√__.

1. Glossy photographs or pages _____
2. Colored illustrations, paper or print _____
3. Photographs with dark background _____
4. Illustrations are poor copy _____
5. Pages with black marks, not original copy _____
6. Print shows through as there is text on both sides of page _____
7. Indistinct, broken or small print on several pages __√__
8. Print exceeds margin requirements _____
9. Tightly bound copy with print lost in spine _____
10. Computer printout pages with indistinct print _____
11. Page(s) _____ lacking when material received, and not available from school or author.
12. Page(s) _____ seem to be missing in numbering only as text follows.
13. Two pages numbered _____. Text follows.
14. Curling and wrinkled pages _____
15. Dissertation contains pages with print at a slant, filmed as received __√__
16. Other _____

U·M·I

KNOWLEDGE-BASED SUPPORT FOR SOFTWARE SELECTION IN INFORMATION CENTERS: DESIGN CRITERIA, DEVELOPMENT ISSUES, AND EMPIRICAL EVALUATION

by

Ajay Shreekrishna Vinze

A Dissertation Submitted to the

COMMITTEE ON BUSINESS ADMINISTRATION

In Partial Fulfillment of the Requirements
For the Degree of

DOCTOR OF PHILOSOPHY

In the Graduate College

THE UNIVERSITY OF ARIZONA

1 9 8 8

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Final Examination Committee, we certify that we have read the dissertation prepared by Ajay S. Vinze

entitled KNOWLEDGE-BASED SUPPORT FOR SOFTWARE SELECTION IN INFORMATION CENTERS: DESIGN CRITERIA, DEVELOPMENT ISSUES, AND EMPIRICAL EVALUATION

and recommend that it be accepted as fulfilling the dissertation requirement for the Degree of Doctor of Philosophy.

Jay F. Nunamaker, Jr.	4/13/88 Date
Douglas R. Vogel	4/13/88 Date
Sudha Ram	4/13/88 Date
W. Gary Wagner	4/13/88 Date
Paul C. Earley	4/13/88 Date

Final approval and acceptance of this dissertation is contingent upon the candidate's submission of the final copy of the dissertation to the Graduate College.

I hereby certify that I have read this dissertation prepared under my direction and recommend that it be accepted as fulfilling the dissertation requirement.

Dissertation Director Jay F. Nunamaker, Jr. 4/13/88 Date

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under the rules of the Library.

Brief quotations from this dissertation are allowable with special permission, provided that accurate acknowledgement of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the copyright holder.

SIGNED: ______________________

ACKNOWLEDGEMENT

To my parents Kalindi and Shreekrishna Vinze, and my wife Amita
Without your love and encouragement this effort would not have been possible.

The past three and a half years have been the most challenging and rewarding time for me. My thanks to the members of my committee, Jay Nunamaker, Douglas Vogel, Sudha Ram, Gary Wagner, and Chris Earley. A special word of thanks to Jay, my major advisor, who was a constant source of ideas and supporting resources to help improve my efforts.

Jean Weber from the Statistics Department, and Daryl Sabers from the Education Psychology Department were not on my committee, but their advice was invaluable. Also, Barb Sears who provided excellent editorial comments.

Special thanks to Mari Heltne and Minder Chen, my partners in building the ICE system. It was a real treat working with you! My thanks also to Kuni Higa, Rosemary Wild and Conrad Royksund for the numerous discussions we had, both academic and non-academic. The CMI ladies, Kendall, Irene and Yi-Ching were very helpful during the conducting of my experiment as were the CMI consultants. Thanks also to Irene, Betty, and Joan for getting me out of a fix on more than one occasion.

My wife, Amita, has put up with my mood swings as I went through the doctoral program. It is amazing that she still maintains her sanity. Her contributions to this dissertation, both tangible and intangible, are enormous.

Finally, my daughter Anika, helped me put things in perspective when events felt out of control.

Table of Contents

List of Figures

List of Tables

ABSTRACT

An information center (IC) is described as an organization designed to help end users help themselves. ICs are expected to provide several services to end users. The services can be summarized as: consultation, distribution and trouble-shooting. The research is focused on a specific consultation activity: software selection. Providing support for selection and evaluation of software for users constitutes 91.5 percent of a typical IC's daily workload.

In the last decade, ICs have proved successful in managing software resources for organizations. The initial success of ICs has increased user expectations and demand for the services offered but, because ICs are considered cost centers in most organizations, there is growing pressure for them to accomplish more with fewer resources.

The research hypothesis is that the knowledge and methodologies of IC consultants, concerning software selection, as well as relevant institutional policies, can be represented in a knowledge base. A knowledge-based system ICE (Information Center Expert) to assist users with software selection has been developed and evaluated in the study reported here.

The development of ICE used two main design criteria: maintainability and transportability. Maintainability was defined as the ability to support frequent updating of the software supported by an IC. This is important because new software tools are introduced in the market at a very rapid rate; to stay competitive an IC must be able continually to adapt to this dynamic environment. Transportability was considered necessary to make ICE usable in many different ICs, each supporting a different set of software. The transportability feature allows different ICs to individualize the system to meet their own site-specific needs.

Validation studies were conducted to test the appropriateness of the recommendations made by ICE, using "blind" validation procedures in which scenarios (in case form) were presented to consultants. The cases were selected to represent problems frequently taken to an IC. Two sets of solutions, those offered by consultants and those provided by ICE, were then presented to experts who were asked to judge the appropriateness of each solution to a case without knowing its source.

To test the comparative advantages of using ICE or IC consultants to obtain assistance with software selection a laboratory experiment was conducted. A hypothetical construct called "Consultation Effectiveness" was used, which included measures for "user satisfaction" with the process, as well as measures for the "task basis" and the "recommendation basis" for evaluating a consultation session.

Chapter 1

Introduction

1.1 Background

There appears to be a consensus that end-user computing will soon consume a majority of organizational computing resources. In some companies, this is already true. It is predicted that by 1990 end-user computing will represent as much as 75% of the total corporate computing capacity of a typical American firm. Consensus also suggests that the best strategy for managing end users is to give them computing tools, establish standards, provide help services, and encourage good computing practices.

Following the lead of IBM, many corporations have incorporated this strategy into a special entity called an Information Center (IC). Hammond [1982] describes the objectives of the IC as "providing users access to data/information on their own terms so that they can solve their own business problems." An information center (IC) is therefore an organization designed to provide "guided service to help end users help themselves." The information center concept is interpreted by different organizations in different ways. Some of the stated objectives of ICs are: more competent users, better information, more positive attitudes toward data processing and enhanced computer literacy.

A knowledge-based system has been developed to support one of the major activities of ICs – software selection. The system called, ICE (Information Center Expert), determines the end user's software requirements by conducting a system-generated dialogue with the user. Based on the user's requirements, and the available resources in the information center, the system recommends software to its users. An additional feature of ICE allows it to track the consultations made with the system, thereby allowing effective follow-ups by IC consultants.

The ICE system was developed in conjunction with ICs at IBM/Endicott, IBM/Tucson and the Center for the Management of Information/University of Arizona. A common characteristic of ICs in each of the three locations is rapid turnover of consultants, but they differ in both the type of users served and the software supported.

In addition to describing the development of the ICE system, this dissertation includes an evaluation of the system based upon two methods of evaluating a knowledge based expert system suggested by Gaschnig, et al. [1983]: (1) evaluation by the domain experts to determine the accuracy of the embedded knowledge and the accuracy of any advice or conclusions that the system provides and (2) evaluation by users to determine the utility of the system.

The validation of the ICE system was conducted by evaluating whether the system provides consistent and acceptable recommendations that are comparable with those of an average consultant, i.e., has recommendation validity. The evaluation also addressed the issue of the IC consultant turnover problem by examining the potential of the ICE system to relieve the consultant work load by handling routine and frequently made queries.

Taking the second suggestion made by Gaschnig et al., [1983], i.e., evaluating

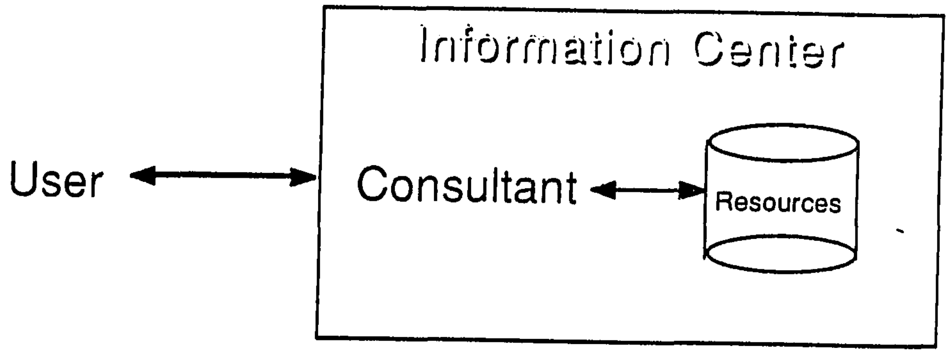

Figure 1.1: Current consulting process

user satisfaction with the system's utility was essential. Validation, though important, is not the main determinant for the successful implementation of a system such as ICE within an organization, because "the criterion of success is whether the expert system is actually used" [Gaschnig et al, 1983, pp. 245]. It is the user's feelings towards the system that will determine the acceptance of the system into the organizational setting, so user's satisfaction with the system must be evaluated.

User receptivity toward change is particularly important when alterations are made to an existing process [Zmud, 1984]. The conventional consultation process for obtaining a software recommendation from the information center is shown in Figure 1.1. Figure 1.2 shows the change in the consulting process resulting from the introduction of ICE.

An important function of an IC in an organization is to provide a way to match end-user needs with the resources available. Currently, consultants perform this matching process, but increased IC consultant workloads limit user access to them. The introduction of ICE provides the end users with an additional channel for getting answers to their software requirements queries.

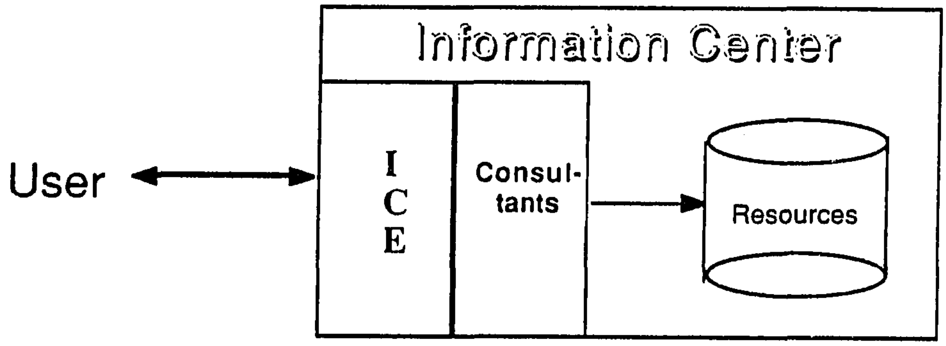

Figure 1.2: Consulting process with the Introduction of ICE

The ICE system therefore can help end-user computing in an organization by giving quick and easy access to software recommendations. With the current implementation of ICE, users can consult with the system in the privacy of their own workspaces. Being able to deal with software needs in private should have a particularly significant impact on new users who might hesitate to use new technology where others might observe them. ICE also enables end users to become familiar with common IC terminology at their own speed.

The research reported here formulated an architecture to develop a knowledge-based expert system for supporting the software recommending activity in an information center. The implementation of this architecture was validated by experts for consistency and completeness of recommendations. Finally, the system was tested to determine users' perception of its effectiveness.

1.2 Statement of the Problem

Prior to the development of the end-user computing concept, application programs were developed solely by data processing department programmers. The introduction of end users into the software development process has caused the application programming responsibility to be shared. In response to this increased involvement of diverse personnel, organizations are using information centers as a framework for facilitating and managing the computing capabilities of end users. A survey by Dickson, et al. [1985], reported that "facilitation and management of end-user computing" was ranked by IS executives as the second most important IS management issue of the 1980s. A follow up on that survey conducted by Brancheau and Wetherbe [1987] reported that "facilitation and management of end-user computing" was ranked sixth among the issues faced by IS management.

A recent study [Brancheau et al., 1985] reported that with the initial successes of information centers, they are being subjected to increased user expectation and higher demand. Further, because information centers are accounted as "cost centers" in the organization, there is growing pressure for ICs to accomplish more with fewer resources.

Rauch-Hindin [1986, pp. 63] suggests that expert systems are particularly helpful in places where "human experts are in very short supply; they are expensive and difficult to get." Expert systems, drawing on a knowledge base representation of human expertise, solve problems with a limited scope that previously required personal attention. The self-learned rules and heuristics of a domain expert, built into an expert system, can be used to produce solutions comparable to those arrived at by a human expert.

1.2.1 Architecture

This research provides a first step toward developing an automated support for the facilitation and management of end-user computing. The system developed uses expert system concepts to acquire and represent the expertise of IC consultants. The importance of the research lies in its provision of a general architecture for managing IC software resources. The prototype was developed for information centers in organizations, but it can be applied to similar advice-giving situations and can be used for the management of resources in other domains.

Expert system development is expensive, so the possibility of transporting and adapting these systems to other similar situations is a desirable quality. Generally, expert systems are very domain specific i.e., once implemented and validated, the systems become environment dependent for user interface and the validity of recommendations. The ability of an expert system to be transportable depends on the concepts of homeostatic processes [Little, 1986] and knowledge stability [Krcmar, 1985]. Homeostatic processes are processes that retain their stable characteristics in changing environments and knowledge stability is the change over time of the knowledge to be represented. The architecture described provides a framework for a knowledge-based system that would be transportable. Further, the architecture incorporates adaptability by allowing the representation of decision biases of experts in a site-specific manner.

The research questions addressed in the development effort of the system that can assist end-user computing in the context of information centers are as follows:

1. What are the architecture requirements for a knowledge based expert system to model the expertise of information center consultants?

2. What are the design requirements for responding to the changing

software resources of information centers with minimal changes to the knowledge base?

1.2.2 System Validation

For ICE to gain acceptance in an organization, it is important to prove the correctness of its recommendations to both users and IC consultants. The users will not use the system unless they have confidence in the recommendations made. Without user cooperation, the system is useless. IC consultants, the alternative source of expertise, must be satisfied with the system if they are to recommend its use to their clients. Recommendation validation is therefore crucial to the success of the system.

The research conducted included the testing of ICE for recommendation validity. Yu, et al. [1979], points out that it is difficult to determine what constitutes expert behavior. Although it is therefore difficult to validate expert systems without first having validated the expertise of the evaluators of the system, there is consensus that the development of an expert system must be followed by its validation.

Two criteria are important for validating ICE. The first is the recommendation validity of the system. It should be able to prove that the advice given by an advising expert system is similar to the advice that would be received if an expert were consulted.

The second criterion, which is related to the first is the completeness and consistency of the knowledge base. Recommendations made by the system not only must be correct but also must be based on a complete set of the facts and correct reasoning based on those facts. The experts involved with the construction of the system described here were concerned with this validation.

The research addressed the stated concerns by dealing with the question:

3. Is the recommendation given by ICE consistent with that given by an average consultant in the information center?

1.2.3 Change in the Process

The consulting process in information centers requires the consultants to have a broad range of skills. Technical competence must be accompanied by effective communication skills. Furthermore, consultants should be able to adapt to both the changing needs of the clients and changes in the resources supported. Finally, they should be able to adjust to the different levels of expertise of their clients.

The designers who set about to introduce ICE into the IC consulting process recognized these facts. ICE was never intended to be a substitute for the IC consultant, but to serve as a source of specialized task – software recommendations. The AMA report [Bohl, 1986] suggests that consultation related to software selection, mainly routine in nature, accounts for a majority of consultants' time. It is in answering these queries that the system will be most useful. If ICE deals with the routine queries, consultants will have more time to deal with exceptions.

The implementation of any system "invariably involves change on part of the users" [DeSanctis and Courtney, 1983, p. 732]. Such is the case with the introduction of ICE. Users will have an additional channel to get recommendations for their software requirements, but they will need to adapt themselves to dealing with the computer for solutions. The IC consultants will be available, but only to handle problems that are not routine.

The research attempted to answer the following question by conducting a laboratory experiment:

4. Is the consultation effectiveness using ICE comparable to the consultation effectiveness of the current consulting process using consultants?

1.2.4 Effect on the Users

Information systems are designed with the goal of providing selected classes of users the best possible information to meet their requirements. The utility of the information presented to the user is dependent on the user's perception of the data [Korfhage, 1985].

With the claim made that ICE supports end users, it is important to test the usefulness of its information to different categories of users. End users are not a homogeneous group, and can vary substantially in their level of expertise. Six categories of end users presented by Rockhart and Flannery [1983] represent the entire spectrum of end user expertise, but for the purposes of this dissertation, we concentrate on two, namely: Non-Programming End Users (NPEU) and the End User Programmers (EUP). One outcome of the research reported here was further clarification of the classification of end users.

The experiment attempted to determine the class of end users that should be targeted for any future automated support. The research question focused on:

5. How does the utility of the information provided by ICE differ between end-user programmers and non-programming end-users of the information center?

1.3 Problem Boundary

Information centers undertake to offer several forms of assistance to help end users meet their needs. This study focused on the task of assisting end users in selection of software for particular tasks.

The ICE system is currently implemented at three geographically dispersed locations. Users have ranged from application programmers with extensive skills in the use of computers, to engineers and financial analysts who use computer packages to do their jobs more efficiently and effectively, to students who use software packages for course related assignments. This study was restricted to the CMI at the University of Arizona and its users. Because the CMI serves the business school, the majority of its users are students in that school.

Hundreds of software packages are available. It is impossible for any organization to have access to all of them, and information centers have been given the job of identifying and acquiring appropriate software for their organizations. The present study was restricted to the 31 software products that are currently supported by the CMI. These products are divided into seven categories: (1) data management, (2) data analysis, (3) document preparation, (4) project management, (5) graphics, (6) utilities and (7) integrated packages. Some software products are placed in more than one category because the software can perform multiple tasks.

Another boundary on the research was established by limiting the categories of end users. The study addressed only two, designated as NPEU and EUP. The NPEUs were MIS majors at the sophomore/junior level, who were expected to have an elementary understanding of computers and to have taken at least one introductory computer course. The EUPs were also MIS majors, primarily at the junior/senior level. These students were expected to have a good understanding of computer concepts and to have taken a minimum of three computer courses. No other subclassification of end users was attempted. Based on classification questionnaire developed for the project, the research identified and described three factors that were used to determine the category of end users.

ICE was developed on the IBM-4381 computer, since the IBM/VM provides the environment required for implementation of the system. The knowledge base was developed for shared-segment implementation, a necessary characteristic if users are to be able to access the system from virtually any location having a networked terminal.

Chapter 2

Review of the Literature

2.1 Introduction

The literature review is divided into three sections. The first examines work related to end-user computing and the organizational setup required to deal with it, namely, the information center. The second section discusses prior work on expert/knowledge-based systems. The third section focuses on the literature on empirical evaluation, i.e., validation and verification of knowledge-based systems.

2.2 End-User Computing

As computing power becomes relatively less expensive and more easily available, systems which are developed and operated by the ultimate users are becoming increasingly popular. Initially, organizations adopted computer technology because of its efficiency in handling routine data processing functions. The introduction of computer users (end users) into the process of developing computerized systems has changed the focus of computing in organizations. When end users can develop software systems to meet their own requirements, data processing in organizations is transformed from "a supply-driven to a demand-driven function" [Atre, 1986, pp.

159].

End-User Computing (EUC) is defined as: "Use or development of information systems by the principal users of the systems outputs, or by their support staff. In effect self sufficient use of computer technology by business professionals to improve productivity" [Brancheau, Leitheiser, and Wetherbe, 1987, pp. 1].

The concept of EUC and its usefulness to the organization is gaining increased recognition. The EUC concept is projected to have a major impact on organizations, in some instances affecting their survival [Henderson and Treacy, 1986]. Both the users of computing technology and their managers are recognizing the productivity gains made possible by the use of EUC tools. With this increased interest in the EUC approach, changes in the corporate computing environment are manifesting themselves [Bohl,1986]. Traditionally, the Data Processing/Information Systems (DP/IS) department was in charge of corporate computing facilities, but the availability of increasing numbers of personal computers in the work place has enabled users to have greater control over their computing environment.

Organizational response to this changing environment is varied. In some organizations, the EUC concept is being accepted with great optimism, which is translated into full support from the DP/IS department for enhancing the EUC concept. In organizations opposed to the EUC concept, even though end users are often at odds with the DP/IS department, the increasing numbers of end users force the DP/IS department to deal with the situation. In the latter kind of situation, the DP/IS department serves in the role of advisor to the end users while maintaining responsibility and control over the larger computing systems.

The concept of EUC can be traced back to the early 1970s. In the beginning, the idea of EUC often overlapped with distributed data processing [Davis and Wetherbe, 1979]. During this period, the EUC literature concentrated on simply categorizing end users [McLean, 1974, Codasyl committee, 1979, Martin, 1982]. The research, however, failed to provide empirical evidence for classifications of end user computing or the structures and processes of end user computing itself. It was only in the early 1980s that end user computing received increased attention from the research community and began to be recognized as a significant component of electronic technology. Rockart and Flannery [1983] conducted a comprehensive field study of end users in the corporate environment. Their work helped develop the most comprehensive classification of end users and, in conjunction with a field study conducted by Benson [1982], forms the basis for the current literature on EUC.

EUC technology delivers some very consequential benefits to organizations, but it also introduces several new concerns for persons using and supporting computer technology in organizations. The benefits of EUC, extensively documented in the literature, can be summarized as: (1) increased efficiency in accessing information, (2) improved effectiveness of decision making by the users and (3) lower development costs [Rockart and Flannery, 1983, Benjamin, 1982, McLean, 1979, Benson, 1982, Leitheiser and Wetherbe, 1986, and Cheney et al., 1986]. The literature also warns of potential risks. The consequences of turning over an organization's computing control completely to end users can be disastrous. The disadvantages of EUC have been described as: (1) system/software incompatibility, (2) loss of data integrity, (3) reduced data security, (3) duplication of effort and (5) under/mis-utilization of the computing hardware and software [Alavi and Weiss, 1986, Danziger and Kraemer, 1986, Brancheau et al, 1987, and Guimaraes, 1985].

Rapid increase in the use of computers in organizations at the level of "user developed and operated software" [Rockart and Flannery, 1983] or end-user computing (EUC) is being observed. Computer technology made its initial impact on organizations by providing efficiency and effectiveness, primarily in the handling of the routine data processing function. The technology has come a long way from those beginnings. For better or worse, it has made a significant impact on the way organizations conduct business. There are conflicting yet powerful images linked to this technology. Some view the technology as "a great problem solver, producing important gains in the efficiency and effectiveness of people in their work...(in the) contrasting view, (the) computer is a problem generator–an expensive disruptive technology that has often failed to match its promise...generated many negative effects for people who use it...seems uncontrollable by these end users" [Danziger and Kraemer, 1986].

Rockart and Flannery [1983] declare that EUC is a concept whose time has come. As such it is a concept which is fairly broad in its perspective if one accepts such a definition as "user developed and operated computing" [Ibid, 1983]. Because end users exist at several levels, both in terms of their expertise and their needs, putting all end users into one generic category with the expectation of having a large and fairly homogeneous group would be unreasonable. A taxonomy on the concept of end users would help identify end-user subgroups, thereby enabling us to deal with specific aspects of EUC.

2.2.1 Categories of End Users

EUC literature provides examples of three different approaches to defining an EUC structure. The Codasyl committee [1979] on end-user facilities presents a three-level classification of end users, basing classification upon the degree of user interaction with the computer. The three levels identified are indirect end users, intermediate end users and direct end users. The indirect users are one step removed from interacting with the computer but are direct beneficiaries of communication with a person who is a direct user. Direct users use the computer in performing their jobs. Intermediate users, while not clearly identifiable, lie somewhere along the continuum between direct and indirect users [Codasyl, 1979].

McLean [1974] provides a second classification of end users. His categories, which he designates as DP professionals, DP amateurs and non-DP trained users, are comparable to the direct, intermediary and indirect users of the Codasyl classification, but he distinguishes non-DP trained users as being able to be "code users," unlike the indirect users in the Codasyl classification, who operate through a link. McLean maintains, however, that the non-DP trained user does not know anything about programming.

The most comprehensive end-user classification to date is provided by Rockart and Flannery [1983]. The categorization they provide moves away from three-part classification by concentrating on the intermediate end user (of the Codasyl classification). They present six categories of end-users: non-programming end users, command level end-users, end-user programmers who utilize both command and procedural languages directly, functional support personnel, end user computing support personnel and DP programmers. Their study further points out that 84% of the end users fall in the first four categories and their more detailed subclassification of this largest set (in terms of number of users in the category) clarifies the EUC picture significantly.

Classification of end users makes it possible to determine who might be the prospective audience for possible technology improvements. The diversity in the end-user population can be of concern to management because it gives rise to the need for providing "multiple software tools...and a need for strongly differentiated education, training and support" [Rockart and Flannery, 1983]. Management is further concerned with bringing together the power of the various end-user computing applications that may be being developed, especially in regard to "incompatible computers, (and) untrained users accessing corporate data" [EDP Analyzer, 1984]. The big debate in EUC revolves around balancing end-user control and end-user creativity.

2.2.2 Categories of Tasks to Be Performed

Just as end users themselves need to be classified, so do the tasks that they perform.

The systems or applications being dealt with by users can be divided into three distinct types [McLean, 1979]:

1. Software for sale
2. Transaction based systems
3. Management information systems (MIS)

The first two categories cover most of the traditional DP applications. Software for sale is primarily the output of software development houses and includes compilers, utilities and other proprietary software packages. Transaction based systems comprise the majority of systems that are supported by DP departments. These are the systems which deal with large volumes of data associated with functions such as payroll, inventory management etc. The majority of these systems are batch oriented.

The EUC applications come under the MIS category, a broad classification within which EUC applications cover systems that are developed to deal with specific user needs. In most cases these systems are single-use systems. Most Decision Support System (DSS) applications fall into the EUC classification of systems if the users purpose is to support a particular decision-making activity. The important point of distinction between EUC systems and the transaction based systems is "the discretionary nature of these (the MIS) systems" [McLean, 1979]. McLean further classifies EUC systems as adversary development, or cooperative development based on the development process. Adversary development is the more typical of the process followed by traditional DP application development. The adversary aspect comes into play because there often is a discrepancy between the user's definition of his requirements to the analyst and his actual needs, resulting in his subsequent dissatisfaction with the system that is delivered. Cooperative development results when the user and the developer share a common understanding of the requirements. When the user participates in the development, the problem of defining the requirements is eliminated.

As improvements are made in both the hardware and software aspects of computer technology, it becomes increasingly apparent that their full potential will not be achieved unless they are truly accepted by the supposed benefactors of the technology, the end-users. Manufacturers therefore have introduced numerous software packages to encourage user acceptance by making the benefits to the user obvious – data base management systems, word processing, spreadsheets and various decision support tools. Concepts such as "user friendliness" and "system usability" are now embraced by both system developers and system users. It is from this background that we see an emergence of end-user computing as a formidable force in future organizational computing. Statistics indicate that EUC applications are increasing at a rate of 50%-90% compared with a growth rate of 5-15% in traditional DP applications [Rockart and Flannery, 1983]. EUC facilitation was rated by MIS executives, consultants and researchers as second in importance only to MIS planning [Dickson et al., 1984].

The growth of EUC is coupled with two other phenomena that are starting to affect organizations. First, personal computers (PCs) have made possible desktop computing. Second, as a result of the tremendous growth in computer applications, DP staffs are overloaded as they try to deal with substantial workloads and keep up with a request backlog. Given this situation, the responsibility for developing lower priority systems (lower in priority from the organization-wide standpoint) which could conceivably enhance productivity is being shifted to the end users. As the

DP workload and the subsequent backlog continue to grow, the importance of end user development of systems is certain to rise.

2.2.3 The Information Center Concept

The information center concept was introduced by IBM-Canada in 1974 when it was decided to experiment with the possibility of end users generating their own systems using fourth-generation languages and PCs. The goal was to reduce the workload of DP personnel [EDP Analyzer, 1987], with the organization playing a supporting role by bringing together the end users and the appropriate technology. The objective of an information center was described as to "provide users access to data on their own terms so that they can solve their own business problems" [Hammond, 1982, p. 133].

An information center has been described as an organization specifically designed to produce "guided service to help users help themselves" [Leitheiser et al., 1986]. ICs commit information systems resources and people–both end users and management–to an information support theme. Further, ICs enhance the EUC approach within organizations by providing end users a focal point to which to address their queries. Having been given the responsibility for support of end users, ICs have been given responsibility for developing organization-wide EUC standards. The role of ICs is thus complementary to that of the traditional DP/IS departments, enabling the IC to work in conjunction with the DP/IS department to provide the user community the widest exposure to the corporate data and computing resources and still to maintain a centralized control.

Information centers are expected to provide several services to end users. These services can be summarized as [Vinze et al., 1987]:

1. Consultation: Work with end users to help them analyze their problems and clarify their needs for computing resources.
2. Training: Function as a center for learning about software and hardware products.
3. Technical Expertise: Provide assistance for the user in selecting hardware and software.

A survey reported in the 1986 AMA Report on Information Centers indicates that 91.5% of these centers evaluate software for end-users on a daily basis. "Requests for assistance in hardware and software selection come thick and fast, and require a matchmaking role between the end user's requirement and the capabilities of the technology" [Bohl, 1986].

It is predicted that by 1990, end-user computing will represent as much as 75% of the total computing capacity of the typical American corporation [Benjamin, 1982]. There is also a consensus suggestion that the best general strategy for managing end users is to give them the computing tools, establish standards, provide data resources, and encourage good computing practices [Royksund, 1987]. Therefore the IC concept in some form is likely to play a significant role in directing the future of computers in the organizational context.

The IC concept is in part an attempt to minimize the various risks to which the organization is subjected in EUC system development. Dealing with these risks, which were discussed previously, has been made the responsibility of ICs by providing the monetary resources for EUC in the organization as part of the ICs operational budget [Alavi and Weiss, 1986]. The consultants or end-user tool specialists in ICs deal with a number of problems and serve as the primary interface between the organization and its EUC activities. The consultants' tasks range from conducting machine familiarization sessions to making software and hardware recommendations to providing the conceptual design of systems for the users. In effect, they are essentially involved in the first two stages of the EUC development life cycle. The job of the consultant is crucial to addressing the issues related to incompatible tools and inaccurate definition of problems. Consultants must ensure that under the "pressure of daily activities, the end-users may not spend sufficient time on problem definition and diagnosis...(since) due to a lack of training in analysis and modeling, end-users are likely to undertake inefficient search and analysis procedures" [Alavi and Weiss, 1986].

Most IC staffs are very small. In a survey in the Minneapolis area, the average IC staff for firms whose enterprises employed more than 20,000 had fewer than six persons [Leitheiser and Wetherbe, 1986]. The average start-up staff for an IC, in organizations whose budgets ranged from under $10 million to more than $1 billion, ranged from 3.1 to 5.3 full time employees, and from 0 to 2.8 part time employees [AMA Report, 1986]. A recent study [Brancheau et al., 1985] reported that end users expect to be even more dependent on the IC in future than they are now, and that they anticipate needing more support services and training.

As information centers achieve a growing acceptance, they are being subjected to increased user expectations, higher demand for services and a growing pressure to accomplish more with fewer resources. A major problem that ICs face is a higher than average personnel turnover rate. This is due to attractive combination of skills–technical competence, communication, problem solving– that the IC staff must have to do their job successfully and the high visibility staff receives in the end-user community.

2.3 Knowledge-Based Systems

Knowledge-based systems research has its roots in Artificial Intelligence (AI) research. The development of cybernetics marked the starting point for AI research which focused on developing theories for activities of the human mind [Wiener, 1948]. The development of computers and easy access to them fueled scientists' interest in exploring the possibility of using electronic calculation power for symbolic processing [Minsky and Papert, 1969; Feigenbaum, 1984; and Rodgers, 1983]

The expert system component of AI research focuses on codifying a human expert's knowledge of problem solving in a narrow domain of interest. Since human experts are in short supply, they are expensive and difficult to get. Although there now are only a few documented successful expert system implementations, businesses are recognizing the potential commercial value of expert systems, both in terms of dollars and the competitive advantage they can provide [Sviokla, 1986a, Rauch-Hindin, 1986]

The literature for expert systems research can be classified into three categories [Sviokla, 1986a]: 1. expert system building, which focuses on the development aspect [Hayes-Roth, Waterman, and Lenat, 1983], 2. theoretical issues [Shortliffe, 1976], and 3. commercial applications for expert system technology [Winston and Prendergast, 1984]. The focus of the present discussion will be on expert system development. The terms knowledge-based system and expert system will be used interchangeably.

As with any new technology, there is a substantial lag between development in research laboratories and "real world" applications. This is reflected in the range of definitions for knowledge-based systems from "a sophisticated program" to "a form of intellectual cloning" [Davis, 1984].

The majority of problems dealt with under the realm of expert system technology in present implementations address a very specific problem in a very restricted problem domain. An expert system is defined as "a computer program using expert knowledge to attain high levels of performance in a narrow problem area" [Waterman, 1986, pp. 11]. Expert systems are useful and efficient because they enable non-experts to solve their problems as an expert would, with the system asking leading questions directed toward finding a solution [Thompson and Thompson, 1985].

Several approaches to expert system development are discussed in the literature. The traditional approach to development of an expert system has four stages [Waterman, 1986, Pople, 1984, Barr and Feigenbaum, 1981]. The first of these is the identification of an expert in the area of interest. The person identified is known as the domain expert. The selection of appropriate experts is critical to development of the system [Prerau, 1987, Davis, 1979]. The second stage, knowledge engineering, consists of the acquiring of knowledge from the domain expert and representing it in machine understandable form [Hayes-Roth, Waterman, and Lenat, 1983, Davis, 1986, Sviokla, 1986b]. Knowledge engineering is an iterative process involving reformulation, redesign and refinements to the conceptualization, formalization and implementation of the system [Hayes-Roth, Waterman, and Lenat, 1983]. The third stage involves the identifying of the expert system development tool. A useful categorization scheme for expert system development tools classifies the existing tools into: general purpose programming languages, general purpose representation languages, and skeletal systems [Barstow et al., 1983]. This development tool differs from conventional programming languages in that it provides both the means for representing the knowledge acquired and an environment in which to build the expert system [Waterman, 1986]. The final stage, interface design, is used to allow the convenient use of the expert system by the user [Cole et al., 1985, Waterman, 1986].

Another approach to expert system development is based on "the extent and depth of the explicit representation of knowledge" [Bobrow et al., 1986]. This classification is separated into "the low road, the middle road and the high road" [Brown, 1984]. The low road deals with the intelligence being placed in the programming environment itself. Systems in this group are coded up on "intelligent machines" using LISP or LISP-like languages as the medium. DENDRAL is often given as an example of a successful implementation using this approach.

Middle road approaches characterize a majority of the applications being developed today. These involve "explicit representation of knowledge, but though some direct programming may be used, most of the interesting behavior of the system is governed by knowledge articulated by experts and represented explicitly in a knowledge base" [Bobrow et al., 1986]. One of the most successful system implemented using this approach is MYCIN.

Knowledge required for building effective expert systems is often talked about at two levels, deep knowledge and surface knowledge. Surface knowledge can be described as "empirical associations but are sometimes 'compiled' from an understanding of structure and function" [Michaelsen et al., 1985]. Deep knowledge, on the other hand, takes into account the underlying theories and principles, allowing an enhancement of the explanatory powers.

The low road and the middle road described above deal primarily with surface knowledge. The high road approach on the other hand is more conducive to knowledge representation involving "deep" conceptual models. Very few systems at present attempt the high road approach.

As indicated by the discussion thus far, the three major issues with regard to expert system development, are: 1. representation of knowledge, i.e., finding the machine equivalent of the human memory, 2. Control and use of knowledge, i.e., developing the machine equivalent of the human capability for problem solving and planning, and 3. acquisition of knowledge, or extraction of knowledge from the sources of expertise.

Knowledge representation is often considered by expert system builders as one of their most challenging tasks. With regard to knowledge representation, it is often suggested [Buchanan et al., 1983] that data or facts cannot be equated to knowledge, thus an umbrella of representation has to be placed over such information to convert it into "knowledge."

There exist several techniques for knowledge representation; rules, frames, and semantic nets are a few of these [Barr and Feigenbaum, 1981]. Rules are arguably the most effective way to represent human expert know-how. Experience shows that experts can express their problem solving techniques best in terms of situation-action rules [Hayes-Roth, 1985, Erman, Scott and London, 1984]. Each of the rules in a rule-based system is a piece of the expert's know-how. Rules can also be classified as data which can be used in conjunction with an inferencing mechanism to mimic the reasoning process. There are primarily two inferencing mechanisms: 1. forward chaining or data driven and 2. backward chaining or goal driven. Forward chaining starts with a collection of known facts and tries all of the applicable rules over and over again, adding new facts as it goes along, until there are no more rules that apply. Backward chaining works from the goal backward to find supporting data. In instances where the rules in the knowledge base do not provide sufficient information, the inferencing mechanism queries the user to provide the required input.

Rules are in the form of the conditional if-then statements used in conventional programming languages. The "if" specifies the condition and the "then" the subsequent action. Despite their similarity, the use of if-then rules in conventional programs differ significantly from their application as building blocks in a knowledge base. Rules in the knowledge base context differ from the if-then statements because of their extreme modularity [Rauch-Hindin, 1986]. A set of rules must be able to control themselves during interaction with the knowledge base by knowing when they should be activated. This is achieved by various inferencing techniques.

The knowledge acquisition process is ongoing [Hart, 1986]. Knowledge is defined as a "collection of specialized facts, procedures, and judgment rules" [Turban, 1988, p. 379]. The knowledge acquisition process is therefore the method used for extracting from the domain expert the facts, procedures and rules they use in performing their tasks. Knowledge acquisition is a very troublesome process because human experts find it so difficult to express their know-how [Hart, 1986].

Several methods have been devised to assist in the process of knowledge acquisition: interviews, protocol analysis, observations, and rule induction [Waterman, 1986, Turban, 1988].

Interviewing is the oldest of the four knowledge acquisition methods, and involves a stepwise decomposition of the problem. The aim of the interview is to elicit from the expert his or her mental model of the problem domain [Turban, 1988]. Protocol analysis is a technique borrowed from psychology [Ericsson and Simon, 1984] in which the experts verbalize their thought process as they perform the task. The knowledge engineer then deduces the various facts, procedures and rules that the expert is using. Knowledge acquisition by observation is a deduction process where multiple observations of an expert in action allow the observer to make inferences about the knowledge used.

The rule induction method is particularly applicable where the knowledge representation scheme is rule based. Two possible approaches for using rule induction are: forward scenario simulation and goal decomposition [Grover, 1983]. In the forward scenario simulation, the expert verbally walks through the steps necessary to reach a predetermined goal. The purpose is to gain an understanding of the heuristic approach that consultants use in determining user needs and resource characteristics. In goal decomposition, the aim is to decompose goals into their subcomponents and use this subclassification to construct rules by interpreting the situations and the objects that constitute the goal.

A form of expert system referred to as Expert Support System (ESS) [Luconi et al., 1986] is of particular interest in the context of the current discussion. For an expert system to be accepted by management, it must first be able to support the expert in routine decision making. Although this seems to be getting back to the DSS concept, there is a difference. The role of DSS is to provide management "another source of information on his or her internal and external business environment. Through an interaction and display facility that may include a command and data query language, report writing, the manager can create explicit models of his firm" [Treacy, 1985]. ESS attempts to apply expert system concepts to a much broader range of problems. This is done by embodying the more routine or structured decision processes of the expert into a knowledge base by applying appropriate knowledge representation techniques. The expert supplements the process when exceptions or situations not modeled in the knowledge base occur. Due to the scope of the system, the knowledge captured for ESS may be "imprecise ... and trigged by the evolving problem context" [Luconi et al., 1986]

The knowledge-based systems being considered are primarily decision-making or recommendation-providing systems. There exist several classifications of decisions. One of these is – programmed-nonprogrammed [Simon, 1960]. Programmed decisions are structured and routine as compared with nonprogramed decisions, which are unstructured and imprecise. Another possible distinction divides decisions into structured, semi-structured and unstructured categories [Keen and Scott Morton, 1978]. AI research is moving toward solving unstructured decision making problems, but expert system research in its current form is focused primarily on structured and semi-structured problems. A possible reason for this is that businesses where such systems are most readily applicable have "a fundamentally different set of concerns than the AI researchers" [Sviokla, 1986a, p. 10]. It is relatively easier to evaluate the accuracy and benefits of systems tackling structured or semi-structured decisions, thereby enhancing the chances of success.

2.4 Validation and Verification

To appreciate the usefulness of a system, it is important to test it within the boundaries of its limitations. Accepting the premise that the real goal of application of expert system technology is to allow a non-expert to go through the motions of solving a problem in the same way an expert would, with the system asking the leading questions [Thompson and Thompson, 1985], inevitably leads to recognition of the need for adequate validation and verification.

In comparison with the extensive literature on design and development issues concerning knowledge-based systems, the literature on expert systems addresses the validation and verification concerns rather infrequently. The need for validation and verification is, however, often pointed out as explaining the relatively small number of documented successful knowledge-based system implementations [Green and Keyes, 1987, Sviokla, 1986b, Lane, 1986, Gaschnig et al., 1983, Yu et al., 1979].

2.4.1 Validation

The validation of knowledge-based systems is defined as "substantiating that a system performs with an acceptable level of accuracy" [O'Keefe, Balci, and Smith, 1987, p 82]. This is also referred to as "summative evaluation" [Hamilton and Chervany, 1981, Schriven, 1972] because it focuses on the outcomes or the end results. Several approaches to validation have been reported. Decision analysis [Keeney and Raiffa, 1976] used the concept of expected utility to judge the alternatives. The theory of measurable multi-attribute value functions was proposed by Dyer and Sarin [1979], who extended the work of Keeney and Raiffa. Turing tests [Gaschnig et al., 1983, Buchanan and Shortliffe, 1984] have been used extensively in the evaluation of MYCIN.

The validation of knowledge-based systems literature is spread along the dimensions of: what to validate, instruments for validation, and techniques for controlling bias [O'Keefe, Balci and Smith, 1987]. The question of what is to be validated is determined by the stage of development of the system [Gaschnig et al., 1983, O'Keefe, Balci and Smith, 1987]. Validation can be performed for either the final outcome of a session with the system, the reasoning that accompanies the process or both the conclusions and reasoning as the situation warrants [Gaschnig et al., 1983]. There is consensus in the literature, however, that validation that takes the form of designating an outcome as correct or incorrect is an oversimplification [Kulikowski and Weiss, 1982]. A more acceptable form of validation is by providing the experts several categories into which they can classify the outcome, for example: "ideal, acceptable, suboptimal and unacceptable" [Hickam et al., 1983].

The use of case scenarios to facilitate validation of an expert system is producing encouraging results [Scambos, 1986, Gaschnig et al., 1983]. There are, however, some concerns about the use of test case scenarios. One of these is that case coverage may not ensure the exhaustive testing of all the conditions, as well as the testing of combinations of the conditions [Myers, 1979, Boehm et al., 1978, O'Keefe, Balci, Smith, 1987]. It should be recognized that the knowledge-based system cannot recognize scenarios "beyond those for which knowledge is explicitly available" [Lane, 1986, p. 122]. It is further suggested that knowledge-based systems are still very fragile, and that they do not handle the boundary conditions well [Lane, 1986, Davis, 1986].

Another concern related to using scenarios for evaluation is the need for "an objective standard...for generally accepted or correct answers" [Gaschnig et al., 1983, p. 261]. There are two approaches to defining standards [ibid, 1983, p. 262]:

1. The correct answer for a problem (in some objective sense).
2. What a human expert (or a group of them), presented with the same information available to the program, say is the correct answer.

A standard for the evaluation of system validity is not clearly definable for all knowledge-based systems. It is the domain that determines the standards to be used for validating the system. As a result, certain domains allow for a more definitive evaluation than others. In the testing of MYCIN, several eminent physicians were used as evaluators [Yu et al., 1984]. Even this distinguished set of experts in several cases could not agree on an acceptable solution, and in some cases their evaluations showed prejudice and inconsistency [ibid, 1984].

2.4.2 Verification

The need for performing verification studies for knowledge-based systems is well documented [Gaschnig et al., 1983, Harmon and King, 1985, Minsky, 1984, Green and Keyes, 1987]. There is, however, a lack of consensus regarding a definition for verification. There are differences regarding both the perspective from which a verification effort should be undertaken, i.e., the user's perspective or the developer's perspective [Green and Keyes, 1987, Lane, 1986] and the most appropriate extent for a verification study, i.e., continuous evaluation at the various stages of development or a more consolidated effort to evaluate the final product [Adrion, Branstad, and Cherniavsky, 1982, Green and Keyes, 1987].

Verification is defined as a method for evaluating the effectiveness of a process and its contribution to accomplishing the objectives of the system. This definition is based on the "formative evaluation" approach [Schriven, 1972] to assessment of the quality of the knowledge-based system and the support it provides to the process.

Effectiveness is increasingly being considered as a criterion for system verification [Lane, 1986]. It is further being recognized that user acceptance of a knowledge-based system is an overriding issue [Lane, 1986, Mavor and Kidd, 1986].

A framework for studying effectiveness is provided by Card, Moran and Newell [1983]:

system + task + user → system performance

The framework draws on structure variables concerning the system, the task and user characteristics to predict the effectiveness of the system's performance. Moran [1981] provides a further discussion of evaluation standards that are applicable for system verification studies [Leitheiser, 1986]:

1. System evaluation - effects of specific system performance
2. Feature evaluation - effects of specific features
3. User factors - effects of human characteristics

The conceptual hierarchy of objectives [Hamilton and Chervany, 1981, Greenberg et al., 1976] views the verification or the evaluation of system effectiveness in two parts: efficiency and effectiveness. Efficiency is concerned with development of the system for use by the end users, and effectiveness is the ability of end users to use the system for accomplishing their goals. It is therefore the effectiveness aspect of the conceptual hierarchy of objectives that is applicable to the verification of the knowledge-based system.

A third option for knowledge-base verification is on the basis of system usability [Damodaran, 1984, Eason, 1984]. The usability concept studies the utility that a system can provide its users. The study of usability focuses on a multivariate causality. The usability concept requires that all variables in the system-task-user context must be considered in order for the results to be generalizable. As a result, this approach focuses primarily on field studies.

The difficulty in performing verification of knowledge-based systems using effectiveness as the basis is a lack of validated instruments. Several studies have been conducted [Bailey and Pearson, 1983, Ives, Olson, and Baroudi, 1983, Neumann, and Segev, 1980, Debons, Ramage, and Orien, 1978] and have resulted in the development of instruments to study the various constructs composing system effectiveness. These instruments were designed for the traditional information systems, however, and as a result do not account for a number of the features that are presented by knowledge-based systems.

User satisfaction is often considered the largest component in determining the success or failure of a computer system [Powers and Dickson 1973]. The majority of instrument development efforts directed toward testing effectiveness consequently have been focused on this factor. At the same time, it was recognized that the instruments developed may not be applicable to all computer based systems, especially those used for "disparate, relatively unstructured, ad hoc decisions" [Ives, Olson and Baroudi, 1983, p. 786].

The evaluation of knowledge-based systems is being recognized as critical to the success of such systems. As a result, while substantial effort is being focused on the development of tools for facilitating the evaluation of expert systems [Scambos, 1986, Bliss, Feld and Hayes, 1986], it is recognized that the uniqueness of the various systems requires that a combination of methods be used.

Chapter 3

Methodology

3.1 Introduction

The research methodology is based on the framework presented by Scott Morton [1984] for research on Management Support Systems (MSS). ICE can be classified as a management support system because it aids the management function of supporting end-user activities in an organization.

Scott Morton [1984, pp. 24] proposed an eight-part research framework: "prototype, construct a methodology, develop a theory, formulate a concept, perform empirical tests, conduct a survey, describe a case, and declare a 'truth'." Some of the categories proposed are not applicable to the research being reported here, which was conducted as a combination of software engineering (system development) and laboratory experimentation. Prototyping, development of instruments, and conducting empirical tests for ascertaining the validity and relevance of the system are therefore the categories of relevance to this dissertation.

3.2 Prototype Development

The research process was initiated by developing a prototype system. The prototype development for ICE was a a three-step process: knowledge elicitation, knowledge representation, and system implementation.

3.2.1 Knowledge Elicitation

Knowledge elicitation was accomplished by interviewing domain experts and using other relevant sources. Domain experts were identified at each of five separate information centers, three at IBM/Endicott, a fourth at IBM/Tucson and the fifth at the Center for the Management of Information at the University of Arizona. Each expert was an IC consultant at one of these locations. The knowledge acquisition process also used other sources such as end users, managers in charge of resource policy formulation, software manuals, internal company documents, and commercial trade journals.

The experts were interviewed extensively, and several consultation sessions were observed to gain an understanding of the process. Because each expert had an area of specialization, it was difficult to judge the scope of system, so other sources of information had to be employed. The literature for knowledge-based system development supports the use of multiple sources of knowledge because "without knowledge, a knowledge-based system cannot be built. There are differing extents to which knowledge is available and obtainable [Krcmar, 1985]."

The following techniques characterize the process used for capturing knowledge for the ICE system:

- Forward Scenario Simulation [Grover, 1983]– Experts were verbally walked through the steps necessary to reach a predetermined goal. This was done by interviewing consultants and observing consultation sessions they held with end users in order to gain an understanding of the heuristic approach that consultants use in determining appropriate end-user characteristics and resource needs.

- Goal Decomposition [Grover, 1983]– experts' support was used to decompose the various consultation goals ence technique. The domain expert consultants, having been instructed that they were being asked to develop sets of resources, were asked to systematically reduce each of these sets or goals, into subgoals and to repeat this process until the subgoals were transformed into heuristic goals which could be implemented.

- User Dialogue Classification– ICE engages the end user in a dialogue in order to determine his/her resource needs. In order to create an effective dialogue, an effort was made to understand the terminology and problem approaches used by the end user. A number of different users were interviewed to determine categories of problems which they might expect the IC to solve.

- Rule Reclassification– Rules developed in each of the previous steps next needed to be reclassified. Additional parameters were defined to refine the knowledge formulated thus far. User dialogue was given increased importance since the aim of the system is to provide them with a meaningful interface.

In steps one and two, two very common software design techniques were applied to expert system development: top-down design and bottom-up implementation. Forward scenario simulation provided an understanding of the overall procedures and methods used by consultants in assessing resource needs. This allowed for a general structure of control to be planned. Goal decomposition permitted classification of user goals into functional areas, which narrows the search space in problem solution.

3.2.2 Knowledge Representation

The choice of a rule-based system was based on the indication in the literature that rules provide an efficient means for representing expert knowledge and "experts tend to express most of their problem-solving techniques in terms of a set of situation-action rules" [Hayes-Roth, 1985, pp. 921]. Further, the problem of software selection being addressed here lends itself to a rule representation scheme "if (certain set of conditions) then (certain recommendation)."

The architectural issues and the techniques developed in the study are discussed in detail in chapter 4. The techniques used address the design issues specific to ICE, but can be generalized to other knowledge-based system development. The focus of the techniques was on facilitating the design and development of "advice-giving" knowledge-based systems. Requirements for developing a flexible, maintainable and transportable system were addressed.

The architecture provided a flexible and maintainable knowledge base by distinguishing between the static and dynamic aspects of the knowledge. The relatively static aspect of knowledge, i.e., the expertise of the various IC consultants, was encoded into the knowledge base. It is recognized however, that this knowledge is not completely static, because the scope of the problem being addressed is expandable. The techniques used for addressing this aspect of knowledge-base development were addressed by partitioning building blocks of the knowledge base, i.e., parameters and rules. Parameters were divided into dialogue control parameters and attribute setting parameters. Two types of rules were used, inferencing rules and monitor rules. The use of this partitioning to provide flexibility and maintainability is discussed in chapter 4.

The dynamic aspect of the system resides in the software resources supported. To provide maintainability, the software solutions were not built into the rules of the knowledge base since inclusion of the software solutions in the rules would necessitate rule additions, deletions or changes for every update to the set of software supported by the system. The software being supported was instead maintained in external files, which are conceptually in the form of relational databases. A maintenance subsystem (MTICE) allows for additions, deletions and updates to these external files.

A "knowledge-base/database" approach was used to assist with the knowledge representation. Such an approach makes it possible to identify the areas in the process that would benefit from knowledge-base techniques and those that can best be addressed by the traditional database approach.

3.2.3 System Implementation

ICE has been developed on the IBM mainframe (4381) at the MIS Department of the University of Arizona. The IBM Expert System Environment (ESE/VM) development shell was used to build ICE. ESE/VM is composed of two parts: Expert System Development Environment/VM (ESDE/VM), used to develop the ICE knowledge base, and Expert System Consultation Environment/VM (ESCE/VM), used to provide end users access to the system. The ESE/VM development shell uses a rule-based knowledge representation scheme. The shell provides editors for building the rules and parameters that constitute the knowledge base and allows for quick implementation of both forward and backward chaining inferencing mechanisms. Facility for conveniently partitioning of the knowledge base is provided in the form of "focus control blocks."

The implementation of the ICE system approached the software selection process as a two step process: (1) profiling the user and his/her requirements and (2) a matching process for the resulting profiles to a profile of the resources available. Resources were defined as the software resources. The elicitation of the user's profile and the requirements profile is handled by the ICE knowledge base. Both the forward and the backward inferencing techniques are used to conduct a dialogue with the user. The knowledge base is composed of 308 parameters, 271 rules, 13 focus control blocks, 25 groups, and 13 screens.

The matching process is handled external to the knowledge base. The matching process is the more structured of the two processes and as such does not require the use of a knowledge based approach. The matching process uses as inputs the user's profile, the requirements profile and the profile of the software resources.

3.3 Empirical evaluation

In addition to the development of the ICE system, the research focused on validation and verification issues for knowledge-based systems following both methods suggested by Gaschnig, et al. [1983]: (1) evaluation by the domain experts to determine the accuracy of the embedded knowledge and the accuracy of any advice or conclusions that the system provides, i.e., validation, and (2) evaluation by users to determine the usefulness of the system, i.e., verification.

3.3.1 Recommendation Validation

The validation study for the system was conducted at the University of Arizona's Center for the Management of Information (CMI), which operates as an information center for the school of business. The goal was to validate the recommendations made by the system by comparing them with the recommendations an IC consultant would make.

A set of 21 cases was used. The set contained three cases for each of the seven categories of software that ICE supports. The cases were constructed to ensure coverage of the conditions addressed by ICE. The cases were evaluated by IC consultants who were involved with the development of the ICE system, but were not members of the group of consultants assisting with the validation process. Having the evaluation done by consultants was decided upon to ensure that the cases represented consultation sessions that are likely to be encountered.

The validation was conducted using a "blind validation" procedure based on the work of A. M. Turing [1950]. Solving the problems presented by the cases was attempted by both the CMI-consultants and by volunteers using the ICE system. Designated experts were then asked to review the cases with the two alternate sets of solutions, with the source of each recommendation being masked. The effectiveness of blind evaluation for controlling the bias of the experts judging the performance of knowledge-based systems has been shown in studies with Oncocin [Hickam et al., 1985]. The results of the ICE validation study are discussed in detail in chapter 4.

3.3.2 Verification of Effectiveness

It is assumed that the purpose of introducing the ICE system into an organization is to reduce the IC consultants' workload without decreasing the usefulness of the IC to the end-user community it supports. Given this assumption, ICE is tested both for the validity of its recommendations and the comparative success of ICE and the IC consultants in matching user needs with the resources of the organization. The validation of ICE using the Turing test has been discussed in the previous section. The following discussion focuses on the laboratory experiment to test the comparative problem-solving effectiveness of ICE and IC consultants using a hypothetical construct, "consultation effectiveness."

The setting of the experiment was the CMI information center, which supports the faculty and the students of the College of Business and Public Administration at the University of Arizona. The use of student subjects in the experiment was deemed appropriate because they are in fact the end-users of the CMI information center. The task provided for the experiment (Appendix A and Appendix B) involved the preparation of a report. Data from a study by Amoroso and Cheney [1987] indicate that report preparation accounts for almost 90% of the consultation conducted by information centers. The preparation of reports is also appropriate for student subjects as it is an area with which they are very familiar.

The laboratory experiment aimed at studying the perceived effectiveness of the introduction of a knowledge-based system from the perspective of end users. Another aspect of interest was to study the response to this technology of the different classes of end-users. To this end, the experiment uses two independent variables: (1) process type–CMI-consultant or ICE, and (2) end-user type–non-programming end user (NPEU) and end user programmer (EUP). The dependent variable is "consultation effectiveness." This construct includes measures for user satisfaction with the process of software selection, as well as other measures for judging effectiveness based on the task assigned and the recommendation obtained from a consultation session.

The independent variable was controlled by making randomized subject assignment to the two processes, ICE and CMI-consultants. The second independent variable, class of end user, was defined on the basis of the taxonomy provided by Rockart and Flannery [1983]. There were, however, no pre-existing validated instruments to assist with user classification. Consequently, a classification questionnaire for distinguishing the types of subjects was developed, using the factors identified by the Rockart and Flannery [1983] study. Chapter 5 reports the validation and reliability checks performed for the questionnaire.

Two instruments titled "effectiveness questionnaires" were developed and validated for measuring "consultation effectiveness," one for the participants who used the CMI-consultants for obtaining a recommendation for their task and the other for subjects who consulted with the ICE system to obtain a recommendation. Chapter 5 presents results of the statistical tests performed for checking the validity and reliability of the instruments.

The experiment was designed as a 2X2 factorial. The choice of this design allowed the use of factorial analysis of variance to analyze the data. This helped separate out the main effects, i.e., the separate effects of the two independent variables on the dependent variable. The joint effect of the two independent variables on the dependent variable also was studied.

A pilot study using eight subjects was conducted to test the procedures used in the experiment. The results of the pilot have not been included in the discussion of the experiment results. No changes in the instruments were made as a result of the pilot, but there were minor changes in some of the procedures that were to be used in conducting the experiment.

The experiment was conducted as two separate studies, using MIS students from the College of Business at the University of Arizona. The results have been analyzed separately. The first study had 96 participants at the start but, for various reasons, 15 participants dropped out. As a result, the effectiveness questionnaire, administered at the end of the first study was completed by 81 participants. The second study had a 100 participants who started and completed the experiment. There were no dropouts. Both the studies were conducted in the fall 1987 semester.

The experiment was designed using the principle of MaxMinCon [Kerlinger, 1986], which ensures an efficient design by focusing on variance control to assist in studying the effects being observed. Stated simply, it requires the experimental design to maximize systematic variance, minimize error variance and control extraneous systematic variance.

Maximizing the systematic variance refers to "variance of the dependent variable influenced by the independent variable or variables of the substantive hypothesis" [Kerlinger, 1986, p. 287]. In this experiment, this is achieved by the choice of subjects. The subjects were enrolled in either the Introduction to Business Programming course, the second course in the MIS course series offered at the University of Arizona, or one of the two courses which can possibly be the final course taken by MIS majors: Data Management, Systems Analysis and Design, or Advanced Business Programming.

The minimization of the error variance refers to "(1) the reduction of errors of measurement through controlled conditions, and (2) an increase in the reliability of measures" [Kerlinger, 1986, p. 290]. To minimize error variance, the experiment was conducted as two studies, the first of which was considered a control for the second. The situation, tasks, and instructions for both the studies were maintained identical to allow the use of the first as the control. Further, two studies made possible a more comprehensive reliability and validity check based on repeated use. This was especially important because the experiment was using newly developed instruments, there being no appropriate previously validated instruments.

The control of extraneous systematic variance refers to the control over "independent variables extraneous to the purposes of the study (being) minimized, nullified, or isolated" [Kerlinger, 1986, p. 287]. Kerlinger [1986] further recommends four methods to control extraneous variance: choose subjects homogeneous on the independent variable, randomization, build variance into the design, and match subjects.

Restricting the subject selection to the MIS department was decided upon to ensure homogeneity among the subject backgrounds. Furthermore, all subjects of a particular category of end users were selected because they were enrolled in a course at a particular level. For example, all end-users of the non-programming end user class were chosen from the Introduction to Business Programming course. This also helped to maintain homogeneity among subjects.

The randomization in assigning subjects to particular treatments was achieved in part through the process by which students are assigned to a particular section of a course. When multiple sections of a particular course were involved, the assignment of a section to a particular treatment, i.e., using the ICE system or consulting with the CMI-consultants, was done on the basis of a coin toss.

In the classification questionnaire, the participants were queried about their grade point average, number of previous relevant courses or exposure to computers, and other factors that could potentially be a cause for a non-programming end user to be classified as an end-user programmer, or vice versa. The data did not show a wide range for the factors considered. Further, to ensure that the groups being compared were differentiable, a t-test was performed for each of the factors identified from the classification questionnaire. Details of the results are presented in chapters 5 and 6. The use of the classification questionnaire allowed possibly extraneous variables to be built right into the design as part of the independent variable.

The use of matching to control extraneous variance, though important, is not wholly appropriate for the experiment design being used. Some of the advantages of using matching were achieved in the existing design by the use of randomization of the subject assignment.

Chapter 4

System Architecture and Validation

4.1 Overview

This chapter discusses the system architecture by focusing on the design issues considered and the system development strategies used.

The ICE system makes it possible to support one of the major activities of information centers, consulting for software selection, by modeling the expertise for software selection from five separate information centers. Three of these are located at IBM/Endicott, a fourth is at the IBM/Tucson location, and the fifth is the Center for the Management of Information at the University of Arizona. The five ICs differ in the type of clients served, as well as the set of software that each supports. The clients served come from a wide range of backgrounds, ranging from expert programmers with extensive computer knowledge to relatively novice students who have never used a computer before.

4.2 Design Criteria for ICE

Two characteristics of knowledge important in expert systems development are context dependency and knowledge stability [Krcmar, 1985]. Context dependency describes the universality of certain knowledge; stability refers to the change over time of the knowledge that is represented. These characteristics dictate two important design considerations for ICE: maintainability, and transportability. A third criterion, flexibility, allows ICE to be adapted to any information center setting with only minimal changes in the knowledge base.

4.2.1 Maintainability

Maintainability, or the extent to which the software supported by the system external to the knowledge base can be updated, is an extremely important issue because software tools are being introduced into the market at a very rapid rate. To stay competitive, an IC must be able continually to adapt to this unstable and dynamic environment.

ICE was developed using ESE/VM, which uses rules as its knowledge representation scheme. In rule-based systems, the inferencing process is accomplished by using either the backward chaining or the forward chaining technique for arriving at any conclusion. The rule structure includes the problem solutions directly into the rules; that is, using IF/THEN statements, recommendations are "hard-coded" into the response portion of the rule. Such a method is not appropriate in the dynamic environment of the IC, where including solutions directly into the rules can cause two problems: 1. the size of the knowledge base would be substantially increased, i.e., the large number of rules required could cause a degradation in the performance; 2. any change in the set of software tools supported by the information center would require a knowledge engineer to reflect the changes by altering (adding/deleting) rules within the ICE knowledge base.

This problem was avoided by defining the software tools supported by the IC in an external data base and using an external search algorithm to match those tools to the current problem definition. This makes it possible for the same set of selection rules to be used to find new software that may offer a better solution to a previous problem/tool match. The maintenance of the software tools supported by ICE external to the knowledge base is controlled by the MTICE (Maintenance Tool for ICE) subsystem. A description of MTICE is presented in a later section.

4.2.2 Transportability

The second consideration, transportability, responds to the fact that knowledge is context dependent. Transportability for ICE means that the same knowledge base can be used in different information centers which support different sets of software. This is an important issue here because no two information centers are alike. An organization such as IBM/Endicott may have more than one IC to support its end-user community, and it is infeasible to develop separate knowledge bases for each of several ICs to meet the requirements of their varied clientele. The system must be adaptable enough to be implemented in different ICs with only minimal changes in the basic rule structures. The maintenance tool for ICE (MTICE) also makes transportability possible. Users at any IC can enter and define their own set of software tools and consultants.

4.2.3 Flexibility

With regard to the ICE system, flexibility is defined as the ability to adapt ICE to any information center setting with minimal changes to the knowledge base. Since rules are used as the knowledge representation method, it is the rules that control the dialogue presented by the system. The development of ICE has provided a method by which changes to the knowledge base can be easily made. This was accomplished by dividing the parameters in the knowledge base into dialogue control parameters (DCPs) and attribute setting parameters (ASPs). This is further discussed in Section 4.3.1.3. The advantage of this flexibility in the ICE architecture is that staff members in different information centers can individualize the system to meet site-specific needs related both to the tools recommended and the dialogue presented to the users.

4.3 ICE Architecture

Using the design criteria discussed in section 2, ICE was conceptually composed of four subsystems: 1. a profiling subsystem (encompassing user profiling and requirement profiling), 2. a requirement-software matching subsystem, 3. a tracking subsystem and 4. a maintenance subsystem (MTICE). Figure 4.1 presents the overall architecture for the ICE system.

4.3.1 Profiling Subsystem

The profiling of users and their requirements is accomplished using ESE/VM (Expert System Environment/VM), is an expert system shell developed by the IBM corporation that is structurally similar to EMYCIN. The development environment provided by ESE/VM allows for the building of a knowledge base of facts and rela-

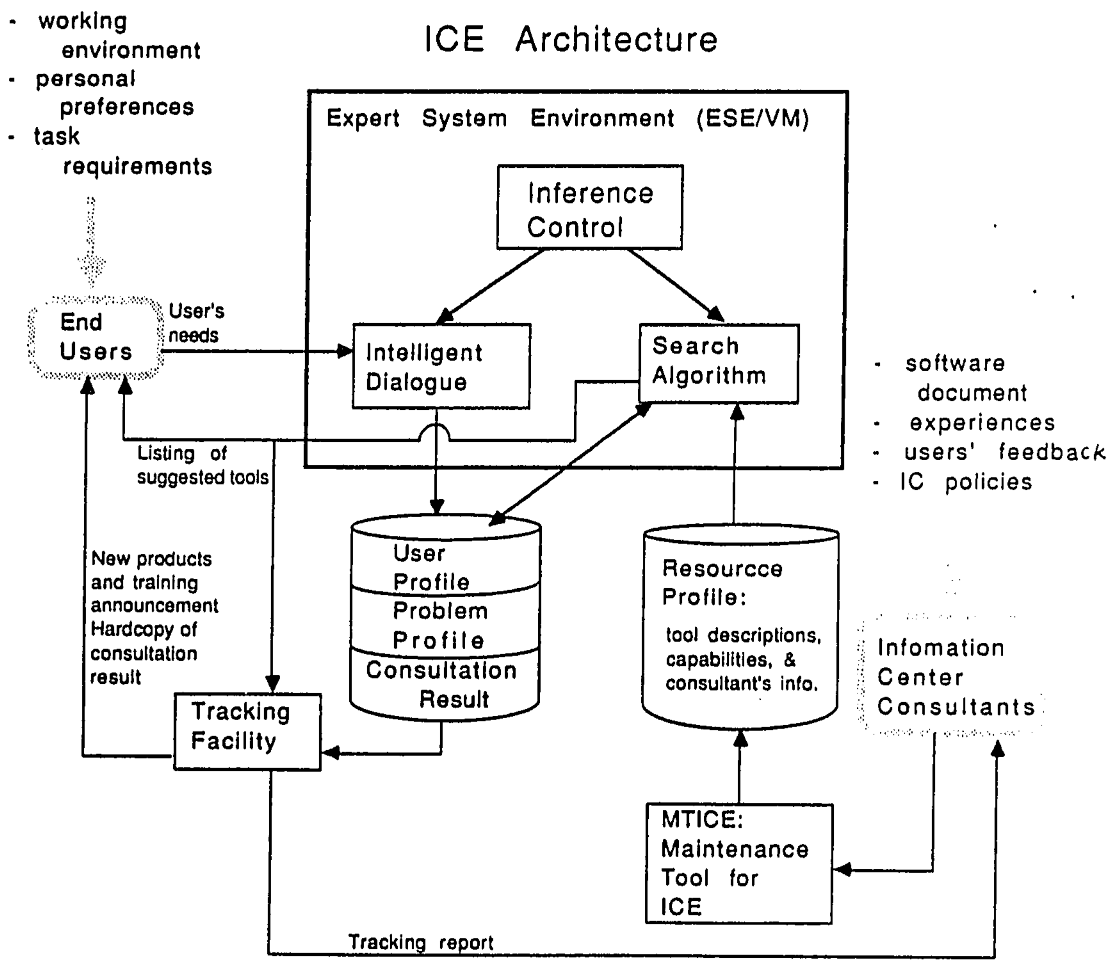

Figure 4.1: ICE Architecture

tionships about users of the IC, the resources supported, requirements of the users and the definitions of the resources themselves. The facts are stored as "parameters," the relationships are represented using "rules" and logical grouping of facts, and their relationship is accomplished by the "focus control blocks (FCBs)."

The profiling subsystem has two components: reasoning control and dialogue control:

4.3.1.1 Reasoning Control

The reasoning control component of ICE controls the process of user consultation. In a standard session of consultation with ICE, the flow of control is as follows: (1) collection of information regarding the user's level of expertise and work environment in terms of computing facilities, (2) analysis of the user's current requirements, and (3) matching the user attributes (background, work environment, and requirements) to the available software resources. The consultation process concludes with software recommendations being made to the user of the system.

Reasoning control in ICE is accomplished by knowledge "chunking" using Focus Control Blocks (FCBs). The use of FCBs allows the production rules to be arranged into logical groupings that are then set up in a hierarchical structure (Figure 4.2).

The parameters of the knowledge base are placed into appropriate FCBs in the same manner. The chunking of knowledge in this way permits different inferencing techniques to be used to meet different needs. For instance, forward chaining is used for accumulating background information on the user, and backward chaining is used for the the analysis of the user's requirement. The use of FCBs also improves the performance of the system, since only a subset of the rule base is dealt with at

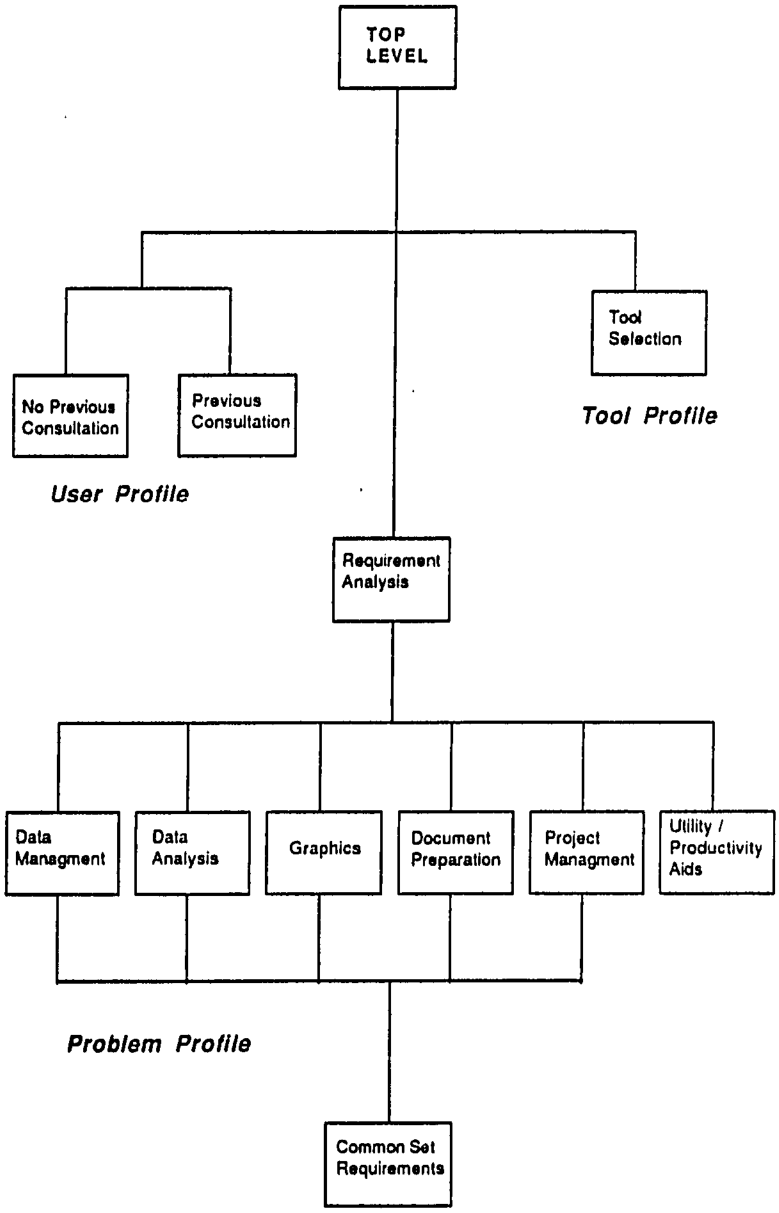

Figure 4.2: Hierarchical structure of the Focus Control Blocks

any given time.

Reasoning control is used to allow a consultation session with ICE to mimic a session with an IC consultant. During the "conversation" between the user and ICE, the user's needs, skills, and knowledge are characterized. Further, through a backward chaining inference process, ICE elicits relevant information about the attributes that describe the users requirements.

A typical consultation session with ICE follows the general format used by IC consultants, based on observations made during the process of system development. When initiating a consultation session with ICE, the user is first asked to provide an identification number, which is then used by the system to check against a user profile file to determine whether the user has previously consulted with the system. One of two conditions may hold true: 1. it is the first time that the user has consulted with the system, thereby activating the No-Previous-Consultation FCB, or 2. the user has previously used the system, in which case the Previous-Consultation FCB is activated.

If the current consultation is the user's first with the system, the user is asked some general information such as: name, department, phone number. Further, the user is prompted to enter information which would aid the system in identifying the user on the basis of skill, previous experience and the preferred computing environment. The list of facts collected is shown in Table 4.1. This set of information constitutes the user profile.

For a user who has previously consulted with the system, the user profile is displayed as it presently exists in the system. If any of the information needs correction, the user is prompted to make any change required.

The use of stereotypes and user-models built from user profiles reduces the

User name	Vinze, Ajay
Department	M.I.S.
Phone number	621-2748
Hardware available	PC used as host terminal
Computer usage	Using software packages
Computer skills	Familiar
Usage Frequency	Frequently

Table 4.1: User Profile

extent of the exhaustive search that must be conducted to determine appropriate software. The user-model makes it possible to eliminate some categories of software from consideration. This is similar to the procedure followed by the experts whose consulting sessions were observed.

After constructing the user profile, ICE queries the user for specific details of the current requirements, prompting him or her to select from one of seven software categories. These were created after an extensive review of the types of software supported by the targeted information centers and are: data analysis, data management, document preparation, presentation graphics, project management, utilities, and integrated packages. Several software tools belong to more than one category.

Once a category is chosen, the system prompts the user to respond to several additional questions designed to elicit information about the software features required by the user. Backward chaining inferencing is used to initiate the dialogue. The process of having a user define specific needs produces the "problem profile."

The third part of the information requirement is the matching of the user and problem profiles with the profiles of the software supported by the ICE system. This is a matching and ranking procedure, and has been described in more detail in a later section.

4.3.1.2 Dialogue Control

Most expert systems have not been developed beyond the research prototype stage [Waterman, 1986]. One reason for this has been the lack of clear and concise interface with the user, which until recently was considered of secondary importance to the design of the knowledge base and inferencing mechanism. Berry and Broadbent [1987] have explained this fact by pointing out that laboratory expert systems have tended to be used by people who love them and are tolerant of their idiosyncrasies. Increasing acceptance of expert systems has made user interface and dialogue control more important issues.

As described in the previous section, the dialogue between ICE and users has three parts. First, the users are asked a series of standard questions to determine their skills and work environments. When the user had previously consulted with the system, the user's profile retained by the system is displayed and an opportunity to make needed changes is provided. Each user profile is built through static dialogue, i.e., the questions remain the same for all circumstances. Appendix I presents a sample consultation session including the user profiling questions.

A second set of questions then is used to determine the needs of the user. After the general category of need has been defined, additional details about the user's requirements are sought. The general categories of user needs (Figure 4.2) were developed after extensive interviewing of IC consultants, and the seven categories cover all the software currently supported by the ICs being operated in IBM/Endicott, IBM/Tucson and the CMI at the University of Arizona. Determining the user's specific needs is accomplished through a backward chaining inferencing process. The querying process is strictly controlled to avoid both redundant and meaningless questions. Details about controlling the dialogue and the techniques used are discussed under query ordering (Section 4.3.1.3).

A third set of questions, referred to as the "common set," follows the need determination. These questions are not need specific, but must be asked for almost every consultation.

Dialogue control refers primarily to the way in which the system and the user interact. Two important aspects are Query Ordering and Screen Layout. These were incorporated after users of the first prototype of ICE expressed dissatisfaction with the interface and the dialogue.

4.3.1.3 Query Ordering

Query ordering for ICE has been accomplished by using two ESE features for controlling the questioning. The first, Focus Control Blocks (FCBs), is structured to permit a logical partitioning of the knowledge base. The concept is similar to the "hypothesis" of NEOMYCIN [Clancey, 1983]. FCBs in ESE/VM allow for the organization of ICE subtasks into a hierarchy in which parameters/rules above a certain FCB are visible to the lower level FCBs, but the FCBs higher up in the hierarchy cannot access the parameters/rules of the lower level FCBs [Hirsch et al., 1983]. In the ICE knowledge base, each FCB represents a subtask. This allows for the questioning to be better controlled, as the inference engine is restricted to a subset of the rules residing in the knowledge base.

In the course of a consultation session, the user's requirement is determined early in the questioning, making it possible to direct the consultation to a relevant lower level FCB. Such an approach reduces the number of parameters/rules that need to be resolved. Because each FCB focuses on a subtask, the questions asked are relevant and make the consultation session more meaningful to the user.

The second method for query ordering has been achieved by dividing the parameters of the knowledge base into two groups: the Dialogue Control Parameters (DCPs) and the Attribute Setting Parameters (ASPs). Parameters in ESE/VM are used to represent domain facts, as compared with rules that represent the relationship among the domain facts in the knowledge base. The division of parameters into DCPs and ASPs has been incorporated to take advantage of two rule types provided by ESE/VM: inference rules and monitor rules. Inference rules are activated by the backward chaining inferencing process. Monitor rules, on the other hand, are a form of "demon" rule. In the case of the monitor rules, the action part of the rule is executed immediately once the premise of the rule becomes true. The inference engine ignores these rules during their processing [IBM, 1986].

DCPs are parameters that define the questions for which the ICE system seeks a response. They can be in the form of multiple choice, boolean, string, or numeric. The backward chaining inference engine is implemented only on the DCPs. ASPs are parameters that define the attributes for the user's profile as well as the user's requirements. It is the ASPs that are passed to the requirement-software matching subsystem to determine the recommendation for the user.

Each FCB in the knowledge base has associated with it a group of DCPs. Once the FCB is activated, the backward chaining inferencing process is used to determine a value for all the DCPs in that FCB. Associated with each of the DCP options is a monitor rule. The monitor rules related to DCPs are of the form "if condition then don't consider certain DCPs." This type of rule further helps reduce the number of rules in an FCB that the inference engine needs to consider. It was earlier pointed out that the use of FCBs helps the inference engine by requiring it to look only at a subset of all the rules in the rule base. ASPs use monitor rules to set values of particular attributes in both the user profile and the problem profile.

The ASPs do not affect the dialogue control process. These parameters acquire value through the use of monitor rules of the form "if condition then ASP1 = 1.0." It is the values of the ASPs that are used in the tool selection algorithm discussed in Section 4.3.2.

4.3.1.4 Screen layout

The screen format plays an important role in user acceptance of a system. Several guidelines exist for constructing an effective screen layout for an interactive system. In developing ICE, the guidelines provided by Cole et al. [1985] were used. These specify four key aspects for screen design: 1. content of display, 2. format, 3. coding and 4. use of color.

Each screen in ICE (with the exception of the user profiling and the final recommendation screens) is divided into four windows (see Figure 4.3). The first window displays the question and answer. The second, or help, window is activated by the PF key (predefined function key), the third is the instruction/warning window, and at the bottom of the screen is the "How, What and Why" window. The standardized format of screens helps users become more familiar with the system.

The default display in the "How, What, and Why" window is "What." "What" provides users with term definitions allowing them to match their understanding of the terms with the system's interpretation of the term.

The format used for responding to the system's questions allows the user to make a single keystroke to prompt an answer. Multiple keystrokes are required only in responding to questions such as asking the user's name. In addition, two features were provided to assist the user during consultation. First, an "undo" feature allows the user the ability to change his/her responses to earlier questions.

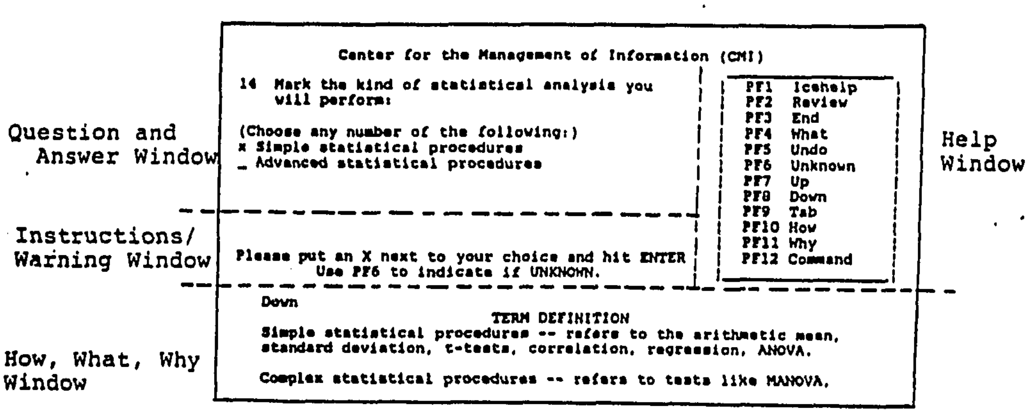

Figure 4.3: ICE general screen layout

Second, an "unknown" feature allows the user the option of not responding to a question. Both these features also are available through the PF key setting.

Color coding is used in each of the windows described above. The question and answer window uses white, the PF key window is green, the instruction and warning window is in red, and the how-what-why window is in blue. The color choices were based on user responses to earlier prototypes.

4.3.2 Software-Requirement Matching Subsystem

Control is passed to the software-requirement matching subsystem once the system completes the query process. As mentioned earlier, the query process uses two types of parameters DCPs and ASPs. The ASPs quantify the attributes that describe both the user's background and his/her requirements. The complete set of ASPs is passed from the profiling subsystem to the matching subsystem.

Upon receiving the ASPs, the matching subsystem activates the selection algorithm for a tool suggestion. The role of the selection algorithm is to rate and rank the various software tools based on the user's background and requirements as defined by the ASPs.

4.3.3 Tool Selection Algorithm

The selection algorithm matches the ASPs passed by the profiling subsystem with the attributes defining the various software tools supported by the IC. This section provides details of the rationale behind the selection algorithm.

IC consultants define the functionalities of tools through a set of attributes. Each attribute is assigned a weighting factor from 1 to 10 according to its importance in tool selection. This weighting factor makes it possible for the biases of an IC to be reflected in the ICE system. The weights defining the importance placed by the CMI staff upon each of the attributes that define the software tools were incorporated in ICE. Attributes assigned weights of greater than 8 were classified as "must-have" attributes.

In the current implementation, each tool's capability is evaluated by assigning a value to the different attributes on a scale from 0 to 10. A capability rating of 8 or higher is considered "very-capable." The attribute weightings and ratings for the tools' capabilities are done using MTICE. During the course of a consultation, users eventually will be able, if they wish, to specify a certainty level for their needs for certain functions. At present, users are given only a binary choice, i.e., they either need a functionality or they don't.

To find tools that can cover all of the user's must-have requirements, the capabilities must be greater than the value of very-capable [Vinze et al., 1987].

For any attribute j,

IF $\mathbf{user\_need(j)} \geq \mathbf{must\_have}$ and

$\mathbf{tool(i,j)} \leq \mathbf{very\_capable}$

THEN

$\mathbf{Rating\_1(i)} = \mathbf{0}$

OTHERWISE

$$\mathbf{Rating\_1(i)} = \sum_{j=1}^{n} \mathbf{user\_need(j)} \times \mathbf{weight(j)} \times \mathbf{min(tool(i,j), user\_need(j))}$$

where:

Rating_1(i): The rating of tool i for First_Choice

user_need(j): user's need for attribute j

weight(j): weighting factor of attribute j

tool(i,j): tool(i)'s capability in attribute j

The confidence level of the tool(i) in First_Choice situation is defined as the ratio of a tool's rating to the rating of an ideal tool:

$$\mathbf{Confidence\_Level(i) = Rating\_1(i)/Ideal\_Rating}$$

In the above formula, the **Ideal_rating** is a tool which is capable of covering all the user's needs:

$$\mathbf{Ideal\_Rating} = \sum_{j=1}^{n} \mathbf{user\_need(j)} \times \mathbf{weight(j)} \times \mathbf{ideal\_tool(j)}$$

where:

Ideal_Rating: The rating of an ideal tool which satisfy all the user's need.

ideal_tool(j): the capability of an ideal tool in attribute j, which is equal to the **user_need(j)**.

Suggestions regarding which tools to use in a given situation depend upon a comparison of the user's needs and the capabilities of the various tools. Depending on the IC's capacity to meet the user's needs with its current tool repository, one of the following three situations will occur:

1. **First-Choice.** As long as there are tools which are capable of covering all the user's must have needs, ICE will list up to nine such tools in descending order of their coverage rates of the user's needs (also called confidence level).

2. **Second-Choice.** When there is no tool qualified to be designated First-Choice, ICE will list up to nine tools, by their confidence levels, that cover a portion of the user's needs. The confidence level in this situation has to be larger than some "cut off" point (called low-threshold) set by the IC consultant.

3. **Last-Resort.** When there is no tool in the First-Choice and Second-Choice categories, ICE will direct the user to appropriate consultants.

4.3.3.1 Confidence levels

A rating scale has been developed to provide users with some indication of the confidence ICE places in the recommendation it makes. On it, the candidate tools are rated and compared against an ideal tool that would perfectly cover every requirement specified by the user. The confidence rating can take on values between zero and one, inclusive, where zero means that one of the desired functions is found in the software tool, and where one indicates a perfect match.

A threshold value is maintained for the confidence levels. Candidate tools falling below the threshold value are eliminated from consideration. The threshold value can be altered by the managers of different ICs to reflect their policy in making a recommendation.

4.3.3.2 Recommendations

A consultation session with ICE is concluded with the system's recommending software tools to the users. The recommendations include name of the software and the confidence level ascribed to it (Figure 4.4).

Confidence levels are used to rank the recommended software, but it is recognized that users may be unfamiliar with the software product named. A user option that allows him or her to choose a software product of interest and browse through a short description of it has therefore been incorporated. The system also has been given the capability of providing the name and phone number of an IC consultant who is responsible for each of the different software products recommended

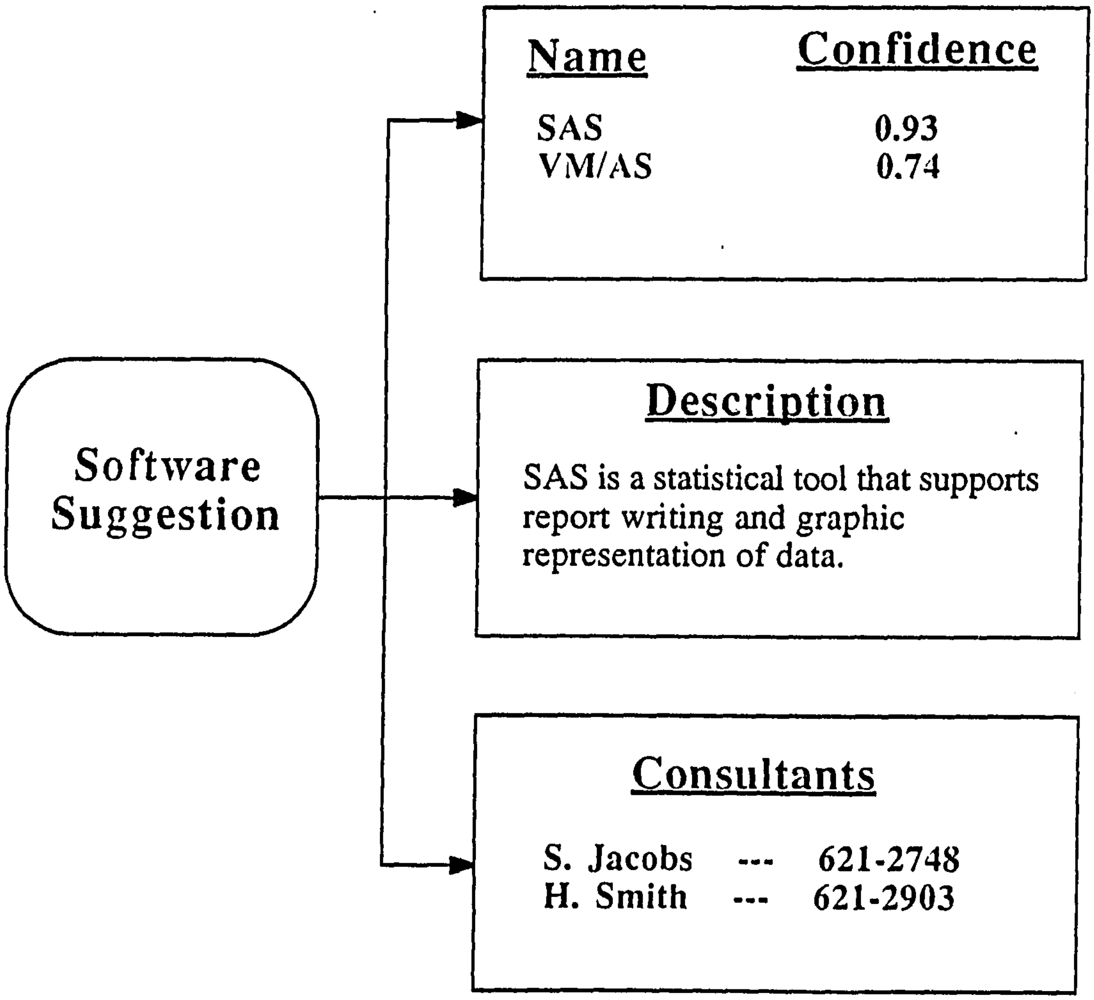

Figure 4.4: Recommendation format

(Figure 4.5).

4.3.4 Tracking Subsystem

The ICE system also has been provided features that help the organization collect data on users of the information center, and on the ability of the IC to meet the computing requirements of its users. A tracking subsystem captures the basic attributes of the user of the system as well as the recommendations made by ICE to each user (Figure 4.6).

This information makes it possible for IC managers to construct a profile of end users in the organization. Most successful ICs have been able to identify key users who develop systems that provide large company payoffs. These ICs have concentrated on helping such users choose application approaches and have provided them with necessary training. An IC that employs ICE can obtain through it information useful in differentiating its user population, providing some services to all and specialized services to certain targeted populations.

The tracking report also enhances information center ability to manage its software resources. The system keeps track of which recommendations made to users regarding use of available software have met: (1) all critical requirements, (2) the majority but not all needs of the user, (3) none of the requirements specified.

The fact that IC records of the situation exist allows the IC to evaluate its software inventory and make effective updates to it. The tracking report can be checked by IC consultants and managers on a periodic basis to determine if the software tools they support meet the needs of the end-user population. If a large number of consultations conclude with situation 1 (the tool recommended met all the critical needs of the user), then the IC is supporting appropriate software tools

```
*------------------------------------------------------------------------------*
|IBM DisplayWrite 4 is a full function word processor with support for many  |
|different printers. It can both accept ASCII, DIF(Lotus), SULK (Multiplan), |
|PCS (Personal Decision Series) and dBase files. This tool has a built in     |
|spell checker as well as providing capability for automatic outlining and    |
|footnotes. Graphics capabilities include a cursor draw. Multiple user        |
|profiles can be created for text and workstations. DisplayWrite 4 provides 4 |
|function math and technical writing support.                                 |
|                                                                              |
*------------------------------------------------------------------------------*
Hit the RETURN key to continue!
```

```
*------------------------------------------------------------------------------*
|Please contact the following consultant(s) for further information:          |
|                                                                              |
|Name : Kendall Cliff                                                          |
|Phone number : 621-2903                                                       |
|                                                                              |
|Name : Yi-Ching Liou                                                          |
|Phone number : 621-2903                                                       |
|                                                                              |
*------------------------------------------------------------------------------*
Hit the RETURN key to continue!
```

Figure 4.5: Additional information to support the recommendation

```
07/07/87
12:37:39
                              User Tracking Report
                              ====================

                         Last Name: Johnson
                         First Name: Paul
                         Department: M.I.S.
                         Phone Number: 621-2748

Situation: First Choice
Tool Number            Software Name           Confidence Level
-------------          ---------------         ---------------
PM3                   VM/AS                      1.00
DA1                   SAS                         .94
```

Figure 4.6: Sample tracking report

for its user population. If, however, a large number of consultations with ICE end with situation 3 (no software can be recommended), then the IC management needs to be concerned and must re-evaluate the software resource inventory. Frequency of consultations ending with situation 2 should send warning signals to the IC management indicating that critical user requirements are not being met, and temporary solutions are being used to meet the immediate needs of the users.

The tracking program provides two services to the users of ICE. First, the user is provided with a hard copy of the recommendations made by the system. The user may use the suggestions made by the system or, alternatively, may acquire a second opinion from an IC consultant. Second, since the program keeps track of the users and recommendations made, the user can be forwarded any notices of updates concerning the software recommended.

4.3.5 Maintenance Subsystem for ICE

The architecture of ICE has been designed for relatively easy maintenance, because the stable knowledge is modeled internally in the rules of the knowledge base and the unstable, dynamic knowledge of the tool environment is maintained in external files that are simple to modify. The maintenance of these external files is controlled by a subsystem called MTICE – Maintenance for ICE. This system is currently PC-based, and allows for the creation of the four files necessary to describe the tool resources. These files are discussed in the Tool Profile in Section 4.3.6.1.

MTICE (Figure 4.7) addresses the two design issues of maintainability and transportability. The maintenance of the four files constituting the resource base has been approached by viewing the four files as relationships in a relational database for which the primary keys are the tool identification number and the consultant serial number. The relations have been normalized to the third normal form [Date,

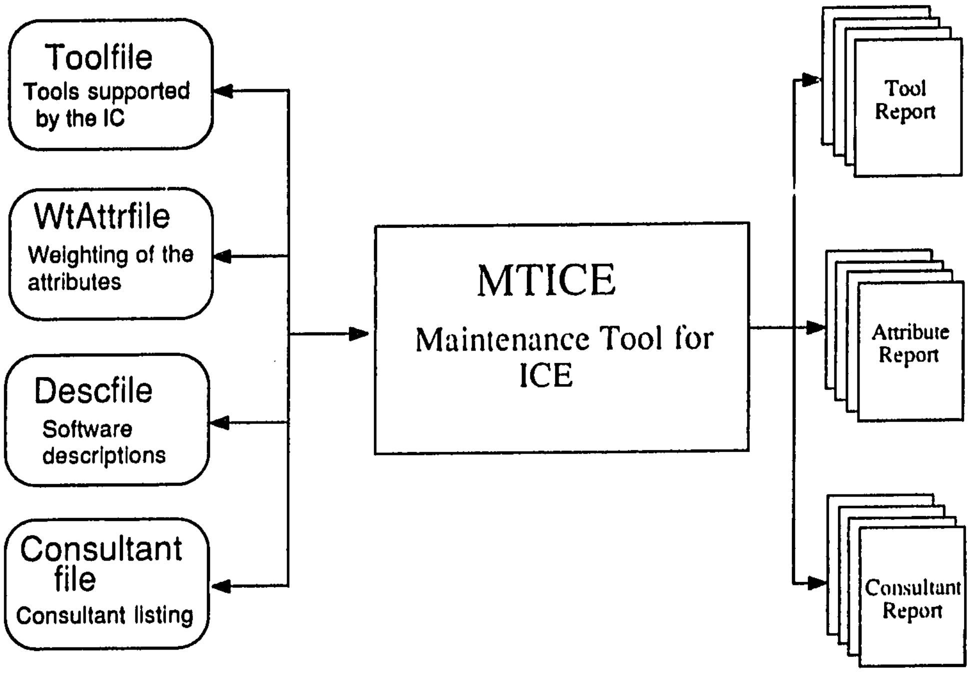

Figure 4.7: Overview of the the Maintenance Tool for ICE (MTICE)

1986] to facilitate insertion, deletion and update. A consistency and completeness check is made each time any of the files are changed.

The transportability issue has been addressed by permitting each IC to be able to maintain its own resource base. Maintenance of the resource base has two parts: first, maintenance of the actual set of tools supported by the IC, and second, reflecting the IC bias regarding those tools. The listing of the tools supported by the IC is maintained in the Toolfile, Descfile, and Consult files (see Section 4.3.6.1). The bias of the IC is built into the WtAttr file and is reflected in the form of attribute weighting.

MTICE also has been provided with a report and browse facility that IC consultants can use to enhance record keeping of the software tools supported. A future modification, currently being developed will expand the browse facility into a IC resource "window shopping" facility.

4.3.6 Knowledge System Programming

Expert systems use several sources to populate their knowledge bases. Values for parameters are acquired from production rules, default values, interaction with users, or external storage. All are appropriate under certain circumstances. In the building of the Information Center Expert (ICE), each of these was used to some degree. The methods can be classified into internal and external methods.

The internal method of acquiring values for parameters in the ESE/VM environment are: (1) the application of the rule base, and (2) the use of default values. The external means of acquiring values for the knowledge base parameters applied in this research were: (1) external storage, and (2) interaction with users.

4.3.6.1 Knowledge Base - Database Issues

Expert systems, in very general terms, are composed of a knowledge base and an inference engine. The knowledge base is a collection of domain knowledge. A database is defined as "a collection of data representing facts. The amount of data is typically large, and these facts change over time" [Wiederhold, 1984]. The major difference between the knowledge base and the database approaches is that a knowledge base contains information at a higher level of abstraction. Facts in a database are normally passive; they are either present or not present. A knowledge base, on the other hand, actively tries to supply missing information [Forsyth, 1984].

Given these definitions, we can say that the knowledge base tries to capture the expertise of the domain expert in the form of rules used by the expert to deal with certain situations. Knowledge relates to the general aspects of the data and, unlike data, it should not change very rapidly over time [Wiederhold, 1984]. The data base, on the other hand, contains values for the parameters that are used to define the rules of a domain expert. Among other properties, databases can be modified efficiently by the insertion, updating, retrieval and deletion of data. Thus, a database could be used as an efficient means of maintaining the values for the dynamic parameters of a knowledge base.

Zobaidie and Grimson [1987] have described a variety of ways in which an expert system might interact with a database system. In an intelligent database, the deductive component is embedded into the database management system. In an enhanced expert system the inference engine of the expert system is provided with direct access to a generalized database. In inter-system communication, an expert system and a database management system co-exist with some form of communication between them. ICE is an example of an Enhanced Expert System.

In the ICE system the knowledge base is divided into three primary compo-

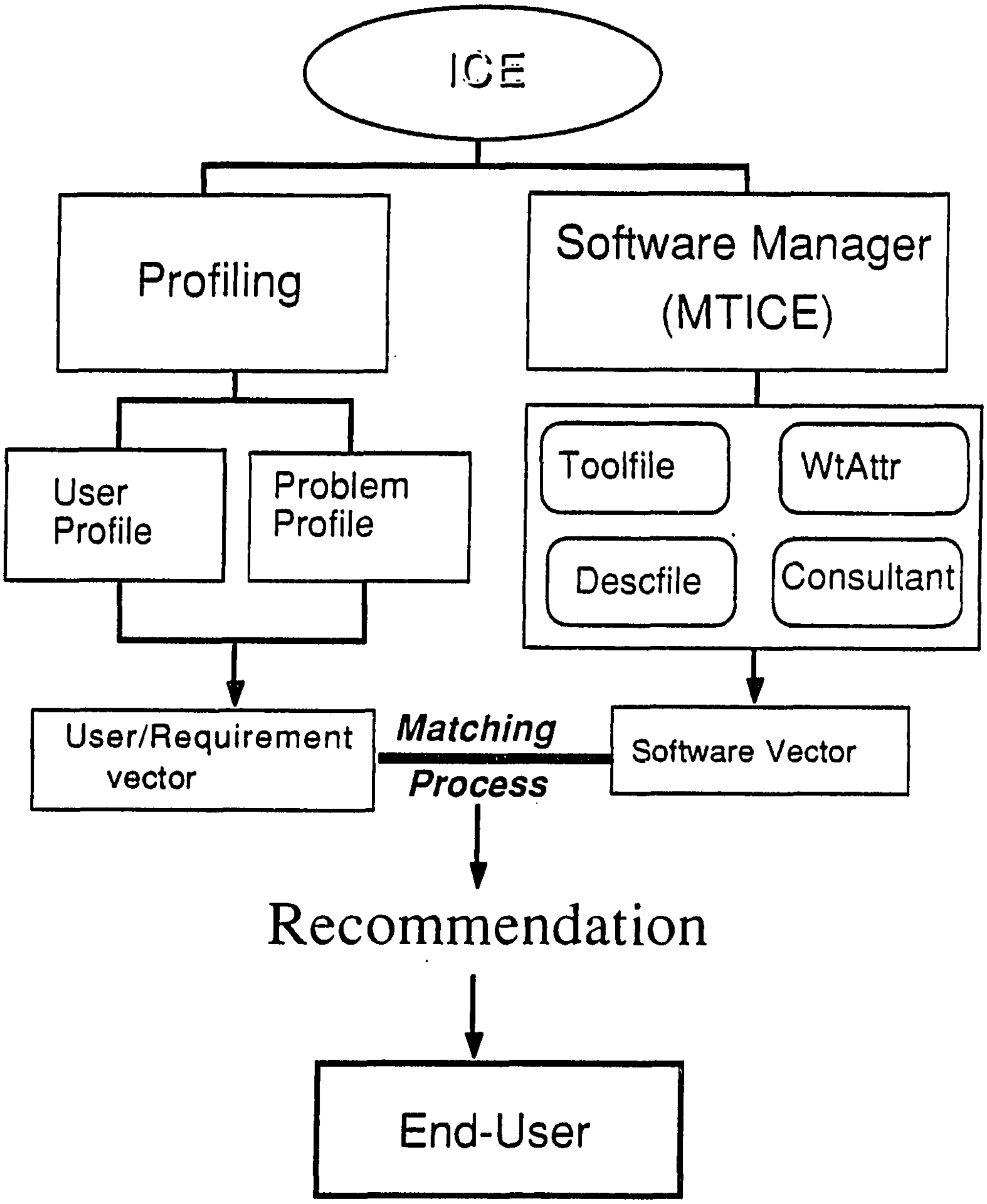

Figure 4.8: Conceptual Overview of the ICE system

nents: User Profile, Problem Profile, and Tool Profile (Figure 4.8).

1. **User Profile:** In the information center setting, users approach consultants with their own particular set of skills, computing environments, and biases. The user profile attempts to capture information defining those characteristics.

2. **Problem Profile:** Each user approaches the information center with some perceived needs which are described in terms that will help identify the resources in the information center that can be used to satisfy them.

3. **Tool Profile:** This is the information center's resource inventory, which includes the ratings (weights) that the IC places on the various attributes used to define the resources (tools).

Two of the three groups, User Profile and Tool Profile, benefit from the application of database concepts. Information about users who consult with the system is stored in the User Profile database, allowing the user to make subsequent consultations with the system without having to re-enter the user profile information.

The Tool Profile of the software tools supported by the IC is maintained as a database consisting of four files:

1. *Toolfile:* Contains each tool's identification number and an array of its attribute ratings.

2. *Descfile (Description file):* Contains the tool identification number, tool name, tool description, and the employee number of the consultant who supports the tool.

3. *Consultant file:* Contains each consultant's name, employee number, and contact phone number.

4. *WtAttr (Attribute Weighting file):* Contains the name of each tool attribute, its definition, and its weighting.

These files are separate flat files in the current implementation of ICE, but the concepts used for access and maintenance are similar to those of a relational database, with each of the files representing one relationship.

One of the considerations in the design of ICE was that it had to be maintained in the dynamic environment of the information center, with a high turnover of resources. It is impractical for the maintainers of the system continually to have to update the rules whenever a new tool is supported so that the knowledge base can reflect the current status of resources in the IC. Having the resource base separate from the rules is one method of dealing with this dynamic situation. Maintaining the resource base (the four files) then becomes an issue of database maintenance rather than knowledge base maintenance. The maintenance is carried out by MTICE.

4.4 Validation of ICE

Validation is a process undertaken for ensuring that the problem being addressed is solved correctly and that the solution is useful [Liebowitz, 1986]. ICE introduces a new process for problem solving for end users in an organization, that is, the users now have a choice of experts (human or the system) that may be consulted for obtaining a recommendation for their computing requirements. As with any problem-solving activity, the determination of the validity of the process is the key to its success [Adrion, Branstad, and Cherniavsky, 1982]. The validation of ICE will help instill confidence in its recommendations for both end users and IC consultants, allowing it to become a more useful instrument in the IC setting.

4.4.1 Methodology

The validation of ICE uses the case approach, with blind validation using a modified form of the Turing tests. This approach was used in the validation studies conducted for MYCIN [Gaschnig et al., 1983, Buchanan and Shortliffe, 1984].

4.4.1.1 Setting

The validation of ICE was conducted at the University of Arizona's Center for the Management of Information (CMI). The CMI, established in 1985, operates as an information center for the College of Business. It provides several services to the user community–students and faculty. Helping users select software to meet their needs is one such service. During the Fall 1987 semester, the CMI was staffed with seven consultants and two full-time employees. The full-time employees, referred to as CMI managers, were responsible for setting policy related to software recommendations. Each of the consultants has an area of expertise, and as a group they are knowledgeable about the various software supported by the center. Furthermore, the consultants are required to be aware of the features of the various software packages that are officially supported by the CMI.

4.4.1.2 Test Cases

The ICE system, as was discussed in an earlier chapter, provides recommendations for seven different categories of software. Case scenarios have been constructed to address each of these. In total, twenty-one cases were prepared (see Appendix H), three cases per category of software recommended. The case development was based on consultation sessions that were observed at the different information centers in an attempt to test most of the features that ICE incorporates. The requirements specified in the cases were prepared to test the various consultation paths, but in some instances they were restricted to make them reflect more accurately the length and the details of a real consultation session.

4.4.1.3 Process

Each CMI consultant was given a complete set of 21 cases and a listing of the various software officially supported by the CMI. The cases were arranged in a different order for each consultant and a random number table was used to assemble the cases. After being instructed that the recommendations were to be restricted to software officially supported by the CMI, the consultants were asked to make software recommendations for each case. Consultants could recommend more than one software package for a particular case and a consultant who considered the case not meaningful was given the option of not making a recommendation for it. It was further requested that the cases not be discussed among the consultants before the exercise was completed. A one week period was allotted to complete the cases.

The evaluation of the recommendations made by ICE used seven graduate students in the M.I.S. Department. Participation was voluntary. Each participant was assigned three cases that were randomly selected from the set of 21 cases. The participants were given a brief demonstration of the ICE system before they attempted their consultations. The tracking subsystem in ICE recorded the recommendations made by ICE for the various cases.

The recommendations made by the consultants were then compiled by bringing together all the solutions and eliminating any duplications. The solutions offered by ICE were compiled from the tracking subsystem. The recommendations from the consultants and from ICE constituted two possible solution sets to each of the cases.

The next step was preparation of the cases for evaluation by the experts. The experts for the validation process were the two managers of the CMI. The two options, recommendations made by ICE and those made by the consultants, were labeled option A and option B and appended at the end of each case. To make this a blind validation, the recommendations generated by ICE and those generated by the human consultants were arranged randomly in the A and B categories to mask the identity of the recommender. In this way, any effect of the evaluating expert's bias toward the role of computer in the IC consultation process would be minimized. In a previous study, conducted for evaluating MYCIN [Yu et al., 1979], it was observed that the bias of the expert with regard to the use of computers for the given task, had a negative effect on the results.

The experts each received copies of the complete set of 21 cases and the two possible solutions for each. The instructions to the experts asked them to judge the given solutions in the context of the case and to categorize their reaction to each solution set as: 1. Option A is better, 2. Option B is the better, 3. Both are equally good, and 4. Neither is correct. The experts were aware that the comparison was between solutions given by ICE and those provided by the CMI consultants, but were not told which option represented which.

4.4.2 Results

The results for the ICE validation study were evaluated based on the Binomial Goodness-of-Fit test. The first hypothesis being tested was:

H1: The ICE system recommendations are as good as the recommendations of the IC consultants.

Table 4.2 presents the raw scores of the expert evaluations for the recommen-

	ICE better	Consultant better	Both equal	Neither acceptable
Expert 1	8	11	1	1
Expert 2	9	9	3	0
Total	17	20	4	1

Table 4.2: Raw Scores of the Expert Evaluations

"as well as": 22 cases
"not as well as": 20 cases
Test proportion = 0.5000
Observed proportion = 0.5238
Z approximation (2-tailed p) = 0.8774

Table 4.3: Binomial Goodness-of-fit Test Results for Hypothesis 1

dations made by ICE and the CMI consultants.

The hypothesis being tested has as its emphasis whether ICE performs as well as the CMI consultants. With that emphasis, the combined score from the categories "ICE better," "Both equal," and "Neither acceptable" can be taken to indicate that ICE performed "as well as" the CMI consultant.

The results from the binomial goodness-of-fit test shown in Table 4.3 indicated that in 22 cases the experts judged the ICE system to perform "as well as" or "better" than the consultants. The number 22 was obtained by aggregating the instances in which the ICE solutions were judged better than the solutions offered by the consultant, the cases in which the solutions were judged equal and the cases in which neither ICE nor the consultant provided a satisfactory solution in the expert's opinion. Using the binomial test proportion of 0.50 the null hypothesis could not be rejected as the 'z' approximation of a 2-tailed 'p' was 0.8774. It could

"ICE better": 17 cases
"CMI consultant better": 20 cases
Test proportion = 0.5000
Observed proportion = 0.4595
Z approximation (2-tailed p) = 0.7423

Table 4.4: Binomial Goodness-of-fit Test Results for Hypothesis 2

therefore be concluded that the solutions given by ICE were comparable to those given by the consultants.

A second hypothesis was tested to check whether either of the two, ICE or CMI consultants, produces superior results. The hypothesis was:

H2: The frequency of better solutions is the same for both the ICE system and the CMI consultants.

To test this hypothesis, the cases in which both ICE and the CMI consultants were judged equally good, and scores for instances in which neither was judged satisfactory were dropped from consideration. Table 4.4 presents the results.

The results indicate that using a pretest test proportion of 0.50, the null hypothesis could not be rejected as the 'z' approximation of a 2-tailed 'p' is 0.7423. It therefore was concluded that neither of the two processes was clearly superior in terms of the recommendations it made. Stated differently, on an average, the two processes provided equally good solutions.

4.4.3 Concerns

For ICE, validation was conducted to ascertain if the recommendations made by the system were "correct" for a given scenario. The judgment as to whether or not a recommendation is "correct" was made by the designated experts.

Studies made by De Dombal [1972], and Bjerregaard et al. [1976] suggest that the use of test cases for evaluating the performance of an expert system is biased toward the system, as "cases are preselected and presented using a restricted set of descriptors that pertain only to its limited domain" [Gaschnig et al., 1983, p. 251].

Another concern with the validation process was the choice of "experts." For the purpose of this validation, an expert was defined as a policy maker for the information center. This definition was selected to reflect the appropriateness of this activity to the goals of the ICE system, i.e., to support the IC in managing its software resources and provide an effective front end to the IC consulting process. As indicated earlier, the experts for the validation process were the two managers for the CMI.

"A danger exists, however, that the results of evaluation studies may be unfairly biased by comparing the performance of a limited expert system with that of its more diversified human counterpart. Surprisingly, there is some evidence that this bias tends to work in favor of the restricted computer-based expert [Gaschnig et al., 1983].

Chapter 5

ICE Effectiveness Experiment – Study I

5.1 Experimental Testing of the ICE System

The introduction of ICE alters the consulting process for information centers by adding another possible process for matching users' needs with the available software resources. An experiment was conducted to compare the effectiveness of two processes available to users seeking recommendations for addressing software requirements: working with IC consultants versus using the ICE system.

The aim of the experiment was to study the two processes from the perspective of the end-user in an attempt to determine whether the ICE system provides adequate support for the software selection process. The perspectives of the users for each of the two processes also were evaluated to ascertain whether the type of user affected the perceived effectiveness of the process.

5.2 Experiment Setting

The ICE system has been implemented as a support system for the Center for the Management of Information (CMI) in the University of Arizona College of Business and Public Administration. The center was established in 1985 and provides several services to the user community, helping users select software to meet their needs. Each of seven consultants at the CMI has an area of expertise and, as a group, they are knowledgeable about the various software supported by the center.

A total of 96 undergraduate students from the Introduction to Business Programming, Systems Analysis, and Design and Advanced Business Programming courses offered by the Management Information Systems (MIS) Department were selected as subjects. The Introduction to Business Programming course is the second course in a series of MIS courses required for obtaining a major in MIS. The Systems Analysis and Design course and the Advanced Business Programming course are two of the last courses taken by students in MIS.

5.3 User Classification

Users in any organization are usually a diverse set; the College of Business is no exception. The need for assistance varies among users and the usefulness of the information provided is dependent on the level of expertise and perception of the data [Korfhage, 1985]. Therefore, for the purposes of this experiment, it was deemed necessary to adopt a scheme for classifying users.

For purposes of this study, the taxonomy of end-user classification developed by Rockart and Flannery [1983] was used because it is recognized as the most comprehensive available. Although useful, this taxonomy was not applicable in its entirety to the users being addressed by this experiment. Only the five underlying factors crucial for making distinctions among the classes of users were adopted for the study. These are "application focus, method of computer use, education and training requirement, and support needed" [Rockart and Flannery, 1983, pp. 777] and this classification was simplified to identify end users as either End-User

Programmers (EUP) or Nonprogramming End-Users (NPEU). EUPs were defined as users who "develop their own applications, some of which are used by other end users" and NPEUs were defined as users "whose only access to computer stored data is through software provided by others" [Rockart and Flannery, 1983, pp. 778]. The implicit assumption in using the factors from the Rockart and Flannery study is that they (the factors) are sufficient for distinguishing end-users.

5.3.1 Classification Questionnaire – Development and Testing

The classification questionnaire used to elicit data about the subject users was composed of two sections. The first section concentrated on general information about the user and included demographic data. Questions were designed to assess the user's education and training with computers, and to check the extent of the user's exposure to either the CMI facility or expert system technology. The second section allowed users to give a self-evaluation of their skills, abilities, and interests in dealing with computers by answering 18 questions by placing themselves on a seven-point scale of bi-polar pairs of responses. The questions were developed to measure the factors underlying the Rockart and Flannery [1983] classification scheme. The responses ranged from 1 (positively criterial) to 7 (negatively criterial). Some responses were reversed as a device designed to check whether the subjects understood the questions and were answering the questions consistently.

The responses to the questionnaire were subjected to a factor analysis, which allows the extraction of factor constructs underlying the questions asked. Factor constructs are themselves variables which are developed by bringing together questions or measures that virtually measure the same construct. A factor construct explains "underlying unities or common factor variances" [Kerlinger, 1986, pp. 570].

The factor analysis performed on the classification questionnaire indicated that three factors were being measured by the questions asked. These factor constructs were named application focus and support requirement (AFSR), outlook on computers (OOC), and training requirement (TR). The name given to each of the factors was based on the questions included in the measurement of that factor. The questions included in the AFSR factor indicated the focus of the end-user in application development, and a self assessment of the support required for such development. The OOC factor focused on the end-user's perception of computers in terms of the utility of computers to provide them with appropriate support. The TR factor included questions regarding the user's assessment of the help they need in learning new software or to deal with problems with the technology. The factors resulting from the pretest questionnaire were consistent with conclusions of the Rockart and Flannery [1983] study.

The classification questionnaire was factor analyzed using the varimax rotation technique. The results of the initial factor analysis are shown in Table 5.1. The results indicated that if Kaiser's varimax technique (which is the default for factor analysis in the SPSSX statistical package) was used, four factors would be extracted, since Kaiser's varimax technique uses 1.0 as the threshold eigenvalue. The choice of 1.0 is arbitrary however, and is not crucial for the extraction of factors. What is essential is that the groupings be logical and meaningful. Such was not the case when four factors were extracted, so a scree plot (Figure 5.1) was used to assist with the process of factor extraction. Examination of the scree plot indicated the possibility that there were three major factors instead of four and that those three factors accounted for 55.4 percent of the total variance. Table 5.2 presents loadings for the three factors extracted. The factor loadings used 0.40 as the cutoff. When the grouping of questions on each factor was examined for logical validity, the three-factor solution was more meaningful than the four-factor solution.

The three question groupings derived from the three-factor solution were then

VARIABLE	COMM	FACTOR	EIGENVALUE	% OF VAR	CUM PCT
Question 17	.29300	1	6.82421	37.9	37.9
Question 18	.63104	2	1.68481	9.4	47.3
Question 19	.60568	3	1.45932	8.1	55.4
Question 20	.67857	4	1.05259	5.8	61.2
Question 21	.57400				
Question 22	.68303				
Question 23	.50811				
Question 24	.61563				
Question 25	.49431				
Question 26	.78614				
Question 27	.57231				
Question 28	.62537				
Question 29	.64439				
Question 30	.73118				
Question 31	.61255				
Question 32	.66557				
Question 33	.66698				
Question 34	.63307				

Table 5.1: Classification Questionnaire: Initial Factor Analysis

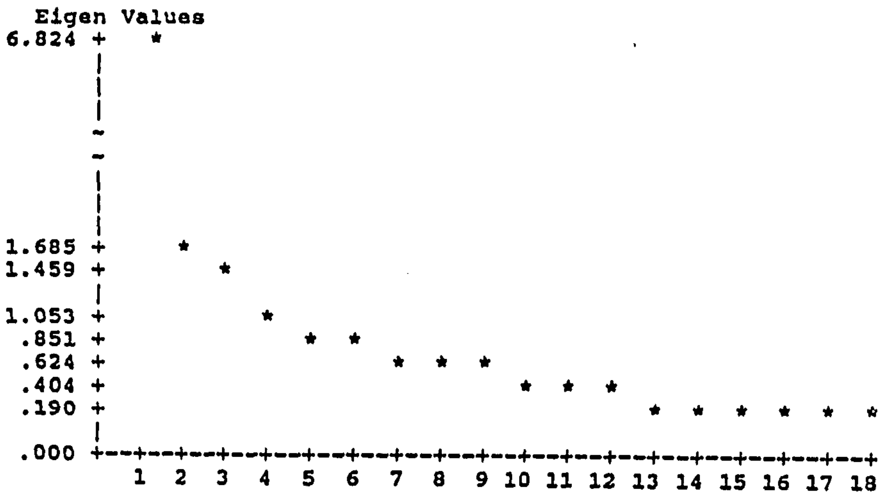

Figure 5.1: Scree Plot

ITEMS	FACTOR 1	FACTOR 2	FACTOR 3
Q33	.79084		
Q20	.74274		
Q31	.72718		
Q32	.72594		
Q27	.66613		
Q28	.61367		
Q23	.43029		.42652
Q19		.74537	
Q21		.73115	
Q24		.68244	
Q22	.49253	.62835	
Q18	.50061	.52508	
Q25		.49342	
Q17		.44907	
Q30			.82556
Q34			.78187
Q29			.66701

Table 5.2: Factor Loading of the Classification Questionnaire Items

Factor 1: Application Focus and Support Requirement (AFSR)		
Cronbach α	Number of cases	Number of items
0.8787	96	8
Factor 2: Outlook on Computers (OOC)		
0.8206	96	5
Factor 3: Training Requirement (TR)		
0.8023	96	5

Table 5.3: Reliability measures for Factors Extracted from the Classification Questionnaire

tested for internal consistency (reliability) by using the Cronbach α coefficient (Table 5.3). The groupings allowed loading of an item on more than one factor. For example, questions 18 and 22 were part of both factor 1 and factor 2 (see Table 5.2).

For supporting the reliability of a subset of items from a factor, Nunnally [1967] suggests that the Cronbach α level be maintained at the .80 level. Tracey [1985] and Srinivasan [1985] suggest lower levels, .70 and .50 respectively. Since this experiment is exploratory in nature and no previously validated instruments were available, the Cronbach α level was set a priori at a relatively high level of .75. Because each of the three factors extracted, AFSR, OOC and TR had a reliability coefficient greater than .80, it was concluded that the factors were internally consistent operationalizations of the theoretical variables.

5.3.2 Results of the Classification

The classification questionnaire was administered to demonstrate that two groups of students involved in the study are distinguishable. A t-test was used for making this distinction. The t-test was used for each of the three factors that had been identified (Table 5.4). For purposes of the t-test, the scales of the questionnaire were adjusted such that lower scores represent a more positive outlook for the factor.

The classification of the students was as follows: group 1 was the set of students from the Introduction to Business Programming course, and group 2 was composed of the Advanced Business Programming and Systems Analysis and Design students. The first hypothesis being tested is concerned with factor 1:

H1: The application focus and the support requirement is independent of the classification of end-user.

For factor 1, AFSR, the mean score for group 1 was 4.0369 ($\sigma = 1.105$) as compared with 2.8536 ($\sigma = 0.939$) for group 2. With the t-value of 5.56, using a two-tailed test and a separate estimate of variance, the null hypothesis was rejected with $p < 0.01$ (df = 94). This indicated that the application focus and the support requirement depend on the classification of the user.

The second hypothesis was concerned with factor 2. This factor assesses the user's outlook on computers. The hypothesis was that:

H2: The outlook on computers is independent of the classification of end-user.

For factor 2, OOC, the t-value is 3.08, with the mean score for group 1 was 2.3967 ($\sigma = 0.924$) as compared with 1.88 ($\sigma = 0.703$) for group 2. Again using the two-tailed test and a separate variance estimate, we rejected the null hypothesis because $p < 0.01$ (df = 94) indicated that the outlook on computers was dependent on the class of the user.

The third hypothesis dealt with factor 3, training requirement of the users. Here we tried to distinguish between the two groups based on this factor. It was

User Classification – Factor AFSR							
VARIABLE	n	x̄	σ	STANDARD ERROR	T VALUE	df	2-TAIL PROB
NPEU	61	4.0369	1.105	0.142	5.56	94	0.000
EUP	35	2.8536	0.939	0.159			
User Classification – Factor OOC							
NPEU	61	2.3967	0.924	0.118	3.08	94	0.003
EUP	35	1.8800	0.703	0.119			
User Classification – Factor TR							
NPEU	61	3.7148	1.115	0.143	3.31	94	0.001
EUP	35	2.9543	1.064	0.180			

Table 5.4: User Classification t-test Scores

hypothesized that:

H3: The training requirement is independent of the classification of end-user.

The resulting t-value is 3.31, with mean score for factor 3 for group 1 was 3.7148 (σ = 1.115) and for group 2 was 2.9543 (σ = 1.064). Once again a two-tailed test and a separate estimate of variance were used. The null hypothesis was rejected, with $p < 0.01$ (df = 94). Table 5.4 presents the complete set of results from the t-test.

The results from the tests presented allowed the assertion that the two groups of students included in the study were distinguishable, based on the factors AFSR, OOC, and TR. The scores also indicated that the subjects in the EUP group showed a more positive outlook for each of the three factors.

5.4 Comparative Consultation Effectiveness

The experiment conducted provided data to evaluate the comparative merits of the two channels as a process for obtaining assistance with software selection. As discussed in the previous section, the end-user selection process was designed ensured that two classes of users were distinguishable. The results of classification process showed that the users could be distinguished based on the three identified factors.

This experiment focused on evaluating effectiveness of ICE and the CMI-Consultants from the perspective of the end-users. There are primarily two schools of thought regarding effectiveness. The first deals with effectiveness in terms of perceived user satisfaction [Ginzberg, 1978; Bailey and Pearson, 1983; Ives, Olson and Baroudi, 1983]. The second view on effectiveness is in terms of system usage [Ein-Dor and Segev, 1978]. The common link between the approaches is the importance of user satisfaction in any measure of system effectiveness. Both of these approaches are important, and it was a focus of the research to determine which is more appropriate. In the present experiment the former definition of effectiveness was used to evaluate the ICE system by measuring the perceived utility of the system to its users by evaluating "consultation effectiveness," a hypothetical construct that is a compound variable consisting of measurements for satisfaction, task basis for effectiveness and recommendation basis for effectiveness. The development of a measure for consultation effectiveness is discussed in the following section.

5.4.1 Effectiveness Questionnaires – Development and Testing

Two questionnaires were developed for measuring effectiveness. The first was used for measuring the effectiveness of ICE, and the second was used for evaluating the effectiveness of the CMI consultants. The questionnaires shared a common set of 20 questions, divided into task-specific and process-specific questions. The remainder of the questions (four in the consultant effectiveness questionnaire, and eight in the ICE effectiveness questionnaire) focused on evaluating the specific features of the process being addressed–ICE or CMI-Consultant.

There is a scarcity of literature addressing the issue of effectiveness from the perspective of end-users. As mentioned in the literature review (chapter 2), the majority of the publications on end-user computing concentrate on the conceptual or theoretical aspects. The development of instruments for measuring consultation effectiveness was based on several alternative measures used for evaluating the effectiveness of traditional information systems. The instruments included concepts of user information satisfaction [Ives, Olson and Baroudi, 1983], user perception of information [Gallagher, 1974], perceived usefulness [Larcker and Lessig, 1980] and user satisfaction [Bailey and Pearson, 1983].

The twenty common questions shared by the two instruments for measuring effectiveness of the CMI-Consultants and the ICE system produced a composite score for consultation effectiveness. This set of questions was tested for internal consistency based on the responses of all the participants, and resulted in a Cronbach α score of .7522 which is within the .75 level set for the present study.

The twenty questions were also factor analyzed. This analysis indicated that the effectiveness score was composed of three factors. The basis for extracting three factors was the scree plot (Figure 5.2). The details of the factor analysis are presented in Table 5.5 and 5.5b.

The level of internal consistency for these three factors, measured by Cronbach's α, vary substantially. The first factor, labeled "user satisfaction," has the highest degree of internal consistency of the three factors indicated by the Cronbach α of 0.8701. The second factor is labeled "task basis for effectiveness" has

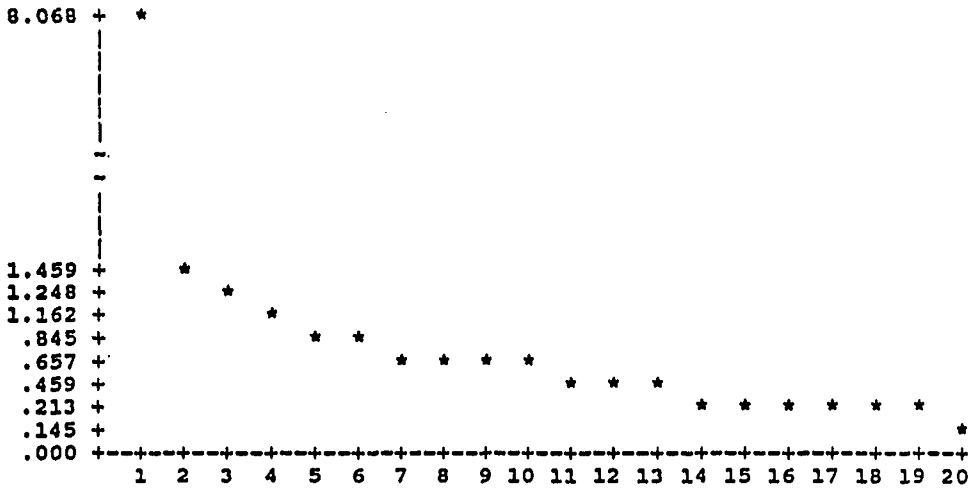

Figure 5.2: Scree Plot for the Effectiveness Questionnaire

VARIABLE	COMM	FACTOR	EIGENVALUE	% OF VAR	CUM PCT
Question 1	.55990	1	6.34038	31.7	31.7
Question 2	.46074	2	1.82662	9.1	40.8
Question 3	.40097	3	1.57461	7.9	48.7
Question 4	.56505				
Question 5	.56508				
Question 6	.63082				
Question 7	.29227				
Question 8	.24686				
Question 9	.55900				
Question 10	.70187				
Question 14	.59775				
Question 15	.28623				
Question 18	.36538				
Question 17	.71061				
Question 13	.53297				
Question 21	.48578				
Question 19	.65848				
Question 22	.30584				
Question 23	.41712				
Question 24	.39887				

Table 5.5: Effectiveness Questionnaire: Initial Factor Analysis

ITEMS	FACTOR 1	FACTOR 2	FACTOR 3
Q17	.84103		
Q19	.74769		
Q10	.71635		
Q1	.70824		
Q13	.65588		
Q4	.64655		
Q23	.64495		
Q14	.61905		.46251
Q18	.60102		
Q6	.56040	.40381	
Q22	.50251		
Q7		.43505	
Q9		-.71343	
Q21		.66942	
Q5		.61104	
Q2		.54335	.40418
Q24			-.54385
Q15			.48699
Q8			.47862

Table 5.6: Effectiveness Questionnaire: Factor Loadings of Questionnaire Items

a reliability score of 0.7707. The third factor labeled "recommendation basis for effectiveness" has a Cronbach α of 0.3329 which is lower than the .75 threshold, but is included as it is consistent with the system effectiveness measurement literature. The results that are presented, include both the composite score for consultation effectiveness, and the scores for the underlying individual factors. This allows for an overall evaluation as well as a factorwise evaluation of the consultation effectiveness.

5.5 Subjects

The classification questionnaire [Appendix E] was administered to all 96 participants prior to providing them with any information regarding the study. The questionnaire was administered to 24 students from the Systems Analysis and Design course, 11 from the Advanced Business Programming course, and 61 from the Introduction to Business Programming course. The assignments of the classes to groups experiencing the two different processes was done on the basis of a coin toss.

The distribution of students included in this study, based on their current class level, is presented in Table 5.7. Over 80% of the participants had some exposure to computers for two years or more. The majority of the subjects (55.8%) had learned one to three computer languages or software products.

5.6 Task

The task chosen for the experiment was selection of software for report writing. The selection of the task was based on a study by Amoroso and Cheney [1987] indicating that report generation accounts for more than 90% of the end-user computing software requirements.

CLASS	FREQUENCY	%	Cumulative %
Sophomore	24	24.7	24.7
Junior	30	30.9	55.7
Senior	36	37.1	92.8
Graduate	4	4.1	96.9
Unknown	2	3.1	100.0
TOTAL	96	100.0	

Table 5.7: Distribution of Participants

The task was described to the students, each of whom was then handed an assignment sheet describing the experiment and the task. The assignment sheet given to each group (Appendix B and Appendix A) included a brief description of the study and provided the students some guidelines for describing their report writing needs. The students were asked to sign up for a convenient time slot during the following week to schedule time for consultation with either a CMI-consultant or the ICE system. To ensure that the students participated in the experiment seriously, they were required to report back to the instructor the software package(s) recommended for their use and, further, involvement in the study was to represent 5% of the total course grade.

5.7 Procedures

As has been noted, the classification questionnaire was administered to the students prior to their being given any information regarding the experiment. Upon completing the questionnaire, the participants were introduced to the background and the purpose of the study. Those who were in the group that would use the ICE system received an explanation of the reason for developing the system as well as of the system's capability for providing assistance with the selection of software (Appendix D). The group of students who were assigned to the CMI-Consultants were given a similar introduction which focused on the purpose of CMI and it was explained how the consultants could assist them with software selection (Appendix C). It was made clear to both groups that the study was to evaluate the two methods of providing assistance with software selection. The introductions for both the groups were kept similar to each other.

5.7.1 Consultation Session with the CMI-Consultants

Each student was allotted a 20-minute time slot to discuss his or her requirements with a consultant. The time requirement was not rigidly observed. The consultants were asked to treat each student as a new case. The students were asked to translate into their own words the requirements given to them in the assignment sheet, i.e., they were required to verbalize their requirements instead of handing the assignment sheet to the consultant. To ensure that this process was followed, all the consultation sessions with the consultant were recorded on tape. No other restrictions were placed on the consulting process.

Consultation sessions were concluded with the consultant's making a software recommendation. The user was then given the effectiveness questionnaire by the consultant. Users were required to complete the questionnaire immediately following the consultation session and to return it to a sealed box.

5.7.2 Consultation Session with the ICE system

A pre-assigned terminal that was continually logged into the ICE system was used, eliminating the need for the user to learn the log-in procedure. Users were allowed to acquaint themselves with the ICE system by conducting a test consultation. Assistance was provided to the users during the test consultations, primarily for the purpose of acquainting the users with the keyboard configuration. No technical assistance was provided during the actual consultation session to get a recommendation for the task designated for this experiment. The users were encouraged to use the help features of the ICE system if any problems arose. In all but four cases, the users needed no help from the technical consultant for ICE.

Upon completion of the consultation session with ICE, the effectiveness questionnaire was administered to the users. As in the case of the students assigned to the CMI-Consultants, the participants were asked to complete the effectiveness questionnaire immediately after the session. The completed questionnaires were returned to a sealed box.

5.8 Results of the Study

Two primary research questions were being investigated in this experiment:

1. Does a relationship exist between the independent variables, the channel for getting assistance with software selection–ICE or CMI-Consultant–and the class of end user–EUP or NPEU–as measured by the dependent variable consultation effectiveness?
2. If the relationship exists, what is its nature?

The study consisted of a 2X2 factorial design. The factorial design was chosen because it allows an analysis of the main effect, i.e., individual effect of the independent variables on the dependent variable, and the interaction effect which is a combined effect of the independent variables on the dependent variable.

The study focused on:

1. The consultation effectiveness of ICE in comparison to the CMI-Consultants for providing assistance with software selection.

2. The effectiveness of the process based on the type of user–EUP or NPEU.

A multivariate analysis of variance (MANOVA) was conducted for the three factors constituting "consultation effectiveness", namely user satisfaction, task basis for effectiveness and the recommendation basis for effectiveness. Hotellings multivariate test of significance Table 5.8 indicated that the type of process, i.e., ICE or CMI consultant, had a significant impact indicated by an F statistic of 10.665 (d.f. = 1, 77). The univariate F-tests Table 5.9 for each of the three factors constituting effectiveness, however, indicated that the factors user satisfaction and the task basis for effectiveness were significant at the 0.01 level. The recommendation basis for effectiveness was however, not significant with the $p = 0.435$. The F-statistic values for the three factors were 11.179, 14.234 and 0.615 (d.f. = 1, 77) respectively.

The Hotellings score relating to the type of end-user, EUP or NPEU, indicated the effect of user type on consultation effectiveness to be significant $p = 0.011$ (Table 5.8). The univariate F-tests for the three factors indicated, however, that only the factor 'user satisfaction' had a significant impact ($p = 0.014$) (Table 5.9).

The interaction effect for the type of user, and the type of process was significant, indicated by the Hotellings test score of 0.233. These MANOVA results indicate a need for further analysis, which is presented in the following sections by conducting factorial analysis of variance on each of the three factors constituting "consultation effectiveness".

	Value	Exact F	Hypoth. DF	Error DF	Significance of F
Type of Process (ICE/CMI)	0.426	10.665	3	75	0.000
Type of User (NPEU/EUP)	0.158	3.963	3	75	0.011
Interaction	0.233	5.835	3	75	0.001

Table 5.8: Hotellings Multivariate Test of Significance - Study I

Variable	Hypoth. SS	Error SS	Hypoth MS	Error MS	F	Sig. of F
Type of Process						
User satisfaction	5.711	39.335	5.711	.510	11.179	.001
Task basis	2.506	13.549	2.506	.175	14.243	.000
Recommendation basis	.237	29.673	.237	.385	.615	.435
Type of User						
User satisfaction	3.259	39.335	3.259	.510	6.380	.014
Task basis	.279	13.549	.279	.175	1.588	.211
Recommendation basis	.110	29.673	.110	.385	.286	.594
Interaction						
User satisfaction	3.695	39.335	3.695	.510	7.234	.009
Task basis	1.130	13.549	1.130	.175	6.424	.013
Recommendation basis	.227	29.673	.227	.385	.591	.444

Table 5.9: Univariate F-tests (d.f. = 1, 77) – Study I

Process	ICE = 1.85 (n = 39)	CMI Consultant = 2.38 (n = 42)
User Class	NPEU = 1.90 (n = 53)	EUP = 2.54 (n = 28)

Table 5.10: User Satisfaction – Cell Means

	NPEU	EUP
ICE System	1.85	1.82
CMI Consultant	1.97	2.88

Table 5.11: User Satisfaction – 2X2 Cell Means

5.8.1 User Satisfaction

The main effects for the factorial ANOVA using user satisfaction as the dependent variable were similar to those stated earlier as being the main effects for the study, i.e., first, if the process affects the level of user satisfaction and, second, if the user class affects the user satisfaction.

The hypothesis for testing the first main effect was:

H4: User satisfaction is independent of the type of consultation method (ICE or Consultant) used.

The statistics for evaluating the hypothesis are presented in Table 5.12. The cell means for the factorial analysis of variance are shown in Table 5.10 and Table 5.11.

The results showed the F-statistic as 6.469 (df = 1, 78), allowing the null hypothesis to be rejected with $p = 0.013$. It therefore was concluded that the type of consultation method did have an impact on the level of user satisfaction. Furthermore, the cell means indicated that the users consulting with ICE for assistance with software selection were more satisfied when compared with users consulting with the CMI-Consultants. The cell means indicated the satisfaction score for users

Source of Variation	Sum of Squares	DF	Mean Square	F	Signif of F
MAIN EFFECTS					
I. PROCESS ICE vs CMI-Consultant	3.305	1	3.305	6.469	0.013
II. USER CLASS NPEU vs EUP	5.006	1	5.006	9.789	0.002
INTERACTION EFFECT					
Process - User class	3.696	1	3.696	7.235	0.009

Table 5.12: Results of the 2-way Factorial Analysis of Variance for User Satisfaction

of ICE as 1.85 and that of the users of the CMI-Consultants as 2.38. It is important to note that it was not possible to place any values on the scores although, because the scales are ordinal, rank ordering of the responses was possible, i.e., it is possible to say that 1.85 represented greater satisfaction than 2.38.

The second hypothesis being tested was:

H5: User satisfaction is independent of the end-user classification.

Table 5.12 shows the F-statistic at 9.798 (df = 1, 78) allowing the null hypothesis to be rejected with $p < 0.01$, indicating that different classes of end-users experienced different levels of satisfaction. The cell means for users classified as NPEU was 1.90, compared with 2.54 for EUP users. On average, the NPEUs therefore experienced greater satisfaction with the consulting process than did the EUPs.

The results further indicated that the interaction effect between the two independent variables was significant, i.e., the type of user and the type of consulting process both needed to be considered to maximize user satisfaction with the con-

Process	ICE = 3.88 (n = 39)	CMI Consultant = 4.32 (n = 42)
User Class	NPEU = 4.13 (n = 53)	EUP = 4.06 (n = 28)

Table 5.13: Task Basis for Effectiveness – Cell Means

sultation process for software selection. The cell means table (Table 5.11) indicated that both the NPEUs and the EUPs showed approximately equal satisfaction in using the ICE system (1.85 and 1.82 respectively). In using the CMI-Consultants, however, the EUP user was far more dissatisfied with the process than was the NPEU user (1.97 and 2.88 respectively).

5.8.2 Task Basis for Effectiveness

In evaluating the factor "task basis for effectiveness," the hypothesis being tested was:

H6: The task basis for effectiveness is independent of the type of consultation method (ICE or Consultant) used.

The results for testing the task basis for the effectiveness are presented in Table 5.13, Table 5.14 and Table 5.15. The F-statistic for testing the above stated hypothesis was 25.301 (df = 1, 78), allowing the null hypothesis to be rejected with $p < 0.01$. By rejecting the null hypothesis we concluded that the task basis for effectiveness is dependent on the consultation method used. Furthermore, using the cell means, it was accepted that ICE users found ICE to be more appropriate for the task given, as compared with the users who were assigned to the CMI-Consultants; scores were 3.88 and 4.32, respectively.

The second hypothesis being tested was:

H9: Task basis for effectiveness is independent of the class of the

	NPEU	EUP
ICE System	3.85	3.98
CMI Consultant	4.13	4.06

Table 5.14: Task Basis for Effectiveness – 2X2 Cell Means

Source of Variation	Sum of Squares	DF	Mean Square	F	Signif of F
MAIN EFFECTS					
I. PROCESS ICE vs CMI-Consultant	4.452	1	4.452	25.301	0.000
II. USER CLASS NPEU vs EUP	0.574	1	0.574	3.263	0.075
INTERACTION EFFECT					
Process - User class	1.131	1	1.131	6.425	0.113

Table 5.15: Results of the 2-way Factorial Analysis of Variance for Task Basis for Effectiveness

Process	ICE = 2.33 (n = 39)	CMI Consultant = 2.47 (n = 42)
User Class	NPEU = 2.42 (n = 53)	EUP = 2.36 (n = 28)

Table 5.16: Recommendation Basis for Effectiveness – Cell Means

	NPEU	EUP
ICE System	2.32	2.36
CMI Consultant	2.56	2.36

Table 5.17: Recommendation Basis for Effectiveness – 2X2 Cell Means

end-user.

The results showed the F-statistic at 3.263 (df = 1, 78). This did not allow us to reject the null hypothesis while maintaining the α level at 0.05. We therefore were unable to conclude that there was a dependent relationship between the class of the end user and user perception of effectiveness, based on the task for the study.

The results however indicated that the interaction effect between the two independent variable was statistically significant with p = 0.013. Table 5.14 for cell means indicates that NPEUs using ICE perceived the effectiveness of the consultation method for the task specified as the more appropriate, with a score of 3.85. The NPEUs using the consultants were the group most dissatisfied with the consultation method used for the task, with a score of 4.13.

5.8.3 Recommendation Basis for Effectiveness

The results for the recommendation basis for perceiving effectiveness presented in Table 5.16, Table 5.17, and Table 5.18, indicate that no statistical significance could be attributed to either of the main effects or the interaction effect.

Source of Variation	Sum of Squares	DF	Mean Square	F	Signif of F
MAIN EFFECTS					
I. PROCESS ICE vs CMI-Consultant	0.500	1	0.500	1.298	0.258
II. USER CLASS NPEU vs EUP	0.191	1	0.191	0.496	0.483
INTERACTION EFFECT					
Process - User class	0.228	1	0.228	0.698	0.556

Table 5.18: Results of the 2-way Factorial Analysis of Variance for Recommendation Basis for Effectiveness

5.8.4 Consultation Effectiveness

For studying the comparative effectiveness of the two processes the dependent variable, consultation effectiveness, was represented by a composite score. An acceptable level of internal consistency, Cronbach $\alpha = 0.7522$, allowed the use of a composite score to represent consultation effectiveness (Table 5.19).

It was noted however, that the relatively higher reliability score the factor "user satisfaction", and the comparatively lower reliability scores for the other two factors, i.e., task basis for effectiveness and the consultation basis for effectiveness indicate the dominant effect of user satisfaction in the measure of consultation effectiveness.

The scores for the composite score for consultation effectiveness were tabulated in descending order, i.e., lower scores represent higher levels of effectiveness and vice versa. The cell means for the factorial analysis of variance are presented in Table 5.20 and Table 5.21.

Consultation Effectiveness – the composite score		
Cronbach α	Number of cases	Number of items
0.7522	81	20
Factor: User Satisfaction		
0.8701	81	12
Factor: Task Basis for Effectiveness		
0.7707	81	5
Factor: Recommendation Basis for Effectiveness		
0.3329	81	5

Table 5.19: Reliability Scores for Factors Extracted from the Effectiveness Questionnaires

Process	ICE = 2.44 (n = 39)	CMI Consultant = 2.89 (n = 42)
User Class	NPEU = 2.56 (n = 53)	EUP = 2.88 (n = 28)

Table 5.20: Consultation Effectiveness – Cell Means

	NPEU	EUP
ICE System	2.44	2.43
CMI Consultant	2.72	3.09

Table 5.21: Consultation Effectiveness - 2X2 Cell Means

Source of Variation	Sum of Squares	DF	Mean Square	F	Signif of F
MAIN EFFECTS					
I. PROCESS ICE vs CMI-Consultant	3.116	1	3.116	12.874	0.001
II. USER CLASS NPEU vs EUP	0.827	1	0.827	3.416	0.068
INTERACTION EFFECT					
Process - User class	0.622	1	0.622	2.571	0.113

Table 5.22: Results of the 2-way Factorial Analysis of Variance for Consultation Effectiveness

The detailed statistics for the factorial analysis of variance for consultation effectiveness are presented in Table 5.22.

The study of consultation effectiveness consists of two main effects. The hypothesis for evaluating the first main effect was:

H7: Consultation effectiveness is independent of the type of consultation method (ICE or Consultant) used.

The results showed the F-statistic as 12.874 (df = 1, 78), thus allowing the null hypothesis to be rejected ($\alpha = 0.05$) with $p < 0.01$. It therefore was concluded that consultation effectiveness was dependent on the method of consultation. Further, the cell means (Table 5.20 and Table 5.21) suggested that the group of end-users using the ICE system expressed a higher degree of consultation effectiveness than did the group using the CMI-Consultants. The group using the ICE system scored an average of 2.44 on the effectiveness scale as compared with 2.88 for the group using the CMI-Consultants.

The second main effect is evaluated by the following hypothesis:

H8: Consultation effectiveness is independent of the end- user classification.

The F-statistic for this effect was 3.416 (df = 1, 78). Maintaining the α level at 0.05, we could not reject the null hypothesis, since p = 0.068. It therefore was not possible to establish dependence of the consultation effectiveness on the class of the end user.

Interaction effect is defined as "the operation or influence of one independent variable on a dependent variable... (based) ...on the level of another independent variable" [Kerlinger, 1979, pp 230]. The interaction effect was used to evaluate consultation effectiveness by studying the joint effect of both the type of process–ICE or CMI-Consultant–and the class of end user–EUP or NPEU.

The factorial analysis of variance (Table 5.22) indicated that the interaction effect was not significant. With the F-statistic = 2.571 (df = 1, 78) the null hypothesis could not be rejected for the α level of 0.05, because p = 0.113. We therefore were unable to identify whether or not the choice of process was based on the class of the end-user.

Chapter 6

ICE Effectiveness Experiment - Study II

6.1 Overview

The focus of this chapter is on extending the results of study I discussed in chapter 5 for the purpose of verification of the effectiveness of the ICE system. This chapter uses a new data set, obtained independently of the data used in study I. Besides helping in verifying the effectiveness of the ICE system, the use of these new data will also help validate the measurement instruments and improve the reliability of the findings.

Such a follow up study is necessary for two reasons: 1. verification is an ongoing process, and repeating it using different subjects will strengthen the reliability of the results, and 2. a lack of validated instruments for measuring consultation effectiveness of knowledge-based systems requires multiple use of the instruments in an effort to validate their consistency and reliability.

The aim of this follow-on experiment was identical to that of the earlier study, namely, to determine the comparative consultation effectiveness of the two processes available to users for obtaining recommendations for their software requirements. The experimental design and the method of data gathering also were identical to those used in study I. The experiment setting was kept as similar as possible to allow for valid comparison of the results. The study took the perspective of the end-user in examining effectiveness.

6.2 User Classification

The subjects in this experiment were undergraduate students from the College of Business and Public Administration at the University of Arizona. The participants, a total of 100 students, were from two sections of the Business Programming course and two sections of the Data Management Systems course. The Business Programming course, the second in a series of MIS courses, is primarily constituted of sophomore and junior level students. The Data Management Systems course, on the other hand, is an elective course for students majoring in MIS, and is taken by senior level students.

Despite the subjects being drawn from the same population as study I, i.e., undergraduate students from the business school, the two sets of data (study I and study II) could not be combined, as the subjects in the two studies differed significantly on one of the three factors AFSR – t-value = 2.01 (d.f. = 193) $p < 0.05$ – used to classify the subjects (Table 6.1). The difference in the two samples based on the AFSR factor is attributed primarily to study I being conducted in the early part of the semester, and study II being conducted towards the end of the semester. This difference reflects a change in the application focus and support requirement (AFSR) of the users caused by the learning process during the semester.

User Classification – Factor AFSR							
VARIABLE	n	x̄	σ	STANDARD ERROR	T VALUE	df	2-TAIL PROB
STUDY I	96	3.6055	1.190	0.121	2.01	193	0.046
STUDY II	99	3.2866	1.014	0.102			
User Classification – Factor OOC							
STUDY I	96	2.2083	0.883	0.090	0.04	194	0.972
STUDY II	100	2.2040	0.816	0.082			
User Classification – Factor TR							
STUDY I	96	3.4375	1.151	0.118	-0.23	194	0.821
STUDY II	100	3.4720	0.962	0.096			

Table 6.1: User Comparison (Study I versus Study II) – t-test Scores

6.2.1 Classification Questionnaire

The students, the "end-users" for the purposes of this experiment, were classified using the taxonomy developed by Rockart and Flannery [1983]. This was the same classification scheme that was used in study I. The users were categorized as Non-Programming End Users (NPEU) and End User Programmers (EUP) in the manner discussed in detail in chapter 5.

Because no previously validated instruments were in existence, user classification questionnaires were developed for the explicit purpose of placing users in the categories identified by Rockart and Flannery [1983]. The development criteria for the classification questionnaires were discussed in detail in the previous chapter. The present discussion focuses on the validation of the questionnaires by demonstrating the reliability of this questionnaire for measuring the constructs identified in chapter 5.

VARIABLE	COMM	FACTOR	EIGENVALUE	% OF VAR	CUM PCT
Question 17	.49325	1	5.78028	32.1	32.1
Question 18	.43836	2	1.91902	10.7	42.8
Question 19	.60094	3	1.73090	9.6	52.4
Question 20	.50843				
Question 21	.49594				
Question 22	.65653				
Question 23	.59505				
Question 24	.49817				
Question 25	.39435				
Question 26	.52509				
Question 27	.49074				
Question 28	.59259				
Question 29	.20221				
Question 30	.56589				
Question 31	.70085				
Question 32	.43770				
Question 33	.64407				
Question 34	.59005				

Table 6.2: Classification Questionnaire: Initial Factor Analysis

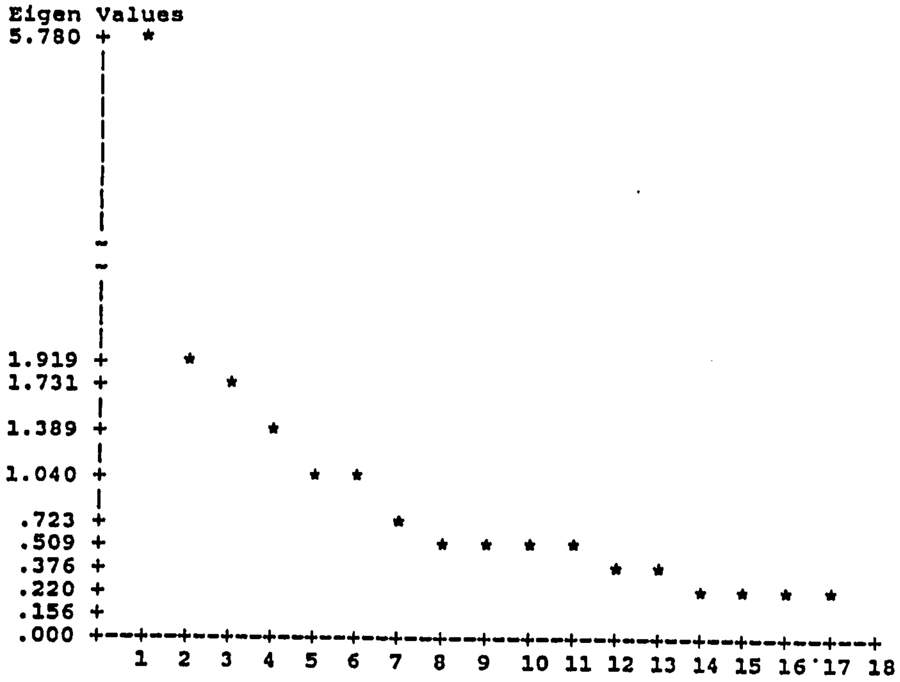

Figure 6.1: Scree Plot

The classification questionnaire was factor analyzed using the new set of data gathered. The results (Table 6.2) indicated the possibility of extracting up to seven factors using the Kaiser's varimax technique (Eigen value = 1.0). As in the earlier study, a scree plot (Figure 6.1) was used to assist in determining the number of factors that should be extracted. On the basis of the scree plot, and on observing the groupings of questions constituting a factor, it was decided that three factors would be extracted.

The number of factors extracted was the same as that in study I, although the questions included in the factors were not identical for the present study and study I. Nevertheless, there was a definite similarity in the focus of the questions across the two studies when a logical grouping of the questions was made, using the

ITEMS	AFSR	OOC	TR
Q22	.76384		
Q19	.73390		
Q23	.70224		
Q17	.69070		
Q18	.65653		
Q21	.61850		
Q24	.59673		
Q31		.80089	
Q33		.76039	
Q27		.68526	
Q32		.63091	
Q20	.43369	.56273	
Q28	.40698	.49922	.42159
Q34			.72612
Q30			.63207
Q26			-.62802
Q25			-.51982

Table 6.3: Factor Loading of the Classification Questionnaire Items

factor loadings as a guide. This indicated that the follow-up study instrument was measuring constructs similar to those measured in study I. The question groupings are shown in Table 6.3.

The three groupings of questions were tested for internal consistency by using the Cronbach α coefficient. As mentioned earlier, the groupings were made with emphasis on the logical validity of the questions being clustered. This concern led to items being loaded on more than one factor to ensure the cohesiveness of the grouping. Since the factors demonstrated a focus similar to that observed

Factor 1: Application Focus and Support Requirement (AFSR)		
Cronbach α	Number of cases	Number of items
0.8193	99	8
Factor 2: Outlook on Computers (OOC)		
0.7844	100	5
Factor 3: Training Requirement (TR)		
0.7175	100	5

Table 6.4: Reliability Measures for Factors Extracted from the Classification Questionnaire

in study I, the factors were given the same titles as those in study I: application focus and support requirement (AFSR), outlook on computers (OOC), and training requirement (TR). The reliability scores measured by the Cronbach α coefficient for the three groups shown in Table 6.4 indicated that, of the factors extracted, AFSR and OOC had reliability coefficients of 0.82 and 0.78, respectively. These were greater than 0.75, the reliability level selected in chapter 5. The Cronbach α score for the factor TR is 0.71 which, though lower than the 0.75 level, still was within the level generally accepted for explorative research [Srinivasan, 1985 and Tracey, 1985].

6.2.2 Distinguishing the Classes of Users

The aim of using the classification questionnaire was to show distinguishability among the subjects in order to separate them into the two classes of interest. A t-test was performed on each of the factors extracted. The subjects were grouped on the basis of the course in which they were enrolled. As was noted in study I data analysis, the scales of the questionnaire were adjusted such that lower scores

User Classification – Factor AFSR							
VARIABLE	n	x̄	σ	STANDARD ERROR	T VALUE	df	2-TAIL PROB
NPEU	57	3.6579	0.934	0.124	4.70	90.08	0.000
EUP	42	2.7827	0.904	0.139			
User Classification – Factor OOC							
NPEU	58	2.4586	0.898	0.118	4.26	93.80	0.000
EUP	42	1.8524	0.517	0.080			
User Classification – Factor TR							
NPEU	58	3.7000	0.994	0.131	2.97	96.05	0.004
EUP	42	3.1571	0.827	0.128			

Table 6.5: User Classification t-test Scores

represented a more positive outlook on the factor and vice versa.

The hypothesis being tested for factor 1 was:

H1: The application focus and the support requirement is independent of the classification of the end-user.

For factor 1 (Table 6.5), AFSR, the mean score for NPEUs was 3.66 (σ = 0.93), compared with 2.78 (σ = 0.90) for EUPs resulting in a t-value of 4.70. Using a two-tailed test with, the null hypothesis was rejected, with $p < 0.01$. The conclusion from this result was that the application focus and support requirement was dependent on the classification of the end user, and that the two groups were distinguishable on the basis of this factor.

The second hypothesis focused on factor 2, the user's outlook on computers. It was hypothesized that:

H2: The outlook on computers is independent of the classification of end-user.

For the second factor OOC, (Table 6.5), NPEU subjects had a mean score of 2.46 ($\sigma = 0.9$). In comparison, EUP subjects scored 1.85 ($\sigma = 0.52$), resulting in a t-value of 4.26. Again, using a two-tailed test and separate variance estimate, the null hypothesis could be rejected, with $p < 0.01$, indicating that the outlook on computers was dependent on the class of user and that the groups were distinguishable on the basis of this factor.

The third hypothesis considered attempts to classify users on the basis of their training requirements; the hypothesis was:

H3: The training requirement is independent of the classification of end-user.

We again compared the mean scores of the two groups. Group 1 had a mean score of 3.70 ($\sigma = 0.99$), as compared with 3.16 ($\sigma = 0.82$) for group 2. The resulting t-value was 2.97. Using the two-tailed test with separate estimates of variance, the null hypothesis could be rejected with $p < 0.01$, indicating that the groups of subjects were distinguishable on the basis of this factor as well. The results followed the trend of scores that were obtained in study I for each of the three factors.

It is interesting to note that though the trend of scores and the distinguishability among the subjects were similar, the comparison of scores revealed that the EUP group in the second study, on average, had lower scores for the factor AFSR, indicating that the group showed a better sense of application focus and a lower support requirement. A possible explanation for this is that the students in the second EUP group all were seniors, as compared with the EUP group in study I, in which some students were juniors and some were sophomores. The difference in scores was negligible between participants in the NPEU group of study I and those of study II. The comparison for factor OOC revealed the subjects from the two studies to be fairly similar in their outlook on computers. For the third factor, TR, there was a greater difference between the study subjects in the NPEU groups than in the EUP groups.

As presented, the results allowed the two groups of follow-up study users to be considered distinguishable, based on the Rockart and Flannery [1983] end user classification scheme. The scores indicated that the participants in the EUP group were more advanced end users than were the participants from the NPEU group.

6.3 Effectiveness Questionnaires – Testing

The development of the effectiveness questionnaires was discussed in the previous chapter. The present discussion concentrates on the validation of these instruments, focusing on the reliability and internal consistency of the factors extracted. The analysis performed on the data was similar to that performed in study I to ensure that the results will be comparable.

The reliability was measured using the Cronbach's α coefficient. As discussed in chapter 5, the two effectiveness questionnaires had 20 questions in common. It was these 20 questions that were used for measuring the effectiveness of the process. The remainder of the questions were process-specific, i.e., they were used for measuring the other characteristics of the two processes. The 20 common questions were tested to determine if the use of a single score to represent the construct being measured, namely consultation effectiveness was possible. The test of internal consistency produced a Cronbach α of 0.84 for the set of 20 questions. This level was higher than the level of 0.75 that was set a priori at the outset of the experiment. The selection of the 0.75 level was based on other similar experimental studies. The

Consultation Effectiveness – the composite score		
Cronbach α	Number of cases	Number of items
0.8367	100	20
Factor: User Satisfaction		
0.9045	100	12
Factor: Task Basis for Effectiveness		
0.4025	100	5
Factor: Recommendation Basis for Effectiveness		
0.5908	100	5

Table 6.6: Reliability Scores for Factors Extracted from the Effectiveness Questionnaires

level of internal consistency for data in the present study also was higher than the results of study I which produced a Cronbach α of 0.75.

Factor analysis (Table 6.7) was conducted on the questions used in measuring consultation effectiveness. Each of the factors constituting consultation effectiveness was analyzed by comparing it with the study I results.

Based on the results, it was decided that the questions included in each factor be kept identical to the groupings made in study I. The data allowed this if multiple loading of a question was done. Since multiple loading had been done in study I to ensure logical validity, this procedure also was undertaken for the second data set (Table 6.8). It is important, however, to note that the factor loadings for the various items were different between the two studies.

Allowing factor compositions to be maintained as identical between study I and study II made it necessary to check the reliability of the different factors prior to using the data from the second study in any analysis (Table 6.6). The first

VARIABLE	COMM	FACTOR	EIGENVALUE	% OF VAR	CUM PCT
Question 1	.70701	1	8.06795	40.3	40.3
Question 2	.16077	2	1.45852	7.3	47.6
Question 3	.45106	3	1.24831	6.2	53.9
Question 4	.49078				
Question 5	.69918				
Question 6	.47381				
Question 7	.48757				
Question 8	.59212				
Question 9	.56353				
Question 10	.62084				
Question 14	.63907				
Question 15	.53612				
Question 18	.60085				
Question 17	.68407				
Question 13	.59985				
Question 21	.42398				
Question 19	.44684				
Question 22	.39268				
Question 23	.57029				
Question 24	.63434				

Table 6.7: Effectiveness Questionnaire: Initial Factor Analysis

ITEMS	FACTOR 1	FACTOR 2	FACTOR 3
Q24	.79236		
Q1	.77533		
Q23	.64288		
Q17	.64162	.43913	
Q18	.64098	.42780	
Q13	.62139	.46230	
Q19	.57935		
Q5	.52504		.51764
Q7	.47742	.45914	
Q4	.45747		
Q8		.76654	
Q15		.63960	
Q14	.44383	.61362	
Q22		.50768	
Q9			-.72931
Q21			.63952
Q6			.57254
Q10	.44007		.53583
Q3			-.51984

Table 6.8: Effectiveness Questionnaire: Factor Loadings of Questionnaire Items

factor, user satisfaction, had the highest degree of internal consistency, reflected by a Cronbach α score of 0.90. This was similar to the results from the first study. The other two factors extracted had relatively lower reliability scores; both had Cronbach α scores lower than 0.75. The second factor, task basis for effectiveness, had a reliability score of 0.40, and the third factor, recommendation basis for effectiveness, had a score of 0.59. It is, however, important to note that the reliability scores for the three factors followed the trend observed in study I, where the scores for the three factors were 0.87, 0.77, 0.33 respectively. Using the reasoning from the first study, namely that the factors extracted were consistent with the system effectiveness measurement literature, a factorwise evaluation of consultation was attempted.

6.4 Results of the Study

The subjects for the experiment, as mentioned earlier, were students from the College of Business and Public Administration at the University of Arizona. The subjects were chosen from the same population of students as in study I. Furthermore, both study I and study II were conducted in the Fall 1987 semester, which ensured that there was no subject duplication, i.e., no student participated in the experiment more than once. Study I was conducted in the second week, while study II was conducted in the tenth week of the fall semester. The choice of subjects was designed to maximize the comparability between the results of study I and study II. Table 6.9 presents the distribution of the subjects for the present study.

The task used in the second study was the same as that in study I. Maintaining the same task for the two studies would not affect the results, since the subjects were different. Further, using the same task enhances the comparability of the results. The task, as described in an earlier chapter, was the selection of software for a report writing assignment. The experience from study I indicated that the subjects were

CLASS	FREQUENCY	%	Cumulative %
Freshman	3	3.0	3.0
Sophomore	10	10.0	13.0
Junior	35	35.0	48.0
Senior	52	52.0	100.0
TOTAL	100	100.0	

Table 6.9: Distribution of Participants (Study II)

able to identify readily with the requirements described in the task. The description of the task to the subjects and other procedures used in the administration of the second experiment also were identical to those used in study I and are therefore not discussed here.

The research questions being pursued in this follow-up study were the same as those in study I, namely:

1. Does a relationship exist between the independent variables, the channel for getting assistance with software selection–ICE or CMI-Consultant–and the class of end user EUP or NPEU–as measured by the dependent variable consultation effectiveness?

2. If the relationship exists, what is its nature?

Similar to study I, A multivariate analysis of variance (MANOVA) was conducted for the three factors constituting "consultation effectiveness", namely user satisfaction, task basis for effectiveness and the recommendation basis for effectiveness. The MANOVA results for study II presented in Table 6.10 and Table 6.11 were similar to those of study I for the type of process, i.e., ICE or CMI Consultant. The results indicated that the type of process has a significant impact on "consultation

	Value	Exact F	Hypoth. DF	Error DF	Significance of F
Type of Process (ICE/CMI)	0.692	21.673	3	94	0.000
Type of User (NPEU/EUP)	0.043	1.348	3	94	0.264
Interaction	0.023	0.745	3	94	0.528

Table 6.10: Hotellings Multivariate Test of Significance – Study II

effectiveness", indicated by an F statistic of 21.67 (d.f. = 1, 96). However, unlike study I, the effect of the type of end-user, NPEU or EUP, on consultation effectiveness was not significant with an F-statistic of 1.348 (d.f. = 1, 96). The interaction effect of the type of end-user and the type of process was also not significant.

The univariate F-tests Table 6.11 were conducted for each of the three factors constituting effectiveness namely, type of process, type of end-user and the interaction effect. The results from this test indicated that only the type of process had a significant impact on consultation effectiveness.

As in study I, the MANOVA results were followed up by a factorial analysis of variance (ANOVA) for each of the three factors constituting consultation effectiveness. The results of the factorial ANOVA are presented in the following sections.

A 2X2 factorial analysis of variance design was used to study the main effects, i.e., individual effects of the independent variable on the dependent variable, and the interaction effect, i.e., the joint effect of the two independent variables on the dependent variable. Similar to the procedure followed in the first study, the three factors constituting consultation effectiveness are analyzed before analyzing the composite score for consultation effectiveness.

Variable	Hypoth. SS	Error SS	Hypoth MS	Error MS	F	Sig. of F
Type of Process						
User satisfaction	13.651	55.491	13.651	.578	23.617	.000
Task basis	6.271	26.462	6.271	.275	22.752	.000
Recommendation basis	3.487	50.454	3.487	.525	6.635	.012
Type of User						
User satisfaction	.341	55.491	.341	.578	.590	.444
Task basis	.398	26.462	.398	.275	1.446	.232
Recommendation basis	.003	50.454	.003	.525	.005	.939
Interaction						
User satisfaction	.172	55.491	.172	.578	.298	.586
Task basis	.426	26.462	.426	.275	1.546	.217
Recommendation basis	.590	50.454	.590	.525	1.124	.292

Table 6.11: Univariate F-tests (d.f. = 1, 96) – Study II

Source of Variation	DF	Mean Square	F	Signif of F
MAIN EFFECTS				
I. PROCESS ICE vs CMI-Consultant	1	13.492	23.342	0.000
II. USER CLASS NPEU vs EUP	1	0.267	0.462	0.498
INTERACTION EFFECT				
Process - User class	1	0.172	0.298	0.586

Table 6.12: Results of the 2-way factorial analysis of variance for user satisfaction

Process	ICE = 1.80 (n = 57)	CMI Consultant = 2.89 (n = 43)
User Class	NPEU = 2.14 (n = 54)	EUP = 2.10 (n = 46)

Table 6.13: User Satisfaction – Cell Means

6.4.1 User Satisfaction

With "user satisfaction" as the dependent variable, the first hypothesis tested was:

H4: User satisfaction is independent of the type of consultation method (ICE or CMI-Consultant) used.

The results (Table 6.12, Table 6.13 and Table 6.14) showed the F-statistic as 23.342 (df = 1, 97) causing the null hypothesis to be rejected with $p < 0.01$. This allowed the conclusion that the user satisfaction was dependent on the type of consultation method.

The cell means table (Table 6.13 and Table 6.14) indicates that users of ICE were more satisfied than the users who consulted with the CMI-Consultants (Satisfaction score: ICE = 1.80, CMI-Consultants = 2.54). The results, though

	NPEU	EUP
ICE System	1.78 (n = 26)	1.82 (n = 31)
CMI Consultant	2.46 (n = 28)	2.67 (n = 15)

Table 6.14: User Satisfaction – 2X2 Cell Means

similar to those from study I, were more pronounced. Since the scales used were ordinal, no subjective value could be placed on the scores, but rank ordering were acceptable. Comparison of the results from study I, showed the users of ICE in the second study to have a higher level of satisfaction, 1.80 vs 1.85. Conversely, for the users dealing with the CMI Consultants, the level of satisfaction dropped from 2.38 to 2.54, causing the effect to become larger.

In testing for the second main effect, the hypothesis was:

H5: User satisfaction is independent of the end-user classification.

From Table 6.12, the F-statistic was 0.46 (df = 1, 97) resulting in p = 0.49. The null therefore could not be rejected, leading to the conclusion that user satisfaction for the consultation process was independent of the class of the end user. This finding was interesting, as it contradicted the finding from the first study, which rejected the null. One possible explanation for this effect can be attributed to a "maturation effect" [Campbell and Stanley, 1966] for the CMI-Consultants. This is further discussed in a later section.

As with the results for consultation effectiveness, the interaction effect was statistically insignificant with an F-statistic of 0.30 (df = 1, 97), resulting in p = 0.58. As was mentioned in the discussion of consultation effectiveness, a lack of interaction further strengthened the findings in the main effect.

Source of Variation	DF	Mean Square	F	Signif of F
MAIN EFFECTS				
I. PROCESS				
ICE vs CMI-Consultant	1	5.934	21.528	0.000
II. USER CLASS				
NPEU vs EUP	1	0.271	0.984	0.324
INTERACTION EFFECT				
Process - User class	1	0.426	1.546	0.217

Table 6.15: Results of the 2-way Factorial Analysis of Variance for Task Basis for Effectiveness

6.4.2 Task Basis for Effectiveness

Prior to analyzing the task basis for effectiveness, it should be mentioned that, because the internal consistency (Cronbach's α coefficient) of this factor was relatively low, the findings should be weighted less. The hypothesis for the first main effect was:

H6: The task basis for effectiveness is independent of the type of consultation method (ICE or Consultant) used.

The results (Table 6.15, Table 6.16 and Table 6.17), indicated the F-statistic as 21.53 (df = 1, 97) causing the null hypothesis to be rejected with $p < 0.01$. It could be concluded, then, that the task basis for effectiveness was dependent on the type of consultation method and, further, that users of ICE indicated a higher level of effectiveness as compared with users who consulted with the CMI-Consultant (task basis for effectiveness scores: ICE = 3.71; CMI-Consultants = 4.19).

Process	ICE = 3.71 (n = 57)	CMI Consultant = 4.19 (n = 43)
User Class	NPEU = 3.91 (n = 54)	EUP = 3.92 (n = 46)

Table 6.16: Task Basis for Effectiveness – Cell Means

	NPEU	EUP
ICE System	3.71 (n = 26)	3.70 (n = 31)
CMI Consultant	4.09 (n = 28)	4.36 (n = 15)

Table 6.17: Task Basis for Effectiveness – 2X2 Cell Means

The second hypothesis for the task basis for effectiveness was:

H7: Task basis for effectiveness is independent of the class of the end user.

The results (Table 6.15) indicated that the F-statistic was 0.98 (df = 1, 97), and therefore the null hypothesis could not be rejected ($p = 0.32$), allowing the conclusion that the class of the end user and the task basis for effectiveness were not dependent.

The interaction effect between the two independent variables, however, was not significant, F-statistic = 0.43 (df = 1, 97) with $p = 0.21$. This result was different from the finding in the first study, which found the interaction to be statistically significant.

6.4.3 Recommendation Basis for Effectiveness

The results of the test using recommendation basis for effectiveness as the dependent variable are shown in Tables 6.18, Table 6.19 and Table 6.20. The first hypothesis being tested was:

Source of Variation	DF	Mean Square	F	Signif of F
MAIN EFFECTS				
I. PROCESS ICE vs CMI-Consultant	1	3.153	6.000	0.016
II. USER CLASS NPEU vs EUP	1	0.008	0.014	0.905
INTERACTION EFFECT				
Process - User class	1	0.591	1.124	0.292

Table 6.18: Results of the 2-way Factorial Analysis of Variance for Recommendation Basis for Effectiveness

Process	ICE = 2.12 (n = 57)	CMI Consultant = 2.49 (n = 43)
User Class	NPEU = 2.32 (n = 54)	EUP = 2.23 (n = 46)

Table 6.19: Recommendation Basis for Effectiveness – Cell Means

H8: Recommendation basis for effectiveness is independent of the type of consultation method (ICE or Consultant) used.

The results indicated the F-statistic as 6.00 (df = 1, 97) causing the null hypothesis to be rejected with p = 0.01. The cell means suggested that ICE is preferred by the users (recommendation basis for effectiveness score: ICE = 2.12; Consultant = 2.49) when the recommendation basis for effectiveness was used for comparing ICE with the CMI-Consultants.

The second hypothesis considered was:

H9: Recommendation basis for effectiveness is independent of the class of end-user.

	NPEU	EUP
ICE System	2.20 (n = 26)	2.05 (n = 31)
CMI Consultant	2.43 (n = 28)	2.60 (n = 15)

Table 6.20: Recommendation Basis for Effectiveness – 2X2 Cell Means

The results showed the F-statistic at 0.01 (df = 1, 97). This did not allow us to reject the null because $p = 0.90$. It therefore was concluded that the class of the end user did not affect his/her evaluation of the process on the basis of the recommendation basis for effectiveness.

The interaction effect between the two independent variables was insignificant, with the F-statistic at 0.59 (df = 1, 97), resulting in $p = 0.29$.

It is interesting to note that study I results did not find statistical significance for either the main effects or the interaction when the recommendation basis for effectiveness was used as the criterion for judging the process.

6.4.4 Consultation Effectiveness

Consultation effectiveness was measured by the average of the composite scores obtained from the effectiveness questionnaires. The relatively high Cronbach α score of 0.83 allows the of composite score. The effectiveness scores were arranged such that lower scores represented a higher level of consultation effectiveness and vice versa.

The first main effect was tested by evaluating the following hypothesis:

H10: Consultation effectiveness is independent of the type of consultation method (ICE or Consultant) used.

Source of Variation	DF	Mean Square	F	Signif of F
MAIN EFFECTS				
I. PROCESS ICE vs CMI-Consultant	1	9.079	28.340	0.000
II. USER CLASS NPEU vs EUP	1	0.126	0.393	0.532
INTERACTION EFFECT				
Process - User class	1	0.316	0.986	0.323

Table 6.21: Results of the 2-way Factorial Analysis of Variance for Consultation Effectiveness

Process	ICE = 2.33 (n = 57)	CMI Consultant = 2.94 (n = 43)
User Class	NPEU = 2.61 (n = 54)	EUP = 2.57 (n = 46)

Table 6.22: Consultation Effectiveness – Cell Means

The results indicated (Table 6.21) the F-statistic to be 28.34, allowing for the null hypothesis to be rejected (df = 1, 97) with $p < 0.01$. This suggested that the consultation effectiveness is dependent on the type of consultation method used.

The cell means presented (Table 6.22 and Table 6.23) indicated that the consultation effectiveness was higher for ICE, regardless of the classification of the user (Effectiveness score: ICE = 2.33; CMI-Consultant = 2.94) The direction of the results was similar to that obtained in study I. The second study scores, however, made a stronger case for the ICE system. The F-statistic was substantially larger for this study when compared with the F value in study I, 12.87 (df = 1, 97).

The hypothesis evaluating the second main effect used the second independent variable, i.e., it judged the consultation effectiveness of the process from the

	NPEU	EUP
ICE System	2.34 (n = 26)	2.32 (n = 31)
CMI Consultant	2.86 (n = 28)	3.07 (n = 15)

Table 6.23: Consultation Effectiveness – 2X2 Cell Means

standpoint of the type of user. It was hypothesized that:

H11: Consultation effectiveness is independent of the end-user classification.

The results (Table 6.21), showed the F-statistic to be 0.39 (df = 1,97). For this F value, the null hypothesis could not be rejected since the p = 0.53. This led to the conclusion that the level of consultation effectiveness of the software recommending process was not dependent on the class of the end-user.

6.5 Summary of results from Study I and Study II

The results obtained in study II followed the trend of the first study, showing consistency in the results. The consistency of the results across the two studies was verified by conducting a MANOVA. The results of the MANOVA (Table 6.24 and Table 6.25) indicate that the effect on the three factors constituting consultation effectiveness, namely, user satisfaction, task basis for effectiveness and the recommendation basis for effectiveness do not vary significantly across the two studies, the F-statistic for each of the factors being 0.768, 0.473 and 0.141 repectively (d.f. = 1, 175). The results for the composite score follow similar trends across the two studies as well, with the F-statistic equal to 0.25319 (d.f. = 1, 175).

The results of a MANOVA (Table 6.26 and Table 6.27) comparing the two studies indicated the three-way interaction between "the type of user, type of pro-

Variable	Value	Exact F	Hypoth. DF	Error DF	Significance of F
Study I versus Study II	0.021	0.926	4	172	0.450

Table 6.24: Hotellings Multivariate Test of Significance Across Study I and Study II

Variable	Hypoth. SS	Error SS	Hypoth MS	Error MS	F	Sig. of F
Study I versus Study II						
Consultation effectiveness	.071	49.608	.071	.283	.253	.615
User satisfaction	.416	94.897	.416	.542	.768	.382
Task basis	.110	40.806	.110	.233	.473	.492
Recommendation basis	.065	80.972	.065	.462	.141	.707

Table 6.25: Univariate F-tests (d.f. = 1, 175) – Study I versus Study II

Effect	Value	Exact F	Hypoth. DF	Error DF	Significance of F
Type of process by Type of user by Study	0.024	1.072	4	172	0.372
Type of process by Study	0.211	9.089	4	172	0.000
Type of user by Study	0.140	6.026	4	172	0.000

Table 6.26: Hotellings Multivariate Test of User, Process, Study Interactions

cess and the two studies" was not significant (p = 0.372), with the Hotellings multivariate test resulting in an F-statistic of 0.24 (d.f. = 1, 175). Furthermore, the univariate F-tests for the interaction was not significant for any of the factors constituting consultation effectiveness. It was not significant for the composite score of consultation effectiveness as well (Table 6.27).

There was a significant ($p < 0.01$) Interaction effect for "type of user by type of study" Table 6.26). This is consistent with our earlier discussion (Section 6.2) that there was a distinction in the users of the two studies on the basis of the factor AFSR. The univariate F-tests indicated statistical significance at the α level of 0.05 for user satisfaction, task basis for effectiveness and the composite score for consultation effectiveness. The effect for the factor recommendation basis for effectiveness was however not statistically significant with p = 0.831.

The interaction effect for the two studies and the type of process was significant with $p < 0.01$ (Table 6.26). The univariate F-tests indicated that the effect "type of process by type of study" was significant for the composite score for consultation effectiveness, as well as for the factors user satisfaction, and task basis for

Variable	Hypoth. SS	Error SS	Hypoth MS	Error MS	F	Sig. of F
Type of User by Type of Process by Study						
Consultation effectiveness	.474	49.608	.474	.283	1.675	.197
User satisfaction	.301	94.897	.301	.542	.555	.457
Task basis	.556	40.806	.556	.233	2.386	.124
Recommendation basis	.686	80.972	.686	.462	1.483	.225
Type of Process by Study						
Consultation effectiveness	5.374	49.608	5.374	.283	18.959	.000
User satisfaction	9.790	94.897	9.790	.542	18.055	.000
Task basis	1.975	40.806	1.975	.233	8.471	.004
Recommendation basis	1.368	80.972	1.368	.462	2.957	.087
Type of User by Study						
Consultation effectiveness	1.298	49.608	1.298	.283	4.580	.034
User satisfaction	6.029	94.897	6.029	.542	11.119	.001
Task basis	1.002	40.806	1.002	.233	4.299	.040
Recommendation basis	.021	80.972	.021	.462	.045	.831

Table 6.27: Univariate F-tests (d.f. = 1, 175) for User, Process, Study Interaction

	AFSR	TR	OOC
AFSR	1.0	0.73	0.63
TR		1.0	0.70
OOC			1.0

Table 6.28: Correlation Matrix (Study I)

	AFSR	TR	OOC
AFSR	1.0	0.67	0.64
TR		1.0	0.65
OOC			1.0

Table 6.29: Correlation Matrix (Study II)

effectiveness. The effect was not significant however for the recommendation basis of effectiveness at α level of 0.05 as p = 0.87. This finding further indicates that the type of process did have affect the end user's perception of effectiveness.

As for the factors used for classifying the end-users: AFSR, OOC, and TR, a correlation test indicated that there was a relatively high degree of intercorrelation between the factors (Table 6.28 and Table 6.29). This indicated that the three factors overlapped in accounting for the differences between the end-users. Further, the zero order correlations between the user classification factors AFSR, OOC, and TR and the measures for effectiveness indicated that though the classification factors significantly assisted with distinguishing the two sets of end-users, they did not directly account for the differences in the users perception of consultation effectiveness.

6.6 Research Validity

The results from the present data set follow a definite trend. The main effect considered, using the independent variable "consultation method," consistently rejected the null hypothesis with $\alpha = 0.05$. This indicated a strong dependence of effectiveness, and the factors constituting it, on the type of process used for consultation–ICE or CMI-Consultant. It is further interesting to note that the ICE system consistently scored higher with the users for consultation effectiveness as well as on the factors constituting consultation effectiveness. On the other hand, the main effect for the independent variable "class of user," did not have a significant impact on the dependent variable. This finding was observed when the test was conducted using "consultation effectiveness" as the dependent variable and when the factors constituting consultation effectiveness were used as well. This indicates that the users perceived ICE as a more effective consultation process, regardless of the level of the user.

The results of this experiment and their implications were not absolutely conclusive; there may be some alternative explanation for them. To strengthen the validity of the findings, some alternative explanations were addressed. The measures taken to minimize their effects will now be discussed.

6.6.1 Task

The first possible rival hypothesis was that the task for the experiment (document preparation) had affected the results of the experiment. Each consultant's having an area of expertise that might be unrelated to the task, could have reduced his or her effectiveness in addressing a task requiring detailed specialized knowledge. This could have affected the outcome of the experiment.

The choice of tasks was based on a study by Amoroso and Cheney [1987] indicating that document preparation accounts for 90% of end-user computing software requirements. Other studies similarly have indicated that document preparation software is one of the major categories supported by information centers [Brancheau et al., 1985]. It therefore was assumed that document preparation is so commonly recurring a problem that consultants should previously have faced it frequently enough to be comfortable with the task. Furthermore, the task met with the approval of the CMI manager, who considered it understandable and meaningful in the context of the CMI operations.

6.6.2 Communication

A second possible rival hypothesis concerned the users involved in the study. These users were required to verbalize their requirements to the consultants. To do so necessitates some basic understanding of computer terminology and the ability to translate the user's needs into the terminology with which the CMI consultants are familiar. The inability to express their requirements adequately could possibly have caused the users to rate the CMI-consultant process as less effective. Use of the ICE system eliminated the need for verbalizing because the the system prompted the users with the questions and then allowed them to choose from a set of options.

This rival explanation, though important, is not really controllable. Subjects for this experiment were chosen from courses offered by the MIS department, and as such can be assumed to have possessed some basic knowledge of the terminology.

6.6.3 History and Maturation

A third rival hypothesis was concerned with the history and maturation effect [Campbell and Stanley, 1966]. History is concerned with events occurring between measurements. Maturation, on the other hand, is the effect of time on the variable. History/maturation was a problem for the CMI consultants, who had to deal with the same case more than 40 times. Even with the cases having been dispersed among seven consultants, effects of repetition on their responses could adversely have affected the results of the experiment.

To reduce this possibility, the cooperation of the CMI-consultants was requested. The consultants were asked to treat each student as a new case, even though the requirements being presented were similar. This request would not affect the results of the study as the study focuses on the process and not the performances of individual consultants. Furthermore, it was the perspective of the user that was being measured, not that of the consultants. As an additional precaution, the consultation sessions were tape recorded. The recordings revealed that each consultation session took approximately 13 minutes. The time required for a consultation session did not vary substantially from the first to the last session for any consultant.

6.6.4 External Validity

External validity focuses on the question of generalizability. Conducting the CMI-ICE experiments in a university environment made the results obtained generalizable to other information centers and the end users they support.

Although the users of the CMI facility were students and faculty of the business college, as well as staff from other departments around the university, the CMI is a separate entity that operates like any other information center described in the literature. The wide range of expertise of the consultants is similar to that in information centers set up for both the IBM/Endicott and the IBM/Tucson facilities. Generalizing the results of this experiment was however, limited by the choice of subjects for the study. As mentioned earlier, the study used only MIS students, whose skill level and the aptitude for interacting with the computer would be expected to be higher than those of end users in the "real world" because of the nature of the course work they had taken and their exposure to computer technology. Rockart and Flannery's classification of end users provides some guidance for the type of end users to whom the results can be extended.

Chapter 7

Summary, Contributions and Future Research

7.1 Summary

Implementation of the ICE system can be anticipated to improve IC operations in several respects. Being able to query the knowledge base containing the expertise of senior level consultants gives the user access to assistance twenty-four hours a day, seven days a week. This feature addresses the problems of its being impossible for senior level experts to be available at all times in the Information Center, as well as the difficulty of attracting senior level experts to work in the IC at any time. In addition, consistent advice will be supplied by the expert system. Numerous consultants, at various levels of expertise, rarely will provide a common solution to a stated problem. The ICE system also facilitates the implementation of corporate policy by providing a single source for dissemination of IC help. Finally, the expert system provides an excellent tool for training new IC consultants. The knowledge base can be queried by new consultants to help them learn the structured process for solving a typical user's questions.

7.1.1 Development effort

The research questions stated in chapter 1 presented concerns related to modeling the knowledge of the IC consultants and to rapidly changing software resources. These concerns were addressed in both the knowledge acquisition and the knowledge representation efforts. To address the concerns of modeling the knowledge of IC consultants, the development effort recognized the importance of using multiple experts, as well as multiple sources of knowledge in the development of a knowledge-based system. Identifying a single domain expert may not always be feasible. Furthermore, when attempts are made to apply expert system technology to more sophisticated problems, restricting the knowledge acquisition activity to a single expert could prove counterproductive. Experts are by definition "very knowledgeable about only a small subset of the tasks in the domain" [McDermott, 1984]. Using multiple experts to build an expert system can avoid the biases that may result from using a single expert. As the scope of tasks is enlarged, it obviously will become necessary to use multiple experts in the development of knowledge based systems. Alternate sources of knowledge such as trade journals, software documentation, and discussions with the users also proved invaluable to the development effort.

With regard to the design criteria of maintainability, transportability, and flexibility, the concepts of context dependency, and homeostatic processes or knowledge stability [Krcmar, 1985, Little, 1986] proved very useful. It should be recalled that the concept of context dependency addresses the degree of universality of certain knowledge and that knowledge stability refers to the changes in knowledge over time.

The partitioning of the knowledge base elements designed for the ICE system exemplifies a technique that can be applied to other knowledge-based system development having similar design criteria. To address the issue of context dependency and knowledge stability the rules and parameter constituting the knowledge base were partitioned. The parameters were divided into two categories: Dialogue Control Parameters and the Attribute Setting Parameters. This division permitted control of dialogue by utilizing two types of rules: Inference and Monitor. A detailed discussion of the utility of this approach was discussed in chapter 4.

7.1.2 Implications Based on Experimental Evaluation

The experiment conducted focused on the "consultation effectiveness" construct to verify the effectiveness of the system. The main finding of the experiment was that the type of consultation method and consultation effectiveness were dependent, leading to the conclusion that the process has an impact upon the user's perception of consultation effectiveness. The processes compared were the use of ICE and of the CMI consultants to obtain advice on computer software. The subjects using the ICE system reported higher consultation effectiveness than did those using the CMI-Consultants. It should be noted, however, that the effectiveness measure for the consultants was an average score for several consultants. It is not implied that the system is more effective than the assistance of any of the consultants on an individual basis. The result points out, nevertheless, that, on average, the users perceived the process using the ICE system to be the more effective of the two processes.

The implications of the results of this experiment are that: 1. ICE is an effective support for the consulting process in information centers, and 2. the appropriate role of ICE is as a front-end support to the consulting process. ICE's having been demonstrated to be an effective support to the consulting process leads to consideration of the possibility that the consultation process can be altered in such a way that the users will consult with the system before approaching the information center. Such a consultation, as was discussed in an earlier chapter, is possible in the privacy of the user's workspace. Using the tracking subsystem (also discussed earlier), consultants of the information center will be able to monitor the users who take advantage of the ICE facility and provide an appropriate followup if indicated to be needed.

The use of ICE as a front end to the consulting process will enable information center consultants to restrict their efforts to helping exceptional cases, i.e., cases that are not adequately covered by the ICE system. Users presenting such cases will be directed by ICE to a specialist consultant who is an expert in the type of problem.

This alteration in the consulting process could enhance the effectiveness of the software selection process for users and also should prove helpful to the consultants by reducing the number of consultations requiring their attention. Furthermore, consultants would be consulted only regarding their areas of expertise instead of having to make educated guesses regarding subjects with which they are less familiar.

The argument for the feasibility of employing ICE as a front end to the consulting process is strengthened by the failure to find statistical significance for the second main effect, namely, that consultation effectiveness is based upon the class of end user. It was shown in the experiment that ICE proved more effective than the average consultant, regardless of the class of the end-users. Two classes of end users were considered. Further, since interaction effects between the independent variables were were found to be insignificant, there is a strong indication that ICE will be successful as a front end to the consulting process, regardless of the class of end-user.

7.2 Contributions

The contributions of the research reported in this dissertation are divided into two categories: the software engineering effort and the empirical evaluation effort.

7.2.1 Development of the ICE system

The ICE system provides a modular architecture that may be extended at a later time to provide more complete support to the information center. The system does not, however, attempt to provide comprehensive support for the various services provided by information centers but concentrates on the selection of software based on the user's background and his/her current requirements.

The design issues addressed in the development of ICE, namely, maintainability, transportability, and flexibility are well recognized and appreciated in the more traditional software engineering efforts. The research related to knowledge-based systems recognizes the importance of these design issues as evidenced by the discussions of the knowledge base – database coupling, and concerns about the static and dynamic aspects of knowledge.

The ICE system addresses the above mentioned issues by incorporating them in the architecture of the system. The concern for maintainability is addressed in two ways. First, the the rule base is used to control the dialogue and the extraction of the requirements of the users. The solutions, i.e., the software packages that are to be recommended, are not embedded in the rules constituting the knowledge base. Second, the maintenance subsystem, MTICE, allows for additions, updates, and deletions of the software resources supported by the ICE system.

The features supporting the maintainability of ICE also allow the system to be transportable. Different information centers support different sets of end users and, as a result, the software inventories of the information centers vary substantially. By allowing the system to be maintainable, ICE can be transported to different information centers and make recommendations based on the the requirement of the center for which it provides support.

7.2.2 Development and Validation of Classification and Effectiveness Instruments

The increased importance of computer technology, and the involvement of end users in both the development and usage of the technology research concerning the impacts of technology on the users, will require evaluation of the effects of technology on various types of users.

Both the classification and the effectiveness questionnaires used in the study were validated based on three types of validity identified by Kerlinger [1986]: content, criterion related, and construct. Content validation is based on judgment in which, "alone or with others, one judges the representativeness of the items" [Kerlinger, 1986]. Each of the questionnaires was reviewed by several faculty members at the University of Arizona to ensure adequacy of the items used.

Criterion-related validity is studied by "comparing the test or scale scores with one or more external variables, or criteria ...the single greatest difficulty of the criterion-related validation is the criterion" [Kerlinger, 1986]. Undergraduate students at different stages in their program of study were selected as subjects, with the criterion for classification being an assumption that the more senior students would score more positively on the scale used to determine the various factors. This assumption that the factors being measured would be affected by the educational experience of the student, as the results reported earlier indicate, proved to be the case. In developing the effectiveness questionnaires, the questions were developed around factors identified by earlier studies (see Chapter 2) related to effectiveness. In the absence of other validated instruments for testing effectiveness of knowledge-based systems, instruments used in testing the effectiveness of the more traditional information systems were used in developing the effectiveness questionnaires.

Construct validation focuses on the "property or properties (that) can 'explain' the variance of the tests" [Kerlinger, 1986]. To ensure construct validity, both the classification and the effectiveness questionnaires were factor analyzed, and each of the factors extracted was evaluated in terms of the construct being studied. A Cronbach-α reliability test also was conducted to verify whether or not the items contributing to a particular construct did in fact assist in explaining the variance.

The use of two independent but related studies made it possible for the first study to be used as a control for the second study. The validation of the instruments was helped by the second study, which corroborated the results of the first study.

7.3 Future Research

Two major areas for extending the research reported in this dissertation are being considered: first, extensions to the ICE system to provide additional knowledge based support to end users, and second, studying the impact of knowledge based systems in general on organizations.

7.3.1 Extensions to the ICE systems

An information center provides comprehensive service to an end-user community. In some organizations, functions of the information center are partitioned into several divisions such as help desk, productivity center, or information center to serve different needs of end users. The development of the ICE system in its present form provides knowledge-based support to the consulting task of making software recommendations. Development of planned extensions will provide support for tracking of end-user needs, help desk support and the distribution of the information center resources being evaluated.

Tracking: The tracking program will provide two services to the users of ICE. First, the user will receive a hard copy of the recommendations made by the system. The user then may use the suggestions made by the system or, alternatively, may acquire a second opinion from the IC consultant. Second, since the program will keep track of the users and the recommendations made to them, the user can be kept up-to-date on the software used by forwarding any notices of updates concerning that software. The tracking facility also will assist the information center staff by collecting data on users of the information center and on the resources of the IC to meet the computing requirements of the users. The information center's ability to manage its software resources will be enhanced by the tracking report. The fact that the IC has consistent and complete records of consultation sessions will allow it to evaluate the software inventory and make effective updates to it.

The planned extension to the tracking system will enforce maintenance. An embedded tracking facility can provide a mechanism for ensuring self maintenance. Enforced updating would be triggered whenever a question being asked is not adequately answered. The adequacy of the solution would be tied to some measure of "confidence" in the answer being provided to the user. The tracking facility will examine the question classification and a profile of the consultants, enabling the system to ascertain the "expert" who could best address the question. The design of the system would allow the response of the expert then to be incorporated into the set of solutions supported by the system.

Help Desk: When users approach the IC with specific problems in application development or hardware use, the IC will be able to provide them with troubleshooting expertise. This function of the help desk now is frequently implemented using a telephone hot-line which users can call for help in learning about hardware and software, or using tutorials and other training programs conducted by the IC. It can, however, take as long as 5 to 10 minutes to get a call through.

Distribution: The IC also functions as a center for controlled distribution of end-user tools (both hardware and software). Proposed distribution control will allow information centers to track tool usage and to respond better to the needs of the end-user community. Frequently-used packages can be updated, while packages which are rarely used can be removed from the IC. Thus, the IC's area of responsibility remains manageable, avoiding unrestricted software proliferation and its required support. The distribution of software by the IC will facilitate the management of new releases of software, as well as ensuring the use of legal versions of the software.

7.3.2 Impact of Expert System Technology on Organizations

The impact of expert system technology can be viewed as a sociotechnical change [Dijck,1984] wherein, to understand the outcome, it is necessary to view the technological changes that have been caused as well as the social or human factors.

The introduction of an expert system into an organization can potentially bring about unexpected changes in an organization. Although organizations operate in a dynamic environment where changes are a common feature, changes, by definition, affect the status quo and as such are threatening to all parties concerned.

As expert systems increasingly prove themselves to be viable technology, the impact of expert systems should be studied from the perspective of the individuals constituting the organization, i.e., the micro perspective. Of particular interest are features of expert system technology that have an impact on job satisfaction, communication and the power structure within the organization.

The effectiveness study conducted was from the perspective of the end user. It is recognized that it is the individuals within an organization who really put into effect any change and aid in its success. It is important, therefore, to take into account the reasons individuals resist change. Effectiveness is one such reason. Job satisfaction, communications, and power are three additional factors that can potentially contribute to the acceptance of expert system technology.

7.4 Concluding Remarks

The dissertation has explored the possibility of developing a knowledge-based system to provide support for software selection in information centers. Both software engineering and laboratory experiments were used as part of the research methodology.

The distinguishing features of this system are the three design issues that were incorporated in its architecture. The prototype system, ICE, is presently implemented in five information centers at three separate locations. The research reported both the results of a comparison of recommendations made by ICE and those made by IC consultants and the finding of an examination of the process of validating the recommendations made by the system.

The experiment conducted served a dual purpose. First, it provided a measurement for evaluating the consultation effectiveness of advice-giving, knowledge-based systems. The importance of such an evaluation is well documented, but validated instruments to make such a measurement had not previously been created. Second, the results from the experiment provided data verifying the effectiveness of a knowledge-based system for implementation in an information center setting for the purposes of supporting end-user computing.

Finally, the development of ICE was a team project. The development team included, in addition to myself, Mari M. Heltne, and Minder Chen, all Ph.D. candidates in the MIS department at the University of Arizona. The development was completed under the guidance of Drs. Jay F. Nunamaker, Jr. and Benn R. Konsynski. Development tasks were divided among the three students as follows: Ajay S. Vinze - knowledge base design and development, user interface, validation and verification; Mari M. Heltne - knowledge acquisition, maintenance subsystem and system design, testing maintainability and transportability; and Minder Chen - tool search algorithm and maintenance subsystem. This division is reflected in the focus of this dissertation.

Appendix A

Task for Users Consulting with the CMI Consultants

Background

The popularity of the IBM-PC is caused, to a degree, by the large variety of software available for using the machine. To some users, however, this variety and number of software is confusing.

The College of Business and Public Administration at the University of Arizona setup the Center for the Management of Information (CMI) in 1985. The CMI provides several services to the user community–students and faculty–in the business school. Helping users select software to meet their needs is one such service.

The CMI is staffed with consultants who are knowledgeable about the various software supported by the center. This study aims to determine if the CMI in its present form adequately meets the software selection needs of *real users*

Task

Any college level course requires you to write reports and/or term papers. This assignment asks you to find out software products that will help you with **document preparation** (reports, term papers, etc.). For the purposes of this study, please use the following as guidelines for your report writing needs.

Report Requirements:

- The report must be prepared on a Personal Computer (PC).
- A table of contents must be prepared.
- Spellings must be corrected.
- Footnotes are needed.
- Use of different fonts (e.g. bold face, different sizes/styles of character).
- Access data from your PC disk.

You are required to contact the CMI consultants for assisting you in selecting software for your need. The CMI consultants have their office in BPA 105, and will be available to you during their regular working hours. To avoid confusion, and long lines, please sign the sheet being circulated for a convenient time slot. For the purposes of this study, your discussions with the consultants will be tape recorded. Your identity will, however, remain confidential.

You are required to note the recommendations made by the consultants and turn them in to the instructor. This assignment will be worth 20 points.

Appendix B

Task for Users Consulting with the ICE System

Background

The popularity of the IBM-PC is caused, to a degree, by the large variety of software available for using the machine. To some users, however, this variety and number of software is confusing.

Recently, an expert system–ICE (Information Center Expert) has been developed to assist users with selecting software based on their needs. The system prompts you with questions, and based on your responses, selects software package(s) that will best meet your requirements. This study aims to determine if the system is useful for *real users*

Task

Any college level course requires you to write reports and/or term papers. This assignment asks you to find software products that will help you with **document preparation** (reports, term papers, etc.). For the purposes of this study, please use the following as guidelines for your report writing needs.

Report Requirements:

- The report must be prepared on a Personal Computer (PC).
- A table of contents must be prepared.
- Spellings must be corrected.
- Footnotes are needed.
- Use of different fonts (e.g. bold face, different sizes/styles of character).
- Access data from your PC disk.

The ICE system exists on the IBM 4381 (mainframe), and can be accessed from terminals in BPA-109. The consultant working in BPA-109 will help you logon to the IBM 4381 and start a consultation session with the ICE system. To avoid confusion, and long lines, please sign the sheet being circulated for a convenient time slot.

During your consultation with the ICE system, you will be prompted with appropriate questions. Please read the questions carefully. The recommendations made by the system will be based on your response to these questions.

You are required to note the recommendations made by the system and turn them in to the instructor. This assignment will be worth 20 points.

Appendix C

Instructions for subjects using the CMI consultants

Thank you for participating in my study. You are the group of students who will use the CMI consultants for assisting you with the selection of software.

Background

The College of Business and Public Administration at the University of Arizona setup the Center for the Management of Information (CMI) in 1985. The CMI provides several services to the user community–students and faculty–in the business school. Helping users select software to meet their needs is one such service.

With the large variety of computer software available in the market, choosing software can be quite a problem.

Purpose

This study aims to determine the best possible method for providing assistance with software selection. The CMI is staffed with consultants who are knowledgeable about the various software supported by the center. This study aims to determine if the CMI in its present form adequately meets the software selection needs of *real users*

Task

Your task is to find out software products that will help you in preparing reports or term papers. The details of the task are on the sheet handed out.

The CMI consultants have their office in BPA 105, and will be available to you during their regular working hours. To avoid confusion, and long lines, please sign the sheet being circulated for a convenient time slot. For the purposes of this study, your discussions with the consultants will be tape recorded. Your identity will, however, remain confidential.

Are there any questions regarding the task or the procedure you are required to follow?

I thank you for your cooperation.

Appendix D

Instructions for subjects using the ICE system

Thank you for participating in my study. You are the group of students who will use the ICE (Information Center Expert) system for assisting you with the selection of software.

Background

The College of Business and Public Administration at the University of Arizona setup the Center for the Management of Information (CMI) in 1985. The CMI provides several services to the user community–students and faculty–in the business school. Helping users select software to meet their needs, is one such service.

With the large variety of computer software available in the market, choosing software can be quite a problem.

Purpose

This study aims to determine the best possible method for providing assistance with software selection. Recently, an expert system (ICE) has been developed to assist users with software selection. The system determines your need, and recommends software that will best meet your requirements. For the purposes of this study, you will use this system to assist you with the task given to you.

Task

Your task is to find out software products that will help you in preparing reports or term papers. The details of the task are on the sheet handed out.

The ICE system exists on the IBM 4381 (mainframe), and can be accessed from terminals in BPA-109. The consultant working in BPA-109 will help you logon to the IBM 4381 and start a consultation session with the ICE system. To avoid confusion, and long lines, please sign the sheet being circulated for a convenient time slot.

It is always advisable to do a few experimental consultations. This will help you feel comfortable with the system. Upon completion of your assignment, don't forget to logout.

Are there any questions regarding the task or the procedure you are required to follow?

I thank you for your cooperation.

Appendix E

Classification Questionnaire

MIS Department
University of Arizona
September, 1987

Please note: The information collected from this questionnaire will be used for statistical analysis by the MIS Department research personnel at the University of Arizona. *Your identity will remain confidential.*

Ajay S. Vinze,

Please read all the instructions carefully, and answer all the appropriate questions. Your cooperation is very much appreciated.

Student Name : ____________________________

Matric Number: ___________________________

Course Number : ___________________________

We would like to have some general information about you, and your experience with computers:

1. What is your current class level:

   Freshman _____ Sophomore _____

   Junior _____ Senior _____ Graduate _____

2. What is your major (e.g. Accounting): _______________

3. GPA:

   0-1: _____ 1-2: _____
   2-3: _____ 3-4: _____

   Please respond to the following questions, providing detailed information where appropriate.

4. Have you ever used a computer?

   Yes: _____ No: _____

   If No, please skip to question 11

5. How long has it been since you FIRST used a computer (years)?

   0 - 1: _____ 1 - 2: _____

   2 - 4: _____ 4 - 6: _____ Greater than 6: _____

6. What is the average time that you spent using a computer during the past year (hours per week)?

   0 - 2: _____ 2 - 4: _____

   4 - 6: _____ 6 - 8: _____ Greater than 8: _____

7. How many computer languages/software products have you learned?

   None: _____ 1 - 3: _____

   3 - 5: _____ 5 - 7: _____ More than 7: _____

   If None, please skip to question 11 and continue

8. How many computer related courses (*including computer language, applications and theory courses*) have you taken?

   None: _____ 1 - 3: _____

   3 - 5: _____ 5 - 7: _____ More than 7: _____

9. How many computer related courses (*including computer language, applications and theory courses*) are you currently taking?

None: _____ 1 - 2: _____

3 - 4: _____ 4 - 5: _____ More than 5: _____

10. If you have taken more than one course, please list the courses taken: ....

....

11. Have you ever dealt with Expert Systems?

Yes: _______ No: _______

(*if No, skip to question 13*)

12. In what capacity?

User: _______ Designer: _______ Programmer: _______

Other (please specify) ____________________

13. Are you familiar with the CMI (Center for the Management of Information) in the college of Business and Public Administration at UA?

Yes: _______ No: _______

(*if No, skip to question 17*)

14. Do you know that the consultants in CMI can assist you in selecting software for your computing needs?

Yes: _______ No: _______

15. Have you ever asked the CMI consultants a software related question?

Yes: _______ No: _______

16. How many times have you consulted with the CMI consultants?

None: _____ 1 - 5 times: _____

5 - 10 times: _____ More than 10 times: _____

The following questions are a self evaluation of your skills, abilities and interests in dealing with computers. Please use the scale provided, and CIRCLE the number that most closely reflects your point of view.

PLEASE CIRCLE

17. Do you feel computers:

Enhance creativity	1	2	3	4	5	6	7	Restrain creativity

18. Is it your view that computers are:

Easy to use	1	2	3	4	5	6	7	Difficult to use

19. Based on your experience, do computers:

Enhance Productivity	1	2	3	4	5	6	7	Deter productivity

20. In terms of using a computer, do you consider yourself:

Very knowledgeable	1	2	3	4	5	6	7	Not knowledgeable

21. In your chosen profession, do you think that you will?

Use computers often	1	2	3	4	5	6	7	Never use computer

22. How would you describe your experiences with the use of computers?

Very positive	1	2	3	4	5	6	7	Very negative

23. Describe your quantitative skills?

Very good	1	2	3	4	5	6	7	Very poor

24. How do you feel about learning computers?

Very enthusiastic	1	2	3	4	5	6	7	Not enthusiastic

25. Rate your typing ability?

Expert typist	1	2	3	4	5	6	7	Poor typist

26. Describe your verbal abilities?

Very good	1	2	3	4	5	6	7	Very poor

27. How often do you use a computer?

Seldom	1	2	3	4	5	6	7	Frequently

28. Do you find it easy to learn new software?

Yes	1	2	3	4	5	6	7	No

29. Are computer manuals (user and reference manuals) sufficient for you to learn a new software package?

Yes	1	2	3	4	5	6	7	No

30. How often do you need to consult other people for your computer related questions?

Seldom	1	2	3	4	5	6	7	Frequently

31. How often do you use computer application systems?

Frequently	1	2	3	4	5	6	7	Seldom

32. How often have you developed programs for computer systems?

Frequently	1	2	3	4	5	6	7	Seldom

33. How often do you assist people regarding computers?

Frequently	1	2	3	4	5	6	7	Seldom

34. How often do you ask for help from others regarding computers?

Frequently	1	2	3	4	5	6	7	Seldom

Thank you very much for your time and cooperation.

Appendix F

Effectiveness Questionnaire for Consultants

MIS Department
University of Arizona
Fall Semester, 1987

Please note: The information collected from this questionnaire will be used for statistical analysis by the MIS Department research personnel at the University of Arizona. *Your identity will remain confidential.*

Ajay S. Vinze

Please read all the instructions carefully, and answer all the appropriate questions. Your cooperation is very much appreciated.

Student Name : ---------------------------

Matric Number: ---------------------------

Course Number : ---------------------------

Please use the scale provided, and CIRCLE the number that most closely reflects your point of view.

Question 1-10 pertains to the general nature of the task assigned to you:

PLEASE CIRCLE

1. In general, how would you describe your experience in completing the task assigned?

   Very enjoyable 1 2 3 4 5 6 7 Very frustrating

2. The task was:

   Much too easy 1 2 3 4 5 6 7 Much too difficult

3. How satisfied were you with your performance of the task?

   Unsatisfied 1 2 3 4 5 6 7 Satisfied

4. Was the process for obtaining solutions for your task:

   Efficient 1 2 3 4 5 6 7 Inefficient

5. Was access to assistance for software selection:

   Convenient 1 2 3 4 5 6 7 Inconvenient

6. Time taken for obtaining a solution for your task was:

   Reasonable 1 2 3 4 5 6 7 Unreasonable

7. Did you get to voice all your concerns about the task prior to the selection of the software tool for your task?

   Sufficiently 1 2 3 4 5 6 7 Insufficiently

8. While completing the task I:

   Understood what was needed 1 2 3 4 5 6 7 Was completely confused

9. The time required for preparing for the task was:

   Excessive 1 2 3 4 5 6 7 Minimal

10. While completing the task I was:

    Not frustrated 1 2 3 4 5 6 7 Very frustrated

Questions 11-24 relate to your experience in consulting with the CMI consultants:

11. Did you feel comfortable approaching the CMI consultants?

    Yes 1 2 3 4 5 6 7 No

12. How difficult was it to get access to the CMI consultants?

    Very difficult 1 2 3 4 5 6 7 Very Easy

13. In your opinion, the consultation process was:

    Flexible 1 2 3 4 5 6 7 Inflexible

14. Were the recommendations provided by CMI consultants understandable?

    Yes 1 2 3 4 5 6 7 No

15. In your opinion were the recommendations:

    Simple 1 2 3 4 5 6 7 Complicated

16. Did you feel confident about the solutions offered by the consultants?

    Yes 1 2 3 4 5 6 7 No

17. How confident were you about the recommendations made?

    Very confident 1 2 3 4 5 6 7 Not confident

18. Did you perceive the duration of the consultation to be:

    Sufficient 1 2 3 4 5 6 7 Insufficient

19. The questions asked by the consultants were:

    Meaningful 1 2 3 4 5 6 7 Meaningless

20. In your opinion, the technical competence of the consultants was:

    Sufficient 1 2 3 4 5 6 7 Insufficient

21. The recommendations made were:

    Imprecise 1 2 3 4 5 6 7 Precise

22. The software recommended for the task was:

    Satisfactory 1 2 3 4 5 6 7 Unsatisfactory

23. If faced with a task requiring software support in the future, would you consult with the CMI staff to aid you with your decision?

    Yes 1 2 3 4 5 6 7 No

24. If given the choice between a consultant, and a computerized system to provide you with software recommendation, which would you prefer?

Prefer a consultant	1	2	3	4	5	6	7	Prefer a computerized system

Please use the back of this page to give any suggestions regarding the task, or your experience with the CMI consultants. Any additional comments would be most helpful.

Thank you very much for your time and cooperation

Appendix G

Effectiveness Questionnaire for ICE

MIS Department
University of Arizona
Fall, 1987

Please note: The information collected from this questionnaire will be used for statistical analysis by the MIS Department research personnel at the University of Arizona. *Your identity will remain confidential.*

Ajay S. Vinze

Please read all the instructions carefully, and answer all the appropriate questions. Your cooperation is very much appreciated.

Student Name : ---------------------------

Matric Number: ---------------------------

Course Number : ---------------------------

Please use the scale provided, and CIRCLE the number that most closely reflects your point of view.

Question 1-10 pertains to the general nature of the task assigned to you:

PLEASE CIRCLE

1. In general, how would you describe your experience in completing the task assigned?

   Very enjoyable 1 2 3 4 5 6 7 Very frustrating

2. The task was:

   Much too easy 1 2 3 4 5 6 7 Much too difficult

3. How satisfied were you with your performance of the task?

   Unsatisfied 1 2 3 4 5 6 7 Satisfied

4. Was the process for obtaining solutions for your task:

   Efficient 1 2 3 4 5 6 7 Inefficient

5. Was access to assistance for software selection:

   Convenient 1 2 3 4 5 6 7 Inconvenient

6. Time taken for obtaining a solution for your task was:

   Reasonable 1 2 3 4 5 6 7 Unreasonable

7. Did you get to voice all your concerns about the task prior to the selection of the software tool for your task?

   Sufficiently 1 2 3 4 5 6 7 Insufficiently

8. While completing the task I:

   Understood what was needed 1 2 3 4 5 6 7 Was completely confused

9. The time required for preparing for the task was:

   Excessive 1 2 3 4 5 6 7 Minimal

10. While completing the task I was:

    Not frustrated 1 2 3 4 5 6 7 Very frustrated

Questions 11-28 relate to your experience with the ICE system:

11. Based on your experience, the ICE system was:

    `Easy to learn   1   2   3   4   5   6   7   Difficult to learn`

12. Do you feel confident about using the ICE system:

    `Yes   1   2   3   4   5   6   7   No`

13. Was the response time in ICE:

    `Unreasonable   1   2   3   4   5   6   7   Reasonable`

14. Screens presented during consultation with ICE were:

    `Simple   1   2   3   4   5   6   7   Too complicated`

15. Were the recommendations provided by ICE understandable?

    `Yes   1   2   3   4   5   6   7   No`

16. In your opinion were the recommendations:

    `Simple   1   2   3   4   5   6   7   Complicated`

17. Did you perceive the duration of consultation with the ICE system to be:

    `Sufficient   1   2   3   4   5   6   7   Insufficient`

18. Did you feel confident about the solutions offered by the ICE system?

    `Yes   1   2   3   4   5   6   7   No`

19. How confident were you about the solutions offered by the ICE system?

    `Very confident   1   2   3   4   5   6   7   Not confident`

20. The "help" and other instructions provided by ICE were:

    `Sufficient   1   2   3   4   5   6   7   Insufficient`

21. Did you feel the consultation session with ICE was:

    `Flexible   1   2   3   4   5   6   7   Inflexible`

22. Were the recommendations made:

    `Precise   1   2   3   4   5   6   7   Imprecise`

23. In your opinion, the ICE system was:

    `Easy to use   1   2   3   4   5   6   7   Difficult to use`

24. The questions asked by the ICE system were:

Meaningful	1	2	3	4	5	6	7	Meaningless

25. The wording of the questions in ICE was:

Complicated	1	2	3	4	5	6	7	Simple

26. The software recommended by ICE was:

Satisfactory	1	2	3	4	5	6	7	Unsatisfactory

27. If faced with a task requiring software support in the future would you consult with ICE to aid you in your decision?

Yes	1	2	3	4	5	6	7	No

28. If given the choice between a consultant and the ICE system, which would you prefer?

Prefer ICE	1	2	3	4	5	6	7	Prefer consultant

Please use the back of this page to give any suggestions regarding the task, or your experience with the ICE system. Any additional comments would be most helpful.

Thank you very much for your time and cooperation

Appendix H

Case Scenarios

Joe has a PC, using it both as a stand-alone PC and as a connection to the host. Joe is proficient in using the computer for text preparation, simple programming, and already uses several software packages in his work.

Current need:

Joe has less than 1000 inventory records for which he wants to do the following:

- perform general queries and data retrieval
- be able to save the queries for re-use
- perform calculations on the data
- view and edit the stored data
- prepare simple reports

Joe realizes that his data is subject to frequent changes. He is willing to spend anywhere from 5-20 hours learning a software package to accomplish this task, and wants to work in the PC environment.

Jill has a PC, and uses it both as a stand-alone PC and as a connection to the host. Jill is proficient in using the computer for text preparation, simple programming, uses several software packages, and is familiar with the VM environment.

Current need:

Jill has 2500 inventory records for which she wants to do the following:

- perform general queries and data retrieval
- be able to save the queries for re-use
- perform calculations on the data
- view and edit the stored data
- prepare simple reports

Jill realizes that her data is subject to frequent changes. She is willing to spend anywhere from 5-20 hours learning a software to accomplish this task, and wants to work in the VM environment.

Mary has a PC, and uses it both as a stand-alone PC and as a connection to the host. Mary is proficient in using the computer for text preparation, simple programming, uses several software packages, and is familiar with the VM environment.

Current need:

Mary has 7500 inventory records for which she wants to do the following:

- perform general queries and data retrieval
- be able to save the queries for re-use
- perform calculations on the data
- view and edit the stored data
- prepare simple reports

Mary realizes that her data is subject to frequent changes. She is willing to spend as much time as needed learning a software package to accomplish this task, and wants to work in the VM environment.

Lewis uses a PC exclusively in his work. He rates himself as a "proficient" user, and finds himself using the computer with increasing frequency to accomplish his tasks. He has used data base programs, word processing packages and simple graphics programs; he knows Basic and Cobol programming languages.

Now Lewis wants to find a package to assist in a Business Planning activity, and thinks a spreadsheet would help. He needs some quick answers to hypothetical questions about financial issues: to perform "what if" analyses. The analyses are complicated enough that he wants to be able to write his own subroutines.

When the analyses are finished, he needs to prepare a customized report for his manager. The report must include charts of the data.

Lewis is willing to spend up to 20 hours learning the new package, and must work in the PC environment.

Ann uses a PC exclusively in her work. She rates her computer skills as "average"– she is familiar with the commonly used functions. In the past, she has used word processing packages and simple graphics programs. She has had one course in the Basic language.

Now Ann wants to find a package to assist in a Business Planning activity, and thinks a spreadsheet would help. She needs some quick answers to hypothetical questions about financial issues: to perform "what-if" analyses.

When the analyses are finished, she needs to prepare a customized report for her manager.

Ann is not willing to spend more than a day learning the new package, and must work in the PC environment.

Jeff has a stand-alone PC which he uses on a regular basis to create simple programs for his department's data analysis needs. He is unfamiliar with a number of software tools that could aid him with the increasing demand on his time to do data analysis. Jeff wants to acquire software to handle the common recurring problems that he has to deal with.

The majority of Jeff's tasks involve the following:

- Simple statistical analysis such as the Means, Standard Deviations, t-tests, and ANOVA.
- Integration of the results into a report.

The data to be analyzed is usually delivered to him on a PC-disk. By choice, he likes to do all his tests online. Based on his workload at present, Jeff can spend 2 or 3 days (about 20 hours) learning any new software.

Cynthia has been using her PC since she received it as a "dumb" terminal. Over the last few months, with an increase in the data analysis aspect of her work, she is starting to feel irritated by the occasional downtime of the mainframe. In the past, Cynthia has used the mainframe on a very regular basis, and like she is very proficient with the programming environment and the software available. Her current need is:

- Perform statistical analysis coupled with business financial planning and forecasting.
- The statistical analysis is of a fairly complex nature including MANOVA, multiple regression and the like.
- Incorporate the analysis into business charts and user defined reports.
- Remain online.

Cynthia wants to find out about software which is available to her in either the VM or the PC environment.

Larry has been a VM hacker since he joined the firm. He has in the past been involved primarily at the assembly/machine language level. Besides being a whiz programmer, he is also quite a statistician, and he has now been asked to analyze an extremely large data set.

In his latest assignment he was asked to:

- Explore relations in the collected data
- Perform forecasting based on the relations
- Use scientific graphs/business charts as appropriate to present the data.

Having been a VM person, Larry wants to get software that can meet his needs in this environment. He is ready to spend as much time as necessary to learn the appropriate software.

Jacob has recently joined the firm with a background in business statistics. Jacob is a novice at using computers. Upon joining the firm, he has decided to acquaint himself with software needed to perform his job efficiently.

His current assignment is:

- Statistical analysis and forecasting
- Preparation of various business graphs and charts
- Report generation in pre-specified format and including the statistical analysis in them
- Performing his analysis online

His computing environment at present consists of a stand-alone PC, and he has substantial time to acquaint himself with any new software acquired.

Irene began using the computer last year to prepare memos and letters with a simple word processing package. She has a PC on her desk, with a connection to the host, but she mainly uses the host to receive mail messages.

Now she has been asked to prepare presentation materials in the form of overhead transparencies. The material includes both alphanumeric text as well as graphics, so she requires a package with the following capabilities:

- different fonts
- varying character sizes
- library of pre-stored symbols

Irene will be entering the data via the keyboard. She is willing to spend 5 - 20 hours learning the package, and must work in the PC environment.

Barbara began using the computer last year to prepare memos and letters with a simple word processing package. She has a PC on her desk, with a connection to the host, but she mainly uses the host to receive mail messages. She is eager to expand her skills in taking advantage of what the host has to offer.

Now she has been asked to prepare presentation materials in the form of overhead transparencies. The material includes both alphanumeric text as well as graphics, so she requires a package with the following capabilities:

- different fonts
- varying character sizes
- library of pre-stored symbols
- color

Barbara will be entering the data via the keyboard. She is willing to spend up to a week learning the package, as she has been told that she will in the future be preparing many of these presentations for her department. She would like to work in the VM environment.

Sarah began using the computer last year to prepare memos and letters with a simple word processing package. She has a PC on her desk, with a connection to the host, but she mainly uses the host to receive mail messages.

Now she has been asked to prepare presentation materials: paper copy. The data is all alphanumeric text. She requires a package with the following capabilities:

- different fonts
- varying character sizes
- color is not required

Sarah will be entering the data via the keyboard. She is not willing to spend much time learning the package, a maximum of 4 - 5 hours, and prefers a menu-driven, easy-to-use approach. She must work in the PC environment.

Jack began using the computer two years ago to prepare documents with PROFS. He has a PC on his desk, with a connection to the host. He knows no programming languages, but has used many of the facilities of the PROFS environment and is eager to expand his knowledge of the VM environment.

Now he has been asked to prepare presentation materials in the form of overhead transparencies as well as paper copies. The material is alphanumeric text, and so he requires the following capabilities:

- different fonts
- varying character size
- color

Jack will be entering the data via the keyboard. He is willing to spend around 20 hours learning the package, and wishes to work in the VM environment.

Sue began using the computer two years ago to prepare documents with PROFS. She has a PC on her desk, with a connection to the host. She knows no programming languages, but has used many of the facilities of the PROFS environment and is eager to expand her knowledge of the VM environment.

Now she has been asked to prepare presentation materials in the form of overhead transparencies as well as paper copies. The material includes both alphanumeric text and graphics, so she requires the following capabilities:

- different fonts
- varying character sizes
- color
- library of pre-defined graphic symbols

Sue will be entering the data via the keyboard. She is willing to spend around 20 hours learning the package, and wishes to work in the VM environment.

Milton has used the VM environment for programming purposes. Now he has been assigned to a project where he will have to prepare many documents that include column processing.

He also desires the capability of a spell-checker. He wants to be able to merge files, and needs to embed graphics in the text.

Milton is willing to spend more than 20 hours learning the package, and must work in the VM environment.

Chad has a PC, which he has used mostly for programming purposes. Now he has been assigned to a project where he will have to prepare documents that include column processing.

He also desires the capability of a spell-checker. He wants to be able to merge files, and needs to embed graphics in the text.

Chad is willing to spend more than 20 hours learning the package, and must work in the PC environment.

Elizabeth has used PROFS for document preparation, but now must migrate to a PC environment. She will continue to do much word processing and editing, and uses the following features regularly:

- production of table of contents
- footnotes
- spelling checker
- merge files

The package must be for the PC environment.

Ernie wants to do simple memo and document preparation on his PC. He is not very familiar with the computer, and does not want to spend much time learning a word processing package–less than 5 hours.

He wants "the simplest word processing program" available, but it must offer a spelling checker. It must be PC-based.

Olivia is a manager in charge of a project in which 5 other people are working under her. The project has about 30 tasks, and involves managing 10 other resources.

Olivia needs the following capabilities from a PC program:

- resource leveling
- slack time analysis
- critical path analysis
- project tracking

Olivia wants the facility for defining the workdays on a user-specific calendar. She must provide simple progress reports to her third-level manager. These reports must include PERT and Gantt charts.

Chris is a manager in charge of a project in which 10 other people are working under him. The project has about 60 tasks, and involves managing 50 other resources.

He needs the following capabilities from a software package:

- resource leveling
- slack time analysis
- critical path analysis
- project tracking
- risk analysis

Chris wants the facility for defining the workdays on a user-specific calendar. He must provide simple progress reports to his manager. These reports must include PERT and Gantt charts. Chris is working in a VM environment.

Wayne has a been using a PC for the past two years. His PC use has been restricted to preparing documents using Word Perfect. His job however requires him to do extensive data gathering, and data analysis. Wayne would like to use his PC to help him with data analysis and data retrieval. He does not however want to spend too much time learning several different packages.

Current need:

- word processing with a spell checker
- ability to perform simple statistical procedures
- view and edit the data
- include data into reports being prepared

Wayne enjoys working on the PC, and is ready to spend a reasonable amount of time learning any new package.

Appendix I

```
                 University of Arizona
     College of Business and Public Administration

     Center for the Management of Information (CMI)
        IIIIIIIII        CCCCCCCC       EEEEEEEEE
            I           CC              E
            I           CC              EEEEE
            I           CC              E
        IIIIIIIII        CCCCCCCC       EEEEEEEEE

        Information     Center          Expert

                       M A I N   M E N U
                       =================

                  1.  Start a consultation
                  2.  ICE description
                  3.  Help
                  4.  Exit

             Choose a number and hit ENTER
```

```
          Center for the Management of Information (CMI)
                                                   ________________
 1 Please enter your Faculty/Staff/Student ID     |  PF1  Help     |
   number.                                        |  PF2  Review   |
                                                  |  PF3  End      |
                                                  |  PF4  What     |
 484060___________________________________        |  PF5  Undo     |
                                                  |  PF6  Unknown  |
                                                  |  PF7  Up       |
                                                  |  PF8  Down     |
                                                  |  PF9  Tab      |
                                                  |  PF10 How      |
           Press RETURN to continue               |  PF11 Why      |
                                                  |  PF12 Command  |
                                                  |________________|
```

Center for the Management of Information (CMI)

PLEASE NOTE

The following questions are being asked
because this is your first interaction
with the system. These questions are for
understanding your work environment.

Note: You need to answer these questions
only this first time.

Press RETURN to continue.

Center for the Management of Information (CMI)

User Profile Questions - Screen # 1

2 Last name

Vinze____________________________

3 First name

Ajay_____________________________

4 Department

_ Accounting
_ Finance
_ Economics
_ Management and Policy
x M.I.S.
_ Marketing
_ Other

5 Telephone number

621-2748____

Use the Arrow keys to move on the screen.
Press RETURN after answering all questions.

Center for the Management of Information (CMI)

User Profile Questions - Screen # 2

6 What hardware is available in your work environment? (The IC has graphics terminals available.)

_ Stand-alone PC
x PC used as host terminal
_ Host terminal

(Choose any number of the following:)

Use the Arrow keys to move on the screen.
Put an X next to your choices and hit RETURN.

Center for the Management of Information (CMI)

User Profile Questions - Screen # 3

7 Which of the following describe your most common uses for the computer?

_ Text preparation
_ Programming
x Using software packages

(Choose any number of the following:)

Use the Arrow keys to move on the screen.
Put an X next to your choices and hit RETURN.

Center for the Management of Information (CMI)

User Profile Questions - Screen # 4

(Choose one of the following:)

8 How would you rate your skills for working on the computer?
_ Unfamiliar
x Familiar
_ Proficient

9 How often do you use the computer to do your work?
_ Seldom
_ Regularly
x Frequently

Unfamiliar - Little or no knowledge of the computer environment.
Familiar - Knowledge of common functions.
Proficient - Adept at common as well as complex functions.

Use the Arrow keys to move on the screen.
Put an X next to your choices and hit RETURN.

Center for the Management of Information (CMI)

CURRENT PROFILE

Employee Name	Vinze Ajay
Department	M.I.S.
Phone number	621-2748
Hardware Available	PC used as host terminal
Computer Usage	Using software packages
Computer Skills	Familiar
Usage Frequency	Frequently

Press RETURN to continue

Center for the Management of Information (CMI)

PLEASE NOTE

Your current user profile has been loaded. The following questions will pertain to your current software needs.

Press RETURN to continue.

Center for the Management of Information (CMI)

10 Which of the following best describes your need for the current consultation? (term descriptions are at the bottom of the screen.)

(Choose one of the following:)

_ Data Management
_ Data Analysis
_ Graphics
x Document Preparation
_ Project Management
_ Utilities
_ Integrated Packages

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Undo
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Put an X next to your choice and hit RETURN

Down

TERM DESCRIPTIONS

Data Management -- data base, file management, queries, reports.

Data Analysis -- statistics, spreadsheets, number crunching or other mathematical manipulation, reports from data, graphical

Center for the Management of Information (CMI)

11 Choose the required facilities:

(Choose any number of the following:)
x Production of table of contents
_ Index
x Subscripts and superscripts
x Footnotes
_ Column-processing
x Spelling checker
_ Simple math symbols
x Automatic outlines

Put an X next to your choice and hit RETURN
Use PF6 to indicate if UNKNOWN.

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Undo
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Determining essential text processing facilities.

Center for the Management of Information (CMI)

12

Mark the capabilities that are REQUIRED:
(Choose any number of the following:)
x Merge files
_ User-defined keys and/or functions
x Choice of different font styles
_ Color highlighting
x Imbedding graphics in the text
_ None of the above

Put an X next to your choice and hit RETURN
Use PF6 to indicate if UNKNOWN.

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Undo
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Determining the key capabilites of the text processing software.

Center for the Management of Information (CMI)

13 Do you need to integrate text with graphics or spreadsheet data from the "Assistant Series" software?

(Choose one of the following:)
_ yes
x no

Put an X next to your choice and hit RETURN
Use PF6 to indicate if UNKNOWN.

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Undo
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Center for the Management of Information (CMI)

PLEASE NOTE

The next few questions are for understanding your data source and operating environment. These questions are needed before a tool recommendation can be made.

Press RETURN to continue.

```
          Center for the Management of Information (CMI)
14                                                         | PF1  Help    |
                                                           | PF2  Review  |
    Is your source data stored in the computer?            | PF3  End     |
(Choose one of the following:)                             | PF4  What    |
_ Yes                                                      | PF5  Undo    |
x No                                                       | PF6  Unknown |
                                                           | PF7  Up      |
                                                           | PF8  Down    |
                                                           | PF9  Tab     |
                                                           | PF10 How     |
                                                           | PF11 Why     |
   Put an X next to your choice and hit RETURN             | PF12 Command |
         Use PF6 to indicate if UNKNOWN.                   |______________|
```

```
          Center for the Management of Information (CMI)
15                                                         | PF1  Help    |
                                                           | PF2  Review  |
    How will you enter your data?                          | PF3  End     |
(Choose any number of the following:)                      | PF4  What    |
x Keyboard                                                 | PF5  Undo    |
x External disk                                            | PF6  Unknown |
_ Magnetic tape                                            | PF7  Up      |
_ Punched cards                                            | PF8  Down    |
                                                           | PF9  Tab     |
                                                           | PF10 How     |
                                                           | PF11 Why     |
   Put an X next to your choice and hit RETURN             | PF12 Command |
         Use PF6 to indicate if UNKNOWN.                   |______________|
```

Center for the Management of Information (CMI)

16

What is your mode of operation?
(Choose any number of the following:)
x Online
_ Batch

Put an X next to your choice and hit RETURN
Use PF6 to indicate if UNKNOWN.

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Undo
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Center for the Management of Information (CMI)

17

In which environment would you prefer to work?
(Choose any number of the following:)
_ VM
x PC/DOS

Put an X next to your choice and hit RETURN
Use PF6 to indicate if UNKNOWN.

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Undo
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Will try to determine software for the environment you choose. May also inform you of tools that would work in other hardware environments.

```
Number   Software Name          Confidence Level
------   -------------------    ----------------

1        DISPLAYWRITE 4          .84
2        DW ASSISTANT            .78
3        WRITING ASST.           .68
4        PERSONAL EDITOR         .65
5        PROFESSIONAL ED         .65



Confidence Level : 0.0 <----------------------------------> 1.0
                 No Match                            Highly recommended
For tools whose confidence level is less than 1.0, check IC consultants for
capabilities of the tool.

Enter the number of the software(1-5) or enter 0 to EXIT ==>
```

```
Number   Software Name          Confidence Level
------   -------------------    ----------------

1        DISPLAYWRITE 4          .84   <------ THIS IS YOUR CHOICE
2        DW ASSISTANT            .78
3        WRITING ASST.           .68
4        PERSONAL EDITOR         .65
5        PROFESSIONAL ED         .65



Confidence Level : 0.0 <----------------------------------> 1.0
                 No Match                            Highly recommended
For tools whose confidence level is less than 1.0, check IC consultants for
capabilities of the tool.

DISPLAYWRITE 4      has been chosen,
1. Detailed Descript., 2. Consultant's info. Enter your choice ==>
```

IBM DisplayWrite 4 is a full function word processor with support for many different printers. It can both accept ASCII, DIF(Lotus), SULK (Multiplan), PCS (Personal Decision Series) and dBase files. This tool has a built in spell checker as well as providing capability for automatic outlining and footnotes. Graphics capabilities include a cursor draw. Multiple user profiles can be created for text and workstations. DisplayWrite 4 provides 4 function math and technical writing support.

Hit the RETURN key to continue!

Please contact the following consultant(s) for further information:

Name : Kendall Cliff
Phone number : 621-2903

Name : Yi-Ching Liou
Phone number : 621-2903

Hit the RETURN key to continue!

LIST OF REFERENCES

1. Adrion, W.R., M.A. Branstad, and J.C. Cherniavsky, "Validation, Verification, and Testing of Computer Software," *Computing Surveys*, Volume 14, Number 2, June, 1982, pp. 159-192.

2. Alavi, Maryam, and Are R. Weiss, "Managing the Risks Associated with End-User Computing," *Journal of Management Information Systems*, Volume 2, Number 1, Winter 1985-86, pp. 5-20.

3. Alberdi, Marco De and James Harvey, "Decision Systems", in *Human Factors of Information Technology in the Office*, Bruce Christie Ed., New York: John Wiley and Sons, 1985.

4. Amoroso Donald L. and Paul H. Cheney, "Factors Affecting the Effectiveness of End-User Developed Applications," In Press, 1987.

5. Atre, Shaku, "The Information Center Holds", *PC World*, August, 1986, 157-163.

6. Bailey, J.E., and S.W. Pearson, "Development of a Tool for Measuring and Analyzing Computer User Satisfaction," *Management Science*, Volume 25, Number 5, May 1983, pp. 530-545.

7. Barr, A. and E.A. Feigenbaum, *The Handbook of Artificial Intelligence*, Volume I, HeurisTech Press, California, 1981.

8. Barstow, D.R., N. Aiello, R. Duda, L. Erman, C. Forgy, D. Gorlin, R. Greiner, D. Lenat, P. London, J. McDermott, H. Nii, P. Politakis, R. Reboh, S. Rosenschein, C. Scott, W. van Memme, and S. Weiss, "Expert Systems Tools," in *Building Expert Systems*, F. Hayes-Roth, D.A. Waterman, and D.B. Lenat (eds.), Addison-Wesley, Reding, Massachusetts, 1983, pp. 283-345.

9. Benjamin, R. I., "Information Technology in the 1990's: A Long Range Planning Scenario." *MIS Quarterly*, Volume 6, Number 2, June 1982, pp. 11-31.

10. Benson, David H., "A Field Study of End-User Computing," *MIS Quarterly*, Volume 7, Number 4, June 1982, pp. 11-31.

11. Berry, Dianne, C. and Donald E. Broadbent, "Expert Systems and the Man-Machine Interface. Part Two: The User Interface", *Expert Systems*, Vol. 4, No. 1, February 1987.

12. Bliss, J., P. Feld, and R.E. Hayes, "Decision Aid Measurement and Evaluation (DAME)," in *Proceedings of the 1986 IEEE International Conference on Systems, Man, and Cybernetics*, IEEE Piscataway, NJ, 1986, pp. 126-130.

13. Bohl, Don Lee, editor, *The AMA Report on Information Centers*, AMA Membership Publications Division, NY: American Management Association, 1986.

14. Bobrow, Daniel G., Sanjay Mittal, and Mark J. Stefik, "Expert Systems: Perils and Promise," *Communications of the ACM*, Vol. 29, No. 9, September 1986.

15. Boehm, B., J.R. Brown, H. Kaspar, M. Lipow, G.J. MacLeod and M.J. Merrit, *Characteristics of Software Quality*, North Holland, 1978.

16. Bramer, M.A., "A survey and critical review of expert systems research," in *Introductory Readings in Expert Systems*, Donald Michie (ed.), Godon and Breach Science Publishers, New York, 1984.

17. Brancheau, James C., Robert L. Leitheiser and James C. Wetherbe, "Allocating Responsbility for End-User Computing Support: Who's Responsible in the 'Shared' Environment," *Working Paper Series*, MISRC-WP-87-08, University of Minnesota, 1987.

18. Brancheau, James C., and James C. Wetherbe, "Key Issues in Information Systems Management," *Management Information Systems Quarterly*, Volume 11, Number 1, March 1987.

19. Brancheau, James C., Douglas R. Vogel, and James C. Wetherbe, "An Investigation of the Information Center from the User's Perspective," *DATABASE*, Volume 17, Number 1, Fall 1985, pp. 4-17.

20. Brown, J.S., "The low road, the middle road, and the high road," *The AI Business*, P.H. Winston and K. Prendergast, Eds., Cambridge Mass: MIT Press, 1984.

21. Buchanan, B.G. and E.H. Shortliffe, *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, Addison-Wesley, 1984.

22. Buchanan, B.G., D. Barstwo, R. Bechtel, J. Bennet, W. Clancey, C. Kulikowski, T. Mitchell, and D.A. Waterman, "Constructing an Expert System," in *Building Expert Systems*, F. Hayes-Roth, D.A. Waterman and D.B. Lenat (eds.). Addison-Wesley, 1983.

23. Campbell, D.T. and J.C. Stanley, *Experimental and Quasi-Experimental Designs for Research*, Rand McNally, Chicago, 1966.

24. Card, S., T. Moran, and A. Newell, *The Psychology of Human-Computer Interaction*, Lawence Erlbaum Associates, Hillsdale, NJ, 1983.

25. Cheney, Paul H., Robert I. Mann, and Donald L. Amoroso, "Organizational Factors Affecting the Sucess of End-User Computing," *Journal of Management Information Systems*, Volume 3, Number 1, Summer 1986, pp. 66-80.

26. Clancey, W.J., "The Advantages of Abstract Control Knowledge in Expert System Design", *Proceedings AAAI-1983*, pp. 74-78, 1983.

27. *Codasyl End-User Facilities Committee Status Report.* North Holland Publishing Company. Information and Management Two, 1979, pp 137-163.

28. Cole, I., M. Lansdale, and B. Christie, "Dialogue Design Guidelines," in *Human Factors of Information Technology in the Office*, Bruce Christie Ed., New York: John Wiley and Sons, 1985.

29. Damodaran, L., "User Support," in *Computer Information Systems and Management*, N.B. Anderson, K.D. Eason, and D. Robey (eds), MIT Press, 1984.

30. Danziger, James N. and Kenneth L. Kraemer, *People and Computers - The Impacts of Computing on End Users in Organizations*, Columbia University Press, NY 1986.

31. Date, C.J., *An Introduction to Database Systems*, Vol. 1, Addison-Wesley, Menlo Park, 1986.

32. Davis, R., "Amplifying Expertise with Expert Systems," *The AI Business*, Winston, Patrick H. and Karen A. Prendergast (eds.), Cambridge, Mass: The MIT Press, 1984.

33. Davis, R., "Interactive Transfer of Expertise," *Artificial Intelligence*, Volume 12, 1979, pp. 121-157.

34. Davis, R., "Knowledge-Based Systems," *Science*, 231, 1986, pp. 957-963.

35. Davis and Wetherbe [1979]

36. Debons, A., W. Ramage, and J. Orien, "Effectiveness Model of Productivity," in *Research on Productivity Measurement Systems for Administrative Services: Computing and Information Services*, L.F. Hanes and C.H. Kriebel (eds.) Volume 2, NSF Grant APR-20546, July 1978.

37. DeSanctis, G. and Courtney, J.F., "Toward Friendly User MIS Implementation," *Communications of the ACM*, Volume 26, Number 10, October, 1983, pp. 732-738.

38. Dickson, Gary W., Robert L. Leitheiser, Mal Nechis, and James C. Wetherbe, "Key Information System Issues for the 1980's," *Management Information Systems Quarterly*, Volume 8, Number 3, September 1985, pp. 135-148.

39. Dyer, J.S. and R.K. Sarin, "Measurable Multiattribute Value Functions," *Operations Research*, Volume 27, Number 4, July-August, 1979.

40. Eason, K.D., "Towards the experimental study of usability," *Behaviour and Information Technology*, Volume 3, Number 2, 1984, pp. 133-143.

41. Ein-Dor, P. and Segev, E. "Organizational Context and the Success of Management Information Systems," *Management Science*, Volume 24, Number 10, June 1978, pp. 1064-1077.

42. Ericsson, K.A., and H.A. Simon, *Protocol Analysis Verbal Reports and Data*, MIT Press, Cambridge, MA, 1984.

43. Erman, L.E., Scott, A.C., and London, P.E., "Separating and integrating control in a rule-based tool," *Proceedings of the IEEE Workshop on Principles of Knowledge-Based Systems*, Denver, 1984, pp. 37-43.

44. Feigenbaum, E. A. *Knowledge Engineering for the 80's.*, Computer Science Department, Stanford University, (1982).

45. Feigenbaum, E.A., "The Simulation of Verbal Learning Behavior," in *Computers and Thought*, E.A. Feigenbaum and J. Feldman (eds.), McGraw-Hill, New York, NY, 1963.

46. Forsyth, Richard, "The architecture of expert systems", *Expert Systems Principles and Case Studies*, Richard Forsyth (ed), NY: Chapman and Hall, 1984.

47. Gallagher, C.A. "Perceptions of the value of a management information system," *Academy of Management Journal*, Volume 17, Number 1, 1974.

48. Gaschnig, John, Philip Klahr, Harry Pople, Edward Shortliffe, and Allan Terry, "Evaluation of Expert Systems: Issues and Case Studies," in *Building Expert Systems*, Hayes-Roth, F., Donald A. Waterman, and Douglas B. Lenat (eds.), Addison-Wesley, Reading, MA., 1983, pp. 241-280.

49. Ginzberg, M.J. "Finding an Adequate Measure of OR/MS Effectiveness," *Interfaces*, Volume 8, Number 4, August 1978, pp. 59-62.

50. Green, C.J.R. and Keyes, M.M., "Verification and Validation of Expert Systems," in *IEEE Western Conference on Expert Systems*, June 2-4, Anaheim, CA, 1987, pp. 38-43.

51. Grover, M., "A Pragmatic Knowledge Acquisition Methodology," *Conference Proceedings of the 8th International Conference on AI*, 1983.

52. Guimaraes, Tor, "The Evolution of the Information Center," *Datamation*, July 15, 1984, 127-130.

53. Hamilton, S. and N.L. Chervany, "Evaluating Information System Effectiveness – part I," *MIS Quaterly*, September, 1981.

54. Hammond, L.W., "Management Considerations for an Information Center," *IBM Systems Journal*, Volume 21, Number 2, April 1982, pp. 131-161.

55. Harmon, Paul and David King, *Expert Systems - Artificial Intelligence in Business*, John Wiley and Sons, NY, 1985.

56. Hart, A., *Knowledge Acquisition for Expert Systems*, McGraw Hill, New York, 1986.

57. Hayes-Roth, F., "Rule-Based Systems," *Communications of the ACM*, Volume 28, Number 9, September 1985, pp. 921-932.

58. Hayes-Roth, F., "Knowledge-Based Expert Systems," *IEEE Computer*, October, 1984, pp. 263-273.

59. Hayes-Roth, F., Donald A. Waterman, and Douglas B. Lenat, "An Overview of Expert Systems" in *Building Expert Systems*, Hayes-Roth, F., Donald A. Waterman, and Douglas B. Lenat (eds.), Addison-Wesley, Reading, MA., 1983, pp. 3-30.

60. Heltne, Mari M., Ajay S. Vinze, Benn R. Konsynski and J.F. Nunamaker, Jr. "A Consultation System for Information Center Resource Allocation," *Proceedings of the 1987 SIGBDP-SIGCPR Conference,* Elias M. Awad, ed., 1987, pp. 20-44.

61. Henderson, John C. and Michael E. Treacy, "Managing End-User Computing for Competitive Advantage," *Sloan Management Review*, Winter 1986, pp. 3-14.

62. Hickam, D.H. et al., "The Treatment Advice of Computer Based Cancer Chemotherapy Protocol Advisor," *Annals of Internal Medicine*, Volume 103, Number 6, Part 1, 1985, pp. 928-936.

63. Hirsch, P., W. Katke, M. Meier, S. Snyder, R. Stillman, "Interfaces for knowledge-base builders' control knowledge and application-specific procedures" *IBM Journal of Research and Development*, Vol 30, No. 1, January 1986.

64. Hopwood, A., *Accounting and Human Behavior*, Englewood Cliffs, N.J.: Prentice Hall, 1974.

65. IBM, *Expert System Development Environment/VM*, Reference Manual, IBM, Program Number 5798-RWQ, 1986.

66. Ives, B., M.H. Olson, and J.J. Baroudi, "The Measurement of User Information Satisfaction," *Communications of the ACM*, Volume 26, Number 10, October 1983, pp. 785-793.

67. Keen, P.G.W. and M.S. Scott Morton, *Decision Support Systems: An Organizational Perspective*, London: Addison-Wesley, 1978.

68. Keeney, R.L. and H. Raiffa, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, John Wiley and Sons, 1976.

69. Kerlinger, F.N., *Foundations of Behavioral Reserach*, Holt, Rinehart and Winston, New York, 1986.

70. Korfhage, Robert R., "Intelligent Information Retrieval: Issues in User Modelling," *IEEE - Expert Systems in Government Symposium*, Kamal N. Karna (ed.), 1985, pp. 474-482.

71. Krcmar, Helmut A. O., "Enterprise-Wide Information Management: Expert Systems for Information Management." IBM Los Angeles Scientific Center, *Report Number G320-2767*, July, 1985.

72. Kulikowski, C.A. and S.H. Weiss, "Representation of Expert Knowledge for Consultation: The Casnet and Expert Projects," in Artificial Intelligence in Medicine, P. Szolovits (ed.) Westview Press, Boulder, CO, 1982, pp. 21-56.

73. Larcker, D.F. and V.P. Lessig "Perceived usefulness of information: A psychometric examination," *Decision Science*, Volume 11, Number 1, 1980, pp 121-134.

74. Lane, N.E., "Global Issues in Evaluation of Expert Systems," in *Proceedings of the 1986 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 121-125, IEEE, Piscataway, NJ, 1986.

75. Leitheiser, R.L., "Computer Support for Knowledge Workers: A Review of Laboratory Experiments," *DATABASE*, Spring 1986, pp. 17-45.

76. Leitheiser, Robert L., and James C. Wetherbe, "Service Support Levels: An Organized Approach to End-User Computing," *MIS Quarterly*, Volume 10, Number 4, December 1986, pp. 337-349.

77. Liebowitz, Jay, "Useful Approach for Evaluating Expert Systems," *Expert Systems*, Volume 3, Number 2, April 1986, pp. 86-96.

78. Little, John D. C., "Research Opportunities in the Decision and Management Sciences," *Management Science*, Volume 32, Number 1, January 1986, pp. 1-13.

79. Luconi, Fred L., Thomas W. Malone and Michael S. Scott Morton, "Expert Systems and Expert Support Systems: The Next Challenge for Management", *Sloan Management Review*, Fall 1986.

80. Martin, J., *Application Development Without Programmers*, Prentice Hall, Englewood Cliffs, NJ, 1982, 102-106.

81. Mavor, A.S. and J.S. Kidd, "Standards for Expert Systems," *Technical Report for Defense Advanced Research Projects Agency*, Essex Corporation, Alexandria, VA, June, 1986.

82. Moran, T., "An Applied Psychology of the User," *Computing Surveys*, Volume 13, Number 1, March 1981, pp. 1-11.

83. McDermott, J., "R1 Revisited: Four Years in the Trenches," *The AI Magazine*, Summer 1985, pp. 32-36.

84. Mclean, Ephriam R., "End users as application developers," *Proceedings of Guide/Share Application Development Symposium*, October 1974

85. McLean, Ephriam R., "End Users as Application Developers," *MIS Quarterly*, Volume 3, Number 4, December 1979, pp. 37-46.

86. Michaelsen, R. and D. Michie, "Expert Systems in Business," *Datamation*, Volume 29, November, 1983.

87. Minsky, M.L. and Papert, S., *Perceptrons*, The MIT Press, Cambridge, MA, 1969.

88. Minsky, M.L., "The Problems and Promise", in *The AI Business*, Winston, Patrick H. and Karen A. Prendergast (eds.), Cambridge, Mass: The MIT Press, 1984.

89. Myers, G.J., *The Art of Software Testing*, John Wiley and Sons, New York, NY, 1979.

90. Neumann, S. and E. Segev, "Evaluate Your Information System," *Journal of Systems Management*, Volume 31, March 1980.

91. Nunnally, J. *Psychometric Theory*, McGraw-Hill Book Co., New York, 1967.

92. O'Keefe, R.M., O. Balci, and E.P. Smith, "Validating Expert System Performance," *IEEE Expert*, Winter 1987, pp. 81-90.

93. Pople, H.E., "Knowledge-Based Expert Systems: The Buy or Build Decision," in *Artificial Intelligence Applications for Business*, Walter Reitman (ed.), Ablex Publishing, New Jersey, 1984.

94. Powers, R.F. and G.W. Dickson, "MIS Project Management: Myths, Opinions, and Reality," *California Management Review*, Volume 15, Number 3, Spring 1973, pp. 147-156.

95. Prerau, D.S. "Knowledge Acquisition in the Development of a Large Expert System," *AI Magazine*, Summer 1987, pp. 43-51.

96. Rauch-Hindin, Wendy B., *Artificial Intelligence in Business, Science, and Industry*, Volume I, Prentice-Hall, NJ, 1986.

97. Rockart, John F. and Lauren S. Flannery, "The Management of End-User Computing," *Communications of the ACM*, Volume 26, Number 10, October 1983, pp. 776-784.

98. Rodgers, E.M., *Diffusion of Innovations*, The Free Press, New York, NY, 1983.

99. Royksund, Conrad, "Critical Sucess Factors in Information Centers," *CMI Working Paper Series*, MIS department, University of Arizona, 1987.

100. Scambos, E.T., "A Scenario-Based Test Tool for Examining Expert Systems," in *Proceedings of the 1986 IEEE International Conference on Systems, Man, and Cybernetics*, IEEE Piscataway, NJ, 1986, pp. 131-135.

101. Schriven, M., "The Methodology of Evaluation: Formative and Summative Evaluation," *Evaluating Action Programs*, C.H. Weiss (ed.), Allyn and Bacon, Boston, 1972.

102. Scott-Morton, Michael S., "The State of the Art of Research," in *The Information Systems Research Challenge*, F. Warren McFarlan (ed.), Harvard Business School Press, Boston, 1984, pp. 13-41.

103. Simon, H.A. *The Science of Management Decision.* New York: Harper and Row, 1960.

104. Srinivasan, A. "Alternative Measures of System Effectiveness: Associations and Implications," *Management Information Systems Quarterly*, Volume 9, Number 3, September 1985, pp. 243-253

105. Sviokla, J.J., "Business Implications of Knowledge-Based Systems," Part I *DATABASE*, Summer 1986a, pp. 5-19.

106. Sviokla, J.J., "Business Implications of Knowledge-Based Systems," Part II *DATABASE*, Fall 1986b, pp. 5-16.

107. Thompson, B.A. and Thompson, W.A., "Inside an Expert System," *BYTE*, April, 1985, pp. 315-330.

108. Thomson, J.D. and Tuder, A., "Strategies, structures and processes of organisational decision", In J.D. Thompson (ed.) *Comparitive Studies in Administration*, University of Pittsburg Press, 1959.

109. Treacy, Michael E., "Future Directions in DSS Technology", *Center for Information Systems Research, Working Paper No. 123*, Sloan School of Management, MIT, Cambridge, MA, January 1985.

110. Treacy, M. "An Empirical Examination of a Causal Model of User Information Satisfaction," *Proceedings of the 6th International Conference on Information Systems*, Indianapolis, December 1985.

111. Turban, E., *Decision Support and Expert Systems*, MacMillan Publishing Company, New York, 1988.

112. Turing, A. M., "Computing Machinery and Intelligence," *Mind*, October 1950, reprinted in *Creative Computing*, Volume 6, Number 1, January 1980, pp. 44-53.

113. Vinze, Ajay S., Mari M. Heltne, Minder Chen, J.F. Nunamaker, Jr. and Benn R. Konsynski, "A Knowledge-Based Approach for Resource Management," *Proceedings of the 1987 International Conference on Information Systems*, Pittsburgh, 1987.

114. Waltz, David L., "Artificial Intelligence", *Scientific American*, 19??

115. Waldrop, M. Mitchell, "The Necessity of Knowledge", *Science*, Vol 223, March 23, 1984.

116. Waterman, Donald A., *A Guide to Expert Systems*, Addison-Wesley, Reading, MA, 1986.

117. Wetherbe, James C. and Leitheiser, Robert L., "A Survey of Information Centers: Services, Decision, Problems, and Successes," *Journal of Information Systems Management*, Volume 2, Number 3, Summer 1985.

118. Wiederhold, Gio, "Knowledge and Database Management", *IEEE Computer*, January 1984, pp. 63-73.

119. Wiener, N., *Cybernetics*, The MIT Press, Cambridge, MA, 1948.

120. Winston, P. H. and K. A. Prendergast, (eds.), *The AI Business: The Commercial Uses of Artificial Intelligence*, The MIT Press, Cambridge, Massachusetts, 1984.

121. Winston, Patrick Henry, *Artificial Intelligence*, Readings, Massachusetts: Addison-Wesley Publishing Company, 1984.

122. Yu, V.L., B.G. Buchanan, E.H. Shortliffe, S.M. Wraith, R. Davis, A.C. Scott, and S.N. Cohen, "Evaluating the Performance of a Computer-Based Consultant," *Computer Programs in Biomedicine*, Volume 9, 1979, pp. 95-102.

123. Yu, V.L., L.M. Fagan, S.W. Bennett, W.J. Clancey, A.C. Scott, J.F. Hannigan, B.G. Buchanan, and S.N. Cohen, "An Evaluation of MYCIN's Advice," in *Building Expert Systems*, Buchanan, B.G. and E.H. Shortliffe (eds.), pp. 589-596, Addison-Wesley, Reading, MA, 1984.

124. Zmud, Robert W., "Push-Pull Theory and Process Innovation,", *Management Science*, Volume 30, Number 6, June 1984, pp. 727-738.

125. Zobaidie, A. Al, and J.B. Grimson, "Expert Systems and Database Systems: how can they serve each other?" *Expert Systems*, Volume 4, Number 1, February 1987.