Order Number 8816314

Knowledge-based support for management of end user computing resources: Issues in knowledge elicitation and flexible design

Heltne, Mari Montri, 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 _____

UMI

KNOWLEDGE-BASED SUPPORT FOR MANAGEMENT OF END USER COMPUTING RESOURCES: ISSUES IN KNOWLEDGE ELICITATION AND FLEXIBLE DESIGN

by
Mari Montri Heltne

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

1988

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Final Examination Committee, we certify that we have read the dissertation prepared by Mari Montri Heltne

entitled KNOWLEDGE-BASED SUPPORT FOR MANAGEMENT OF END USER COMPUTING RESOURCES: ISSUES IN KNOWLEDGE ELICITATION AND FLEXIBLE DESIGN

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

J.F. Nunamaker, Jr.	March 28, 1988
	Date
E. Sue Weber	March 28, 1988
	Date
Joey F. George	March 28, 1988
	Date
George Summers	March 28, 1988
	Date
Dave Pingry	March 28, 1988
	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 J.F. Nunamaker, Jr. — March 31, 1988
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: Mari Montis Helton

ACKNOWLEDGEMENT

When we came nearest to being better than we are, And almost what we wished.

My mother, Iva! I think of her first. I wish she were here, and know that she is, as I am half of what she was. Else I would not have worked this hard; come this far.

Professional help has been superb: Jay Nunamaker, George Summers, Sue Weber, Joey George and Dave Pingry–my committee! Special appreciation to Jay, both for the opportunity and the responsibility to work on the Information Center Project with IBM, and to George for fine teaching and for work beyond his own retirement. Fred Wallace from IBM was not on this committee, but he could as well have been.

This work has made dear friends of research companions: Ajay Vinze, Minder Chen, and Irene Liou Chen, as it has of Amita Vinze, Kuni and Miki Higa, Liang Chee Wee and Yi-Ching Liou. And more: Nancy Nunamaker, Wayne Eirich, Bill Kelly. Irene Hop, Betty Albert and Joan Seidel deserve special awards.

Four people convinced me that the project was possible and sanity was preservable, perhaps because they had gone before, perhaps because they forgot how painful it was. To Susan for needing coffee more than I, Paul for being the VAXmail philosopher, and Lynda for absolutely everything! Judith Nye came to counsel in person because she could no longer afford the phone bills. Life is better because of all of you!

Let this be a first: I owe my landlords–Becky and Stan–for having provided us with a lovely house and cactus garden. They made "going home" when away from home a daily delight.

My children! For Michael Montri, my teenage son, and for Daniel, who is 8 but who will also someday be teenaged: now we can go trout fishing! And for Gail and Marty in Tucson–children who adopted me!

And to Conrad, who wrote those lines that hang on my wall before I ever knew him: they are mine, now. And it is time to go out to Saetre!

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Effective resource management requires tools and decision aides to help determine users' needs and appropriate assignment. The goal of this research was to design, implement, and test technological tools that, even in a dynamic environment, effectively support the matching of users and resources. The context of the investigation is the Information Center, the structure used to manage and control the computing resources demanded by end users. The major contributions of the research lie in two areas: (1) the development and use of a knowledge acquisition methodology called Resource Attribute Charts (RAC), which allow for the structured definition of the resources managed by the IC, and (2) the design, implementation, validation, and verification of the transportability of Information Center Expert, a system that supports the activities of the IC personnel. Prototyping, the system development methodology commonly used in software engineering, was used to design the general architecture of the knowledge acquisition tools, the knowledge maintenance tool, and the expert system itself. The knowledge acquisition tools, RAC, were used to build the knowledge base of ICE (Information Center Expert). ICE was installed at two corporate sites, its software recommendations were validated, and its transportability from one location to another was verified experimentally. The viability of a rule-based consultation system as a mechanism for bringing together knowledge about users, problems, and resources for the purpose of effective resource management was demonstrated.

Chapter 1

INTRODUCTION

The ability to identify resource needs and to acquire and allocate resources is essential to the effective management of any organization. Resource management consists of those activities involved in the optimal allocation and administration of an organization's human, financial, physical and informational resources needed to fulfill the organization's mission and achieve its goals and objectives [Bender 1983]. Effective resource management requires two things: (1) tools for determining the need for resources, and (2) decision aides for finding the best allocation of those resources [Rowe, Mason, and Dickel 1985].

In the past, resource allocation problems have been solved with mathematical modeling techniques. These algorithms, implemented on computers, have demanded that facts (variables) be known with certainty. In the real world, however, this requirement is not often met. The objective of this research is to determine whether technology and decision aides can be designed and implemented that can, even in an environment where facts are uncertain, effectively identify resource needs and facilitate an effective assignment of organizational resources to users with specific needs or problems.

The process of managing organizational resources is characterized by the model shown in Figure 1.1. It has four major components: (1) user - - the consumer of the

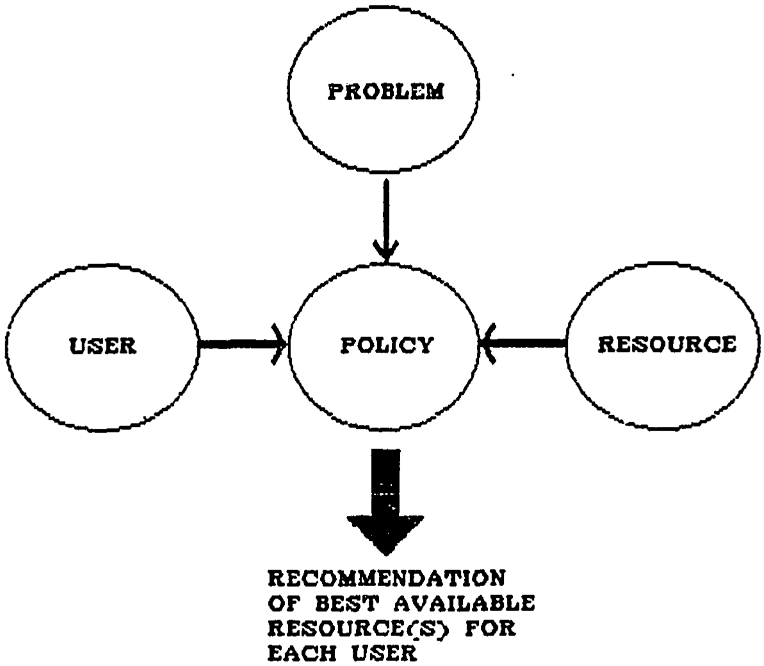

Figure 1.1: Model of the Environment for Resource Management

resource, (2) problem - - the context within which the resource is needed, (3) resources - - things (people, machines, facilities, land, information, and goodwill) that must be acquired, utilized, and allocated in order to carry out a plan or solve a problem [Rowe et al. 1985] and (4) organizational policies, priorities, and procedures that dictate the assignment or allocation of the resources to the users for particular problems. Users and resources must be identified and categorized, and the problems analyzed in a manner that is both appropriate for the user's situation and consistent with prevailing organizational policy.

The technology investigated in this study for assisting with identification of resources and matching them with the appropriate user and problem is the expert system. The context of the investigation is the Information Center, where users of organizational computing facilities come in search of hardware and software resources as solutions to their problems.

This chapter first gives the background of the investigation. Next, the goals and research questions are defined in the Statement of Problem. Last, an overview of the rest of the dissertation is presented.

1.1 Background of the Research

This section defines the boundaries of the research by giving background information on the types of resources relevant to this investigation. The focus is on reusable resources and the problems involved in their management. Specifically, the management of end user computing resources through the structure of the Information Center is examined. Within this structure, the opportunity exists for expert systems support.

1.1.1 Reusable Resources

The organizational resources relevant to this research are those that are reusable. "Reuse" of resources is the use of previously acquired resources, both objects and knowledge, in new or different situations. "Reusability" is defined as "a measure of the ease with which one can use those previous concepts and objects in new situations. Ideally, reuse is a matching process between new and old situations" [Prieto-Diaz and Freeman 1987]. Resources that are reusable, such as program code, system design specification modules, Request for Proposal (RFP) modules, or site licensed software packages, are normally developed or accumulated over long periods of time. Reusability - - the ease with which resources can be reused - - is widely believed to be a key to improving productivity and quality within organizations [Biggerstaff and Richter 1987]. While it is a strategy that holds great promise, it is one whose promise has been largely unfulfilled. Reusable resources are often under-used and neglected because of poor documentation and management. It takes longer to locate the wheel than to reinvent it. So, in many organizations, people duplicate previous activities because they do not know of the availability of existing resources, or cannot locate them. The cost of this duplication is substantial.

To reuse resources, one first must be able to find them. Reusing resources is not attractive unless the effort to reuse them is less than the effort to obtain or create new ones. The management of such resources is thus a classification and retrieval problem [Prieto-Diaz and Freeman 1987]. Reusable resources must be classified by relevant attributes so the resources can be located and matched to new situations when needed. Tools can be developed to help accomplish this process of classifying and retrieving. This research proposes, implements, and tests a method to accomplish this process.

1.1.2 Methodology for Acquiring Knowledge of Reusable Resources

One of the essential problems in managing reusable resources is locating and retrieving them. Questions such as the following need answers: What objects or fragments of resources are candidates for reuse? How should such objects be stored? Once stored, how are the resources located? Once located, how are resources assigned to appropriate users? [Prieto-Diaz and Freeman 1987]

To address these questions, a method was developed for helping managers identify attributes of resources so that the resources can more easily be searched for and retrieved. The method, which allows for the structured definition of those resources, is part of the knowledge acquisition stage of expert system building. Knowledge acquisition is commonly thought to be the least understood process in the paradigm and very few guidelines are available for facilitating the process [Clifford, Jurke, and Vassiliou 1982].

Although considerable research is currently being devoted to techniques for facilitating this process, more work needs to be done. The methodology proposed in Chapter 4 facilitates acquisition of the resource manager's knowledge about resources and the consultation process. Resource Attribute Charts (RAC) were developed and used to structure that knowledge for use in the knowledge base. The methodology also provides a software tool for building and maintaining the knowledge base. Thus, a framework is developed within which experts can be guided in communicating their expertise and knowledge about resources to knowledge engineers.

One of the instances in which this matching process is frequently done is in the management of computing resources, where reusable software products are matched with end users and their needs.

1.1.3 End User Computing Resources

There is a consensus in the literature that, before long, end user computing (EUC) will consume a majority of company computing resources [Canning 1984, 1987]. The term "end user computing" refers to the concept of all employees having access to computing power; EUC calls for employee participation in productivity improvement, where users themselves learn to satisfy some of their own computer requirements to ease the difficulties caused by expanding backlogs in the Information Systems (IS) department [Lodge 1983].

It is predicted that by 1990, EUC will represent as much as 75 percent of the total computing capacity of the typical American corporation [Bohl 1986]. IS managers seem to agree that the best general strategy for managing end users is to give them the resources, such as software tools and data, establish adequate standards and policies, and encourage good computing practices. The question is: How?

In many corporations, this strategy was given a location and a name: the Information Center. The mission of the Information Center (IC) is to "help users help themselves" by collecting and disseminating information about available computing resources: equipment, user developed systems, software packages, data, and training programs. Several types of service to the end user are offered, the most important of which are consultation, training, and technical expertise [Brancheau, Vogel, Wetherbe, 1985].

In the last decade, the information center concept was adopted repeatedly as the strategy for management of EUC. In the environment of the Information Center, the resource management model shown previously in Figure 1.1 takes on the specific characteristics of 1) users: end users, 2) problems: users' needs relating to computer hardware and software, 3) resources: software, hardware, and/or training resources supported by the Information Center, and 4) policies: the knowledge, procedures and priorities used by IC consultants who assess requirements and allocate resources within the center. End users come to the IC with specific problems needing solutions. These solutions are usually in the form of hardware, software, or training resources supported by the IC according to organizational policies. These policies are followed by the consultant when recommending solutions.

1.1.4 Proposal of Technology to Support End User Computing

Whatever the demonstrated value and continuing need for information centers as a methodology for managing end user computing resources, there are hard questions to be answered if they are to continue to be successful. A recent study [Brancheau, Vogel, and Wetherbe 1985] reported that end users expect to be even more dependent on the Information Center in the future than they are now, that they anticipate needing more support services and training, and that it will be more important than ever to remain "current" on new applications of technology. Thus, information centers are being subjected to increased user expectations, higher demand for integrated applications, and growing pressure to accomplish more with fewer resources. These pressures have placed information support services in a difficult situation.

Another major problem faced by information centers is the high rate of turnover in personnel. The unique combinations of skills found in IC personnel are difficult to find. IC consultants often become involved in important projects throughout the firm, and this visibility and opportunity often results in many job shifts [Konsynski 1984-85]. The potential turnover rate in the IC is high, causing loss of knowledge and experience in techniques for consulting with and training end users. Experienced IC consultants have developed implicit heuristics that enable them rapidly to assess an end user's needs and make appropriate recommendations. Such skill is very difficult to replace when those people leave the IC for another assignment. So the expertise must be captured and applied in other ways.

Harmon and King [1985, p. 198] suggest that knowledge systems are particularly helpful in places where "a few key individuals are in short supply . . .(where) they spend a substantial amount of time helping others." IC consultants, in short supply, spend a great deal of time helping end users to identify appropriate hardware and software resources to meet their technological needs. Therefore, help is needed to "help users help themselves."

Just as the IC provides support for technology enhancement, technology can be used to support the IC. The "consultant" system, ICE, designed and implemented for this dissertation guides end users in their use of the IC resources. It not only facilitates responsible resource assignment, but may also increase the productivity of consultants.

The assumption is that some of the consultants' responsibilities can be offloaded by coding and transferring the "know-how" of the IC personnel, and facts about the IC resources, into an expert system. The "expert" is the IC consultant who knows how to recognize and solve problems in the areas of software and training needs. This expert collects facts, investigates, forms rules in decision making, and makes inferences. The Information Expert System (ICE) models these activities; it orients new users to IC services, consults on recurring problems that have software or training solutions, and maintains a history of user interest and experience in order to make appropriate updates in software recommendations.

ICE does not replace the Information Center personnel, but is, in effect, a front-end processor for the IC [Heltne et al. 1987]. It complements the human consultant by handling the ordinary and recurring requests, screening users for specific needs, and making suitable suggestions based on those needs. This will reduce the dependencies on IC personnel, and allow users access to the current inventory of IC resources.

Knowledge systems are recommended as a useful technology in situations where (1) expertise is needed in many different physical locations, and (2) where experts agree on solutions [Waterman 1986]. Information centers are instances where the first of these criteria is true, but experts at the different locations do not necessarily agree on the software solutions given to end users. Therefore, the architecture for the knowledge system supporting information centers was designed to allow each specific site to model its own set of solutions without changes being made to the rule base. The knowledge about software tools supported by each center was extracted and translated into data structures used by the expert system in reaching a recommendation of software for end users. This knowledge is location-variant. That is, it varies depending upon the site in which it is used.

Even within the bounds of a single IC, the data used by the system is time-variant, because supported software changes frequently. This leads to a tremendous maintenance problem if the knowledge base of the expert system must be updated with rule changes by a knowledge engineer. The problem of acquiring and maintaining knowledge in such an environment has been a major bottleneck in expert systems research. In Chapter 5, an architecture is presented for an expert system that can deal with both location-variant and time-variant data.

The physical structure of the ICE system is like that proposed by Frederick Hayes-Roth [1984] in Figure 1.2. The expert system is presented as a computer tool with distinctive development and operational environments. Using the tools on the left side of the diagram, the knowledge system is constructed by the knowledge engineer. The knowledge system itself, shown in the center, consists of three key components: a knowledge base of facts and heuristics, an inference engine, and a user interface. These components are developed within the programming environment selected by the engineer. Once development is completed, the system enters operation, and often accesses external data bases, connects to communications networks, integrates with existing equipment, and/or receives data directly from sensor systems.

This research concentrates on the portions of the diagram showing knowledge acquisition and maintenance in the development environment, and the design of a rep-

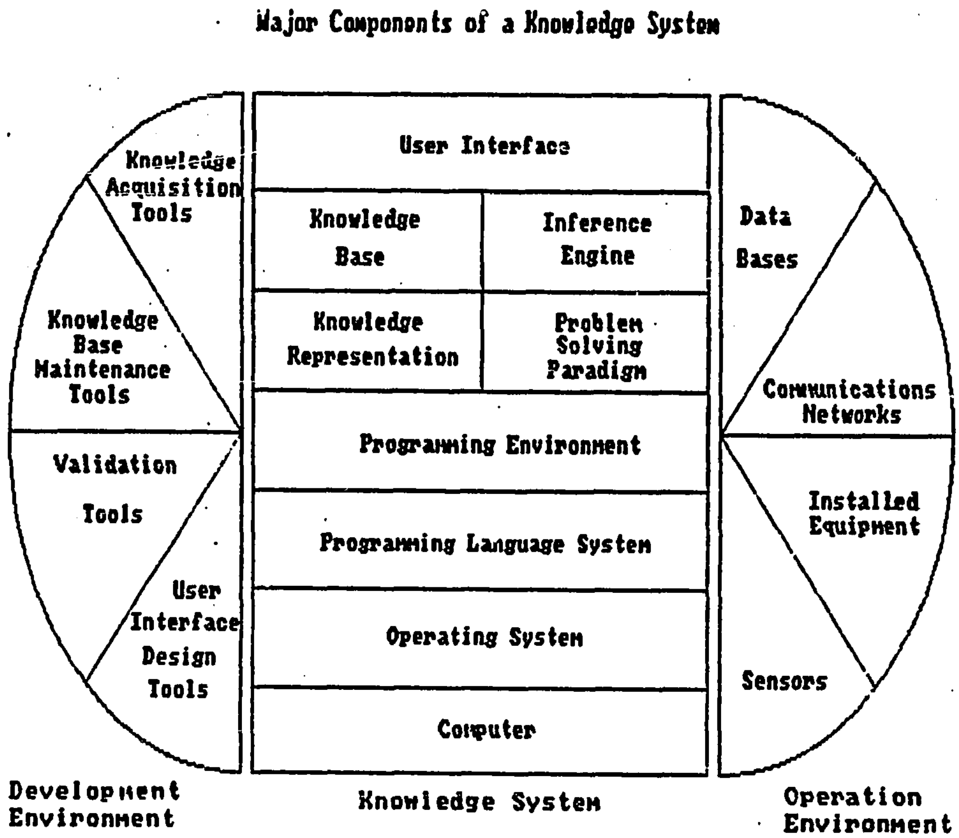

Figure 1.2: Physical Structure of Knowledge Based System [Hayes-Roth 1984]

resentation scheme that accommodates data which is location- and time-dependent, as previously described. The major research goals and questions deal with designing and implementing methods and tools for resource identification and matching in the dynamic environment of an Information Center. Those goals and questions are now stated.

1.2 Statement of Problem

The major objective was stated at the beginning of this chapter, and is incorporated here into the overall goal:

> The goal of this research is to determine whether technology and decision aides can be designed and implemented that can, even in an environment where facts are uncertain, effectively identify resources and facilitate an effective assignment of organizational resources to users with specific needs or problems.

Four subgoals are identified in order to reach the goal; for each subgoal, a research question is posed and a methodology is proposed for answering the question. The first three subgoals deal with knowledge acquisition and maintenance in an environment that has time-dependent data, and the last deals with flexibility and transportability of a knowledge base that has location-dependent data. Subgoals 1 and 2 are the subject of Chapter 4, Subgoal 3 is addressed in Chapter 5, and Subgoal 4 in Chapter 6.

Subgoal 1: Conceptual Architecture

> Subgoal 1. Propose a conceptual design of the architecture for an expert system in which the attributes of users and resources are identified and matched within the context of the user's problem. The architecture must allow the representation of knowledge that is both location- and time-dependent.

Research Question 1: What expert system architecture will allow for the creation and maintenance of a knowledge base where the knowledge is both time- and location-dependent?

Methodology 1: A system development methodology often used in software engineering, prototyping, is utilized to design the general architecture of the knowledge acquisition tools, knowledge maintenance tools, and the expert system itself.

The architecture is described in Chapter 4.

Subgoal 2: Knowledge Elicitation Methodology

> Subgoal 2: Develop and implement a methodology called Resource Attribute Charts (RAC) for the effective identification and weighting of the attributes of available resources.

Research Question 2: Can a valid and reliable methodology for identifying and representing the characteristics (attributes) of users and resources be developed to acquire knowledge and maintain a knowledge base for an environment with time- and location-variant data?

Methodology 2: The knowledge acquisition methodology named Resource Attribute Charts (RAC) is developed, based on Kelly's Repertory Grids from psychology. The methodology is used in a case to build the knowledge base for ICE (Information Center Expert). The attributes are used by the expert system to match users and their problems to the available resources.

The Resource Attribute Charts methodology of knowledge elicitation is presented in Chapter 4.

Subgoal 3: Implementation and Validation of ICE

> Subgoal 3. Implement and validate a specific instance of the proposed architecture of the expert system (ICE: Information Center Expert) and the

> accompanying maintenance tool (M-ICE: Maintenance for ICE), within the context of managing resources for end user computing in the Information Center.

Research Question 3: Can the use of an expert system facilitate the selection of resources for users in a dynamic organizational setting, specifically the setting of the Information Center?

Methodology 3: The conceptual architecture of a generalized resource management system is implemented in a specific case described in Chapter 5. ICE is built and installed at two major corporations and at the Center for Management of Information at the University of Arizona. The validity of the recommendations given by the ICE system is demonstrated.

Subgoal 4: Verification of Transportability

> Subgoal 4. Demonstrate, through pre- and post- studies of the second information center where ICE was installed, that the knowledge acquisition methodology (RAC) and knowledge maintenance tool (M-ICE) makes possible the transport of the knowledge base from one location to another, allowing each site to identify and define its own set of resource solutions without the need for a knowledge engineer to change the rule base.

Research Question 4: Do the acquisition and maintenance tools (RAC & M-ICE) provide for the adaptation of ICE to different information center locations without changes to the rule base?

Methodology 4: A field experiment is conducted, using 20 cases in pre- and post-tests. This experiment is described in Chapter 6.

Realization of the first three subgoals of this research would demonstrate the feasibility of an expert system that facilitates the consultation process for resource assignment in a dynamic environment. Achievement of the last subgoal would show that the proposed design of the system architecture allows for local customization of the knowledge base without the need for rule changes.

1.3 Overview

Chapter 2 summarizes the literature relevant to this research. Chapter 3 discusses the research process. Chapter 4 presents the methodology developed for knowledge acquisition and maintenance of the knowledge base, and Chapter 5 summarizes the implementation of those methodologies in a case where the expert system was installed at a corporate Information Center site. Chapter 6 reports the results of the transportability study. Finally, Chapter 7 discusses contributions and future research possibilities.

Chapter 2

REVIEW OF THE LITERATURE

The literature review consists of three sections. The first discusses studies of the management of the resources associated with end user computing, particularly those in which the Information Center is the vehicle for management. The second section reviews previous research in expert systems as they are applied to business, and specifically to resource management. The knowledge acquisition phase of expert systems development is emphasized in the third section.

2.1 Literature on the Management of EUC Resources

Dr. John Richardson, U. S. Department of Commerce, declared at the 1970 ACM Conference, "Information conserves other resources through improved decisions" [Carlson 1979]. Information is thus perceived as a vital corporate asset that should be invested in, controlled, and used in the management of other resources. The responsibility for the management of the information resource no longer belongs solely to the centralized information group, but is increasingly shared by end users who are demanding the information support.

This sharing of responsibility offers great promise in productivity gains for end users but needs to be managed and guided, as documented by Dickson et. al [1985]. Their study reported that facilitation and management of end user computing was ranked by information systems (IS) executives as the second most important IS management issues of the 1980's, exceeded only by IS planning.

2.1.1 The Rise of End User Computing

In the last two decades, organizations have experienced tremendous growth in end user computing. Summarizing the evolution of end user computing, Richard Benjamin [1982] reports that in 1970, end user computing at Xerox corporation was a negligible component of the 3.5 million instructions per second (MIPS) computing capacity. By 1980, EUC had grown to almost 40% of the 70 MIPS installed capacity. That is an increase of twenty times in total capacity in a decade. By 1990, EUC is expected to represent about 75% of the total workload, and the total capacity will probably increase to between 1350 and 2700 MIPS. While total capacity will grow by a factor of 12.5 to 25 during this decade, EUC will grow by a factor of 39 to 77.

EUC growth follows a pattern similar to the stages described by Nolan and Gibson [1974]. They observed that organizations go through four stages in the introduction and assimilation of new technology: (1) Introduction, (2) Proliferation, (3) Control, and (4) Mature Usage. During the proliferation stage, the idea catches on and spreads quickly until costs often get out of hand. In the third stage, control is exercised in order to contain the growth of costs. Finally, mature usage occurs in the fourth stage. An organization can be in several stages simultaneously, for different forms of technology.

McKenney, McFarlan and Pyburn [1982] also identify four phases of technology assimulation: (1) Identification and Initial Investment, (2) Experimentation and Learning, (3) Management Control, and (4) Widespread Technology Transfer. This version of the four phases has the advantage of casting the important second phase in a somewhat different light from Nolan and Gibson's negative connotations; "proliferation" leaves the impression that users are being almost irresponsible by adopting the new technology too rapidly. However, this is the stage when learning and experimenting take place. It is a trial and error phase. If too much control is exerted too soon, important new uses of the technology may be discouraged. However, control is necessary if the technology is to be used effectively in the organization. When the technology in question is the introduction of personal computers and user application software, management must find a way to enforce standards and offer guidelines for effective computer use by end users. The structure commonly adopted as a control and support mechanism for end user computing is the Information Center.

2.1.2 Information Centers–Support and Control for EUC

The introduction of the Information Center concept is credited to IBM/Canada, and is defined by IBM as a function which can exist within–or alongside–the traditional data processing department. It interfaces with end users, guiding them in the application of easy-to-use interactive tools, program packages and techniques to enable them to solve their own problems [Stearns 1984].

2.1.2.1 Definition of the IC Function

Information centers are one framework for facilitating and managing the computing capability of the end users. ICs are a mechanism for making prospective computer users aware of alternatives, and for matching users' requirements with those alternatives [Konsynski 1984- 85]. It is fundamentally a consulting and service facility that allows departmental users access to their own data, wherever it resides. ICs aim to provide training and guidance for end users, to achieve the right mix of hardware and software, and to make available as many of the corporate information resources as deemed judicious by corporate policy [Atre 1986].

ICs offer answers to several problems that organizations presently face. While allowing for sharing of data, the computer resources remain under centralized control, allowing better management of people and resources. Support specialists are pooled and can serve many departments. ICs have been described as an organization specifically designed to produce "guided service to help users help themselves" [Leitheiser and Wetherbe 1985]. They provide a framework for developing the end user capability in a controlled, phased fashion. The need for guidelines in the evolution of end user computing capability is already leading to increasing adoption of the IC concept. Atre [1986] reports that impressive numbers of businesses have implemented ICs. IC installations have been growing at a 20 percent rate since 1980. Over eighty percent of the companies with more than one billion dollars in annual revenues have ICs, and many have multiple centers.

2.1.2.2 Services Currently Offered by ICs to End Users

The importance of services offered by ICs was measured by Brancheau, Vogel and Wetherbe [1985]. Users rated troubleshooting–hotline/technical support (77%), consulting (74%), and training (64%) as the most important services. Database extraction and research on new products were also mentioned (33%) as useful services. Newsletter services achieved the lowest ranking, being mentioned by only 28% of the users.

A more recent report of interviews with 25 companies with ongoing information centers [Carr 1987] shows that eighty percent of surveyed managers indicated that user support was a prime duty. Unfortunately, this term is not defined in more detail. Fifty percent of the respondents indicated training and education, and 45% listed consultation as important duties. Other responsibilities include product evalutation and planning (30%), troubleshooting and problem resolution (20%), technical interface with the data processing department (15%), and assistance on PCs (10%).

Canning's study of information centers [1983] found that the uses of the computer by end users could be categorized as shown in Figure 2.1. The study showed that executives, managers, and professionals were the users most likely to perform most of the activities referred to on this list. Secretaries might use a number of the capabilities and might spend more hours per day using their computers, for word processing, for instance, than the executives, managers and professionals, but the latter probably would make much broader use of computers than do secretaries. It is this new, more unfamiliar use by executives, managers, and professionals, that constitute the larger challenge.

1. accounting, reporting, and calculating aids
2. writing aids
3. search and retrieval aids
4. communications aids
5. presentation aids
6. planning, scheduling, monitoring aids
7. analysis aids
8. record processing aids
9. aid to developing new programs

Figure 2.1: End User Usage By Category [Canning 1983, p. 4]

One of the earliest survey of end users [Rockart and Flannery 1983] shows a diverse range of applications important to end users (Table 2.1). The primary focus of about ten percent of the applications were "operational" paperwork processing systems such as inventory systems or commission check producing systems. Another fourteen percent were report generation systems, often involving information databases taken from production systems which turned out reports regularly or on demand. Twenty percent of the applications provided software merely to extract particular data items from the database, or to do simple command-level manipulation of items in the database. Over half the applications supported more complex analysis of data. Included in these systems were financial analysis, engineering calculations, simulations, operations research and optimization models.

Table 2.1: End User Applications [Rockard and Flannery 1983, p. 779]

Purpose	Number	Percentage
Operational paper processing	24	9
Report generation	39	14
Inquiry/simple analysis	58	21
Complex analysis	135	50
Miscellaneous	15	6
Total	271	100

According to a study done by IBM on its own information center tools, the most popular were interactive query processing and report generation. These allow end users to access corporate data without the need for programming help. Other popular software tools included a financial planning system, a text processing system, and the PROFS office system which included computer messaging [Lodge 1983].

Survey respondents to an American Management Association questionnaire [Bohl 1986, p. 24] reported the following frequency of requests by end users for service (Table 2.2). The most frequent requests (right column) are for spreadsheet and word processing assistance.

Atre [1986, p. 162] reports that "software consulting is the IC's forte." Her study shows that 40 percent of ICs regularly consult on graphics and statistical analysis software, and almost as many on financial modeling programs. The study suggests that at a bare minimum, the IC library should include software in eight categories: (1) spreadsheets for numerical analysis, (2) word processors for company correspondence and reports, (3) business graphics, (4) statistical reporting, (5) communications software for remote data

Table 2.2: AMA End User Report of Frequency of Service Requests

Nature of Request	Provide Such Service	"Very Frequent Request"
Assistance in Hardware Selection	90.8%	41.0%
Assistance in Software Selection:		
Spreadsheets	89.8%	50.5%
Database Management	89.2%	34.0%
Word Processing	87.1%	47.4%
Graphics	85.2%	25.1%
Communications	81.8%	19.1%
Project Management	75.4%	9.6%

access, (6) electronic mail, (7) project managers, and (8) data management.

The AMA study referred to above [Bohl 1986] reported that another significant activity of ICs is training. Only one IC in twenty escapes some sort of training responsibility. The major training load has to do with microcomputers; 83.4 percent train end users in micro hardware, 86.4 percent in micro software. Mainframe software training is provided by 68.8 percent of the sample.

Single ICs often employ a number of methods to train different personnel in different hardwares and softwares. The respondents reported using in-house trainers (58.7 percent), self-instruction (22 percent), contracted providers of training programs, either on-site (10.7 percent) or off-site (8.6 percent). All of the respondents in the study report increasing use of computer-based training. 1986 marked the first year that more training programs used tutorial disks in their instruction than written manuals or audio cassettes.

2.1.2.3 Anticipated Support Services

The respondents to the AMA study [Bohl 1986] were asked to predict what changes they foresaw in their ICs within the next five years, and many pointed first to the training role. Common answers were "Less instruction in the basics," "More consultation, less teaching." One respondent noted, "As it stands now, we do all their thinking for them. They won't even try to solve a problem without consulting us. This just has to change." The IC needs to find ways to offload the basic, routine questions so that they can provide a larger variety of support services: networking, access to corporate data, closer work with MIS in development, and more applications work for the user clientele [Bohl 1986].

2.1.3 Problems Faced by Information Centers

Most powerful tools, including computers, bring with them some unfortunate side effects. That experience has been true for end users as well as for data processing departments. As more and more people in an organization begin to demand their own hardware and software, costs accelerate, incompatible personal computers are obtained, and poorly structured and undocumented programs are written. End users need education and training to promote understanding of the programs and data they will use. Without understanding of what they are doing, the users can generate invalid results [Gulden and Arkush, 1983].

As stated in Chapter 1, Brancheau, Vogel, and Wetherbe [1985] reported that end users expect to be even more dependent on the IC in the future than they are now, that they anticipate needing more support services and training, and that it will be more important than ever to remain "current" on new applications of technology. Thus, information centers are being subjected to increased user expectations, higher demand for integrated applications, and growing pressure to accomplish more with fewer resources. These pressures have placed information support services in a difficult situation.

2.1.4 Support for the Information Center

The anticipated huge growth in EUC means that organizations should make preparations for it. The IS function should be able to influence and handle the growth satisfactorily when it occurs. Gulden and Arkush [1983] suggest a method that includes several strategies. One is identifying who the current and potential end users are, ranging from operational managers to staff to executives. Another strategy is identifying what these users will need, from single purpose support to generalized tools and data. A third strategy suggested is identifying how best to support the users, employing tactics "ranging from 'high priests' to coaches to a telephone 'hot line,' as well as identifying where the support should be located" [Gulden and Arkush 1983, p. 418].

Having identified who the major groups of end users are, or who they are likely to be, the IC staff must then offer guidance in selecting equipment and software, training, providing controlled access to company data and downloading selected data. The support that end users demand has already overwhelmed many information centers.

Harmon and King [1985, p. 198] suggest that knowledge systems are particularly helpful in places "where a few key individuals are in short supply...(where) they spend a substantial amount of time helping others." Just as the IC supports the effective use of technology in the organization, technology can be used to support the IC in that effort. Knowledge (expert) systems can capture the knowledge of senior clerks, mechanics, or managers, and "provide new people in an area with the information the existing employees already have" and can be used as procedural guides to "walk inexperienced persons through the task" [Santarelli 1985, p. 26].

2.2 Expert Systems Literature

Many technologies have been developed and used successfully to assist in resource management. As computers became increasingly larger and faster, mathematical programming became a useful means of supporting scheduling and distribution activities and in assigning personnel, facilities and materials to jobs or tasks. All these activities can be regarded as resource management.

Mathematical programming techniques consist of algorithms that assign resources to users, whether those users be projects, departments, investments, or individuals. The mathematical models, such as liner and nonlinear programming, integer programming, and dynamic programming, are examples of areas in which computers have been useful and where human capabilities are limited: high-speed calculating and storing/retrieving enormous amounts of data. They are algorithmic programs, "completely defined, step-by-step procedures for solving problems" [Webster and Miner 1982, p. 62].

These exact computations can solve only a fraction of human problems. What is missing from many of these solutions is the ability to model real world situations where not every variable is known with certainty, and where decision makers follow rules of thumb–heuristics–in making decisions under conditions of uncertainty. The challenge is to devise computer programs that can emulate the way humans employ past experience to solve new problems.

This challenge has been pursued using expert systems technology. Tanimoto [1987] defines an expert system as "a computer system or program which incorporates one or more techniques of artificial intelligence to perform a family of activities that traditionally would have to be performed by a skilled or knowledgeable human" [p. 461]. These systems solve real-world problems, and have become increasingly popular in the business world because they support problem- solving, diagnosis, advising, decision making, and control activities.

In contrast to data processing systems which automate time consuming clerical functions by collecting and processing large volumes of data algorithmically, expert systems are specialized, often addressing small tasks typically performed by professionals in a few minutes or hours. Examples are interpreting, diagnosing, planning or scheduling [Hayes-Roth 1984]. Unlike what happens in the algorithmic data processing approach, the expert system generally examines a large number or possibilities or constructs a solution dynamically. It is not locked into any specific decision path as traditional programs are. It picks from alternative paths in its search for a conclusion, and it weighs facts and assumptions, making choices appropriate for each particular problem presented to it [Tanimoto 1987, p. 461].

The 1980s have witnessed a tremendous growth in the number of successful applications of artificial intelligence (AI) expertise to real-world systems. High on the list of AI technologies that have been applied in the marketplace are expert, or knowledge-based systems. The rapid formation of expert system companies, often in close collaboration with major academic AI research centers, and the interest shown by large corporate research and development departments, attests to the belief in the economic viability of this technology transfer [Clifford, Jarke, and Vassiliou 1982].

2.2.1 Expert Systems for Managers

The development during the last two decades of expert systems for prediction and diagnosis gives rise to the suggestion that such systems would be useful to managers as well. Blanning [1984a, p. 2] suggests that expert systems for managers "would contain judgmental assumptions and rules that a knowledgeable and experienced manager uses in arriving at a recommendation or decision and would analyze them in a way that would be useful to a practicing manager." Such a system

> captures the specialized knowledge that managers bring to bear on the decision-making tasks they perform, and it uses this knowledge to diagnose potential

> or actual problems, make recommendations, and offer explanations of its diagnoses and recommendations [Blanning, 1984b, p. 311].

There are three types of managerial tasks for which such expert systems have already been developed: problem diagnosis, scheduling and assignment, and resource allocation [Blanning, 1984a]. The goal of research in these areas has been to provide tools that exploit new ways to encode and use knowledge to solve problems, not to duplicate intelligent human behavior in all its aspects [Duda and Shortliffe 1983]. Examples of expert systems that support the managerial tasks listed above are now given.

2.2.2 Problem Diagnosis

A diagnostic problem has been defined by Reggia, Nau and Wang [1984] as a problem in which one is given a set of abnormal findings for some system, and must explain why those findings are present. Problems of this kind are very common in management, so general methods for expert systems which support the decision making of human diagnosticians is an important issue. Examples of diagnostic systems follow.

Bouwman [1983] describes an expert system that evaluates the financial performance of a company based on the financial statements and published data about that company and its industry. It is a small rule-based system, using a small number of quantitative indicators judged most important in modeling the diagnosis process.

Dungan [1983] describes a system that simulates an auditor making bad debt decisions on accounts receivable. The system, called AUDITOR, decides whether delinquent customer credit accounts should be reported as collectable in a company's financial statements.

Wicklund and Roth [1987] discuss the development and implementation of an insurance underwriting expert system. The underwriting function reviews applicant data for determination of insurability. Each driver is classified in terms of age, gender, years of driving experience, and driving record in order to develop an estimate of that driver's risk level.

ACE, developed by AT&T Bell Laboratories, provides trouble-shooting reports and analyses for telephone cable maintenance. The system analyzes maintenance activity data and generates reports describing the physical location of the trouble and the characteristics of the network at that spot. ACE is not an interactive system, but rather interfaces with a database management and report generation system that contains information about telephone network repairs. Forward chaining inference techniques are used in a rule-based system [Wright and Miller 1984].

Financial Advisor system from Palladian Software, Inc., analyzes a proposed investment in, for instance, a new plant, product, or acquisition of another company [Canning 1985]. The difference between Financial Advisor and a spreadsheet analysis is that with the latter, the user decides which factors to include in the model and which to leave out. If something is overlooked, no correction is made. With Financial Advisor, all the factors that ought to be considered in such an investment are included by the system. The expert system can also be customized to a particular company.

2.2.3 Scheduling and Assignment

Expert systems have been developed for office scheduling and personnel assignment. Fikes [1981] presents Odyssey, a frame-based system for office scheduling and the scheduling of business trips. A similar system called NUDGE [Goldstein and Roberts 1982] is used in scheduling meetings. Barber [1983] discusses a personnel assignment system based on IF/THEN rules describing personnel capabilities and job requirements.

2.2.4 Resource Allocation

The problem of allocating resources to users can be eased by identifying the relevant attributes that must be matched between user and resource. The resources matched to users can be objects, procedures, information, money, or tools. They can be tangible or intangible. Several expert systems for resource allocation have been developed. A few of the more successful ones are now described.

In an early work, Clarkson [1963] presented a model for portfolio management in which a system prepares a portfolio using information about the financial performance of stocks and preferences of the clients. A later effort by Cohen and Lieberman [1983] resulted in a system called FOLIO, which helps portfolio managers determine client investment goals and select portfolios that best meet those goals. FOLIO determines client needs during an interview, then recommends percentages of each fund that provide an optimum fit to client's goals. The goals are inferred by forward chaining through a rule base; then a linear programming scheme is used to maximize the fit between goals and the portfolio.

DECMAK uses rules about the performance of various types of equipment to allocate funds to the purchase of equipment [Bohanek, M. et al. 1983]. DECMAK is a tree- structured system in which rules are used at each level to give variable values at the next level. It provides the user with an evaluation of the various alternatives, and an explanation of how they were computed. Limited features are available for sensitivity analysis and report generation.

Slagle and Hamburger [1985] describe an expert system for resource allocation in a particular military domain. The system, called Battle, acquires reliable information of two kinds: generally applicable expertise in the subject matter, and information from a user that is specific to the current situation. The system constructs an allocation tree to determine weapon allocation recommendations to military commanders in combat situations. Knowledge about the weapons and the combat activity is represented in rules with certainty values. Battle was developed by the Naval Research Laboratory.

Blanning [1984] suggests that expert systems for resource management might also include environmental considerations–laws, contractual requirements, government regulations, or accounting rules–and these might be described by logic expressions. Systems should contain knowledge needed to support a variety of resource allocation decisions.

2.3 Knowledge Acquisition Literature

Two aspects of an expert system distinguish it from more traditional computer systems: overall architecture, and method of development. The architecture of an expert system consists of two interacting components: a "knowledge base" and an "inference engine." The knowledge base is usually divided into two components: the specific "facts" known by the expert, and the more general principles, rules or "problem solving heuristics" which come from the accumulated experience in the field. The inference engine is the mechanism that manipulates the rules to make inferences and decisions. For resource management expert systems, the knowledge must include facts about available resources, and heuristics about how the resources are used most effectively. In the specific implementation of a resource management expert system in an Information Center (IC), knowledge of IC resources and methods for allocating them must be modeled.

The other distinguishing aspect of knowledge-based systems stated above was the manner in which they are constructed. The knowledge and heuristics used by the human expert must be transferred to the knowledge base. "This dialogue is the least understood process in the expert system paradigm" [Clifford et al. 1982]. This dialogue is often called "knowledge acquisition," and is the first part of the iterative approach called expert system prototyping. Just as the human expert never stops developing or expanding expertise, the expert system must be structured to easily facilitate change or continued growth of its capabilities.

"Knowledge does not come off-the-shelf, prepackaged, ready for use" [Hayes-Roth 1984, p. 18]. The process of extracting knowledge involves eliciting from experts or other sources the basic concepts of the problem domain. Getting an expert to articulate problem solving knowledge is one of the main problems in building expert systems. Often the knowledge is ill-specified or incomplete because the experts themselves don't always know exactly what it is they know about the domain [Feigenbaum and McCorduck 1984]. Acquiring such knowledge usually involves "a long series of incremental interview, build and test cycles" [Boose 1986, p. 69].

There are a variety of ways knowledge can be acquired for use in an expert system: being told, analogy, example, observation, discovery, experimentation, reasoning from deep structure [Michaelsen et al. 1985]. These manual methods are very labor-intensive, and the process of transferring knowledge from human experts into the software knowledge base is a major difficulty in the building of expert systems [Michalski 1983]. The transformation of the human knowledge into decision rules is no simple matter, because experts typically do not structure their decision- making in any formal way, and may have difficulty isolating and describing the steps of their reasoning [Shortliffe 1976].

Feigenbaum and McCorduck [1983, p. 80] state that "if applied AI is to be important in the decades to come. . .we must develop more automatic means for what is currently a very tedious, time-consuming, and expensive procedure." Several projects have investigated ways of easing these problems, and where possible, automate the knowledge acquisition process. The efforts are varied in their approaches to solving the problem, but all have attempted "to provide a framework within which the system can guide the expert in communicating his/her expertise to the system" [Clifford, Jarke and Vassiliou 1983, p. 9].

Buchanan et al. [1983] present the development of an expert system as an evolutionary process. The key task, acquiring knowledge from experts, resists linear, one-pass techniques. Instead, knowledge acquisition and system- building interact inseparably. Automated tools that can aid this process include induction programs and intelligent editing programs. A few of the most significant results follow.

2.3.1 Induction Programs

The principle of induction is that "the expert provides a set of examples of different types of decisions, called the training set. He also supplies the relevant factors, often called the attributes, influencing the decision" [Hart 1985, p. 25]. The algorithm then uses the training set to induce general principles and formulate the decision process. From these, the system can predict decisions for examples not contained in the original set.

There is some controversy over the role of machine induction. While some authorities dismiss the methods almost worthless, others [Michie and Johnston 1985] stress its advantages and suggest that in the future, inductive systems will be useful sources of knowledge. An advantage of induction programs is that the expert often finds it easier to provide examples of cases rather than describing the decision-making process itself. It allows the expert to describe "what" rather than "how".

A well known example of induction is the work of Michalski and Chilausky [1980], who studied soybean diseases, producing two sets of diagnostic rules. The first set was obtained directly from the expert, and the second was induced from examples. The training set consisted of descriptions of the environment, the conditions of the plant, and the diagnosis of the expert. When presented with new examples, the induced rules behaved much more efficiently than the expert.

2.3.2 Intelligent Editors

Intelligent editors are intended to eventually replace the knowledge engineer by allowing the expert to interact directly with the system [Fellers 1987]. There is currently great interest in the development of these programs, and a few of the most significant efforts are now introduced.

The best known example of an intelligent editor is TEIRESIAS, a program which facilitates the interactive transfer of knowledge from a human expert to the expert system through a high level dialogue conducted in a restricted subset of natural language [Davis 1979].

Following in the footsteps of TEIRESIAS is MORE, a program that elicits knowledge from the domain expert and builds a knowledge base. The knowledge base can then be interpreted by an inference engine to perform some heuristic classification task. MOLE is the successor of MORE. It differs from its predecessors in that its heuristic assumptions are made explicit and are exploited in the knowledge acquisition process. MOLE is "a knowledge acquisition tool that uses its head" [Eshelman and McDermott 1986, p. 950]. It is "smart" in that it asks as few questions of the expert as possible while still building a reasonable knowledge base for performing a task. MOLE's approach to knowledge acquisition is to use its heuristic assumptions about the world and assumptions about how domain experts express themselves to clarify knowledge elicited from the expert.

Another editor, ETS (Expertise Transfer System) first interviews a human expert, then constructs and analyzes aspects of the knowledge that the expert uses to solve a particular problem [Boose 1986]. ETS is used to interview experts to uncover vocabulary, conclusions, problem-solving traits, trait structures, trait weights, and inconsistencies [Boose 1987]. It is then used to construct rapid prototypes, often in just a few hours, and aids the expert in determining if there is sufficient knowledge to solve the problem.

ETS is based on methods from psychotherapy developed by Kelly [1955] called Personal Construct Theory, in which each individual is a personal scientist seeking to predict and control events by forming theories, categorizing, testing hypotheses, and weighing experimental evidence. In order to interpret the results in a meangingful way, a method is adopted from the repertory grid [Kelly 1955]. The grid is used together with the computer to enable the person to examine his own conceptual structure. The method assumes that each person can express this structure as a "unique system of bipolar dimensions known as personal constructs through which he experiences life, and categorizes his experiences" [Shaw 1981, p. 33].

The personal construct model matches the expert at work, in that it describes the development and use of his knowledge. The grid helps elicit and analyze the model. It is a two-dimensional array of observations, interlaced so that each dimension has meaning in the context of the other. It is a "finite system of cross-references between personal observations an individual has made and the personal constructs he has erected to make sense of his experiences" [Shaw 1981, p. 33]. The set of constructs represent how a person thinks and feels about a certain topic, and these personal observations are known as elements. The elements are chosen from the set of all possible observations that are relevant to the current purpose. Care must be taken to insure that the elements are known and meaningful to the individual, and that each construct is important in the context of the particular problem.

This grid is composed of constructs and elements, which are similar to the attributes and examples of induction. The construct is a bipolar characteristic which each element has to some degree. Examples of constructs are heavy-light, large-small. As the expert selects the construct he must understand what makes a valid construct, and how it is used. The bipolar definition of constructs can be extended to include a scale of numerical values (1-very heavy, 2-heavy, 3-medium weight, 4-light, 5-very light) [Hart 1986, p. 134]. Each element is then rated by the expert according to each construct, thus enabling elements to be ranked or compared.

The repertory grid, often used in psychology, is easily adapted as a way for an expert to represent thoughts about a particular problem. By comparing different objects he can set up a grid of elements and constructs that describe his perception of a particular problem. The method helps the expert focus on key issues, and is an easy method for categorizing and rating important principles.

2.4 Summary

Literature from three areas was relevant to this investigation. The first dealt with the management of EUC resources. The rise in the demand for computing resources by end users is documented in studies cited in this chapter. These resources are commonly managed by the organizational structure called the Information Center. As requests for service from IC personnel increases, new ways to support the IC activities will have to be found.

Expert systems literature was searched for relevant applications to management problems. Reports of previous research in expert systems for management indicate that systems are being implemented successfully in situations demanding problem diagnosis as well as scheduling, assignment, and allocation of resources. Current systems exist in environments where knowledge is relatively stable.

Knowledge acquisition literature was summarized. Acquiring knowledge from the experts continues to be a tedious, time-consuming procedure. Important efforts to structure or automate the process were briefly described.

Chapter 3

METHODOLOGY

The research process is adopted from a model proposed by Michael S. Scott Morton [1984, p. 24] as a typology for research in information systems development. It includes the activities of (1) constructing a methodology, (2) building a prototype, (3) describing a case, and (4) performing empirical tests, either laboratory or real world. These steps were followed in this study and are discussed in terms of the prototyping systems development methodology.

Waterman [1986, p. 137] divides this prototyping methodology into five phases: (1) identification of the important features of the problem, (2) conceptualization– deciding what concepts, relations and control mechanisms are needed, (3) formalization of the knowledge representation, (4) implementation of that formalized knowledge into a working computer program, and (5) testing–evaluating the performance and utility of the expert system, revising as necessary.

3.1 Identificaticn

The important features of this problem were described in Chapter One. Information Center consultants possess skills used in analyzing the software needs of end-users. This consultation process must be modelled in a machine-user dialogue that gathers information about user background and current problems. The software resources of the IC must be defined in the knowledge base in such a way that tools can easily be added, modified, or deleted without rule changes. The problem is one of matching user, problem, and resource attributes to result in effective resource allocation for end users.

3.2 Conceptualization

The conceptualization phase involved deciding what concepts and relations were most important in representing knowledge about resources, users, and problems, in the domain of the IC. A methodology that provided adequate discrimination between key concepts was developed and utilized to help the experts in this process. That methodology, called Resource Attribute Charts (RAC), was proposed because of the difficulty in the early stages with eliciting knowledge from the experts (IC consultants). The charts assisted the consultants in deciding what concepts, or elements, should represent knowledge about IC users and resources, and how these elements were related. The RAC acquisition methodology is described in detail in Chapter 4.

3.3 Formalization

A generalized architecture was designed to represent the elements and relationships identified in the previous phase. The architecture models the attributes of users and resources, and matches them in an environment where information is both location- and time-dependent. This resource management architecture is also presented in Chapter 4.

3.4 Implementation

In prototyping, the expert system builder gives the user a succession of mock-ups which capture increasingly the appearance and functionality of the software [Sviokla 1986]. The prototyping methodology serves two purposes. It facilitates the communication between the expert and the knowledge engineer, and it helps the development team discover the expert's problem-solving methods. With each refinement, the knowledge used by the expert is more clearly expressed.

Demonstration, research, and field level prototype systems were built. The final system, ICE (Information Center Expert), was installed in information centers at two corporate sites and one academic institution. The system was implemented on the IBM 4381 using a shell called Expert System Environment (ESE). The Maintenance system for ICE (M-ICE) was developed on the IBM PC and was used to build the knowledge base for ICE after the knowledge was acquired from the IC experts using the RAC methodology. The actual implementation of the generalized architecture design described in Chapter 4 is presented as a case in Chapter 5. "Describing a case often provides a rich sense of the context and nuances of an application" [Scott Morton 1984, p. 25]. The actual attributes describing the resources in that case are included in Appendix A.

3.5 Testing

Scott Morton [1984] indicates that real-world tests should be used to evaluate the effectiveness of the methodology and the system. Two different evaluations of ICE were performed. The first was to determine if ICE made resource recommendations that IC consultants agreed were appropriate–a validity test. The second experiment tested the design claim of transportability.

The validation test was conducted using 20 cases based on real-world activities of end users. IC consultants were asked to make software recommendations based on the skills and expertise of the users, and the descriptions of their current needs for software. The same case was then analyzed by ICE. The expert system is a valid system if the recommendations made by ICE match the recommendations made by the IC consultants, or are judged to be better than those made by the human consultant. The validation experiment is described in Chapter 5 as part of the implementation case description.

In addition to the validation tests, another empirical test was conducted to examine whether the ICE knowledge base was transportable to locations with different resource solutions. The final question from the Statement of Problem is addressed. The question was, "Do the acquisition and maintenance tools (RAC and M-ICE) provide for the adaptation of ICE to different information center sites without changes to the rule base?"

Twenty cases based on actual users and resource needs were given to the IC consultants at the second corporate site where ICE was installed. The IC consultants' recommendations for these cases were then tested against ICE's solutions. At the time of the test, the ICE knowledge base consisted of the tools used at the first site. Thus, it was predicted that, in many cases, the recommendations would not match ICE's solutions.

The knowledge maintenance tool M-ICE was then used to define the knowledge base of the software resources of the second site by assigning values to the resource attributes. No rule changes were made. M-ICE allowed the new knowledge base to be built by the IC consultant without the intervention of the expert system builder. The new knowledge base was tested using a "blind" study. This transportability experiment is described in detail in Chapter 6.

3.6 Summary

The methodology used in the development and validation of ICE was really a spiral of the cycle of activities just described. After the first limited demonstration prototype was tested, the system was expanded to its full capacity and the cycle was repeated. The result was the research prototype system. User testing followed, showing the need for dialogue refinement. That process is not discussed in this disseration.

The final system has evolved to the stage of field prototype, whose validity has been demonstrated experimentally. That system, ICE, will continue to be refined and revised as it is subjected to real problems in the user community.

Chapter 4

DESIGN OF GENERALIZED ARCHITECTURE FOR RESOURCE MANAGEMENT EXPERT SYSTEM

Knowledge, as defined in the context of expert systems development, means "those kinds of data that can improve the efficiency or effectiveness of a problem solver" [Hayes-Roth 1984, p. 16]. It is the key factor in the performance of an expert system and consists of two types: facts and heuristics. Facts of a domain are widely shared knowledge, commonly agreed upon by practitioners of the field. Heuristic knowledge is comprised of "good practice and good judgment" where valid algorithms generally do not exist. It is the experiential knowledge that a human expert acquires over many years.

Human experts solve problems by employing both facts and heuristics. When human problem solving is automated, these domain-specific facts and heuristics must be organized and represented by the architecture of an expert system in such a way that the system can reach the same conclusions as the human expert. Building an expert system thus requires careful selection of representation schemes for the knowledge it is to contain. The representation must have both a logical and a physical structure. Presented here are the logical and physical designs of an expert system architecture which supports the process of assigning organizational resources to users. The system must represent the facts and heuristic knowledge used by managers in resource assignment.

4.1 Logical Structure of the Expert System

At the most abstract level, the logical structure of the resource management expert system for the information center environment is represented by the model shown in Figure 4-1. The power of a model lies in the abstraction and simplification of reality (eg., Robinson 1984). Consequently, the figure identifies only the most crucial elements of the information center environment: end users, their computing problems, IC resources, IC consultants and the policies they enforce.

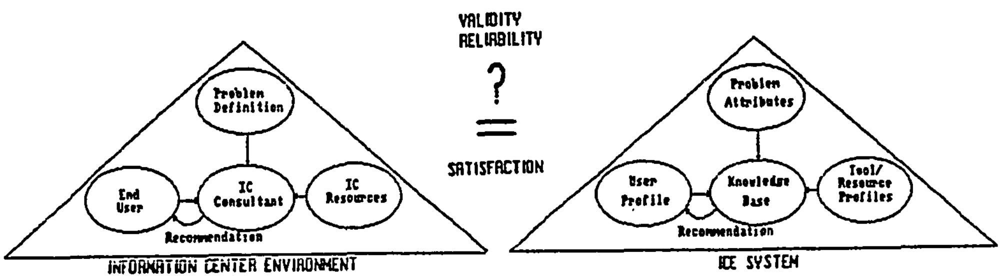

Figure 4.1: Logical Structure of Resource Management System

The model shows knowledge about the resources, users, and problems represented in the knowledge base of the expert system as profiles of attributes describing each of these components of the model. By drawing inferences from the user profile and resource profile within the context of a specific problem, the system then identifies available resources and draws conclusions about the appropriate assignment of those resources to users. The recommendations made by the expert system should not be different from the recommendations made by the average resource manager.

4.1.1 Physical Design of the Expert System

The task of the knowledge engineer is to translate the logical model into a physical design–an architecture with data structures containing the knowledge used by the human expert to perform problem-solving activities. The physical representation is a set of appropriate data structures through which the knowledge in the knowledge base can find its way into the memory of a computer [Feigenbaum and McCorduck 1984, p. 84].

The first two research questions posed in Chapter One are addressed in this chapter:

> (1) What expert system architecture will allow for the creation and maintenance of a knowledge base where the knowledge is both time- and location-variant?
> (2) Can a valid and reliable methodology for identifying and representing the attributes of users and resources be developed to acquire knowledge and maintain a knowledge base for an environment with time- and location-variant data?

The challenge is to represent knowledge of users, problems, and resources as data structures so that they can be conveniently accessed for problem matching, and easily updated and maintained to remain an accurate representation of the dynamic IC resource environment. This task is described in the next section on knowledge representation. The last section of this chapter describes the methodology developed to elicit the experts' knowledge of resources.

4.2 Knowledge Representation

The first research question concerning choice of data structures to represent the facts and heuristics is answered by proposing a generalized architecture of a knowledge system for resource management. The architecture design is used to implement an actual system within an information center, as described in Chapter 5. The problem to overcome is that in the past, expert systems have often not been flexible enough to be updated as quickly and easily as they must be [Hayes-Roth, 1984].

> There is a formidable problem of knowledge-base management. . . How shall knowledge be organized, controlled, propagated, as well as updated in terms of its features and properties, and their relationships with each other in a knowledge base? [Feigenbaum and McCorduck 1984, p. 84]

The difficulties of designing the architecture of a knowledge base increase if the data changes dynamically and needs frequent updating. It has been previously stated that the data used in resource management decisions are not static in time, and are often specific to a particular location. Therefore, important issues that were considered in the design of this architecture were flexibility and maintainability. Maintenance and acquisition of knowledge are addressed in the last major section of this paper. A discussion of flexibility follows.

4.2.1 Designing for Flexibility

Early in the development process, a search of artificial intelligence and expert system literature was conducted to determine if current technologies could be used to represent the kind of knowledge needed in matching users and their needs to available resources. Data structures were chosen to represent (1) the profiles of the users, problems, and resources–facts, and (2) the "rules of thumb" that tell how experts use these facts in their decision making–heuristics. These structures are now described.

The knowledge needed for this environment thus was of two types: a declarative component of facts (describing the attributes and relationships of users, problems and resources in external profiles) and a deductive component that enables reasoning about the facts. Researchers in artificial intelligence have developed a variety of alternative formalisms for knowledge representation that could be used for these two components. Options include logical expressions, hierarchical frames, semantic networks, objects, rules, and procedures. All have previously been used successfully to represent knowledge of an expert in a knowledge system. Each method has advantages and disadvantages.

The data structures chosen had to represent the concepts and intentions of the expert faithfully, facilitate the process of finding gaps and errors in the knowledge base, and had to allow separation of domain knowledge from the interpretation program so that the knowledge base could be enlarged or corrected easily [Duda and Shortliffe 1983, p. 266].

Current research shows a trend toward heterogeneous approaches of representation. Humans tend to exploit several different representations of a single problem during its solution, so the knowledge base representation must do the same.

Figure 4.2 shows the generalized architecture of an expert system designed to make resource recommendations to users. This architecture utilizes heterogeneous data representations. The declarative component is a series of external files which contain attributes describing the various parts of the model: users, problems, and resources. The values of user and problem attributes are set through dialogue with the user, and the values of the resource attributes are set through the knowledge acquisition and maintenance tools described in the last section of this chapter. The deductive component consists of the rules which control how the facts are used. These components are now described in more detail.

4.2.2 Declarative Component

The declarative component uses a data structure that is a simplified semantic network utilizing object-attribute- value triplets [Harmon and King 1985, p. 39]. In this specialized case, the object to attribute link is a "has-a" relationship, and the value to attribute link is an "is-a" link.

Nodes of the network are classified into one of three categories: objects, attributes, or values. Adequate representation of resources and problems requires the use of a modified frame structure to hold the values of the attributes of all the different categories. A "frame" of knowledge is somewhat similar to a "record" in a data file, but has more features than most records [Canning 1985].

Figure 4.3 shows the structure of the objects (users, problems and resources), and their attributes and values. The values are represented in the "leaves" of the tree, and are defined as strings or vectors of real numbers in external Pascal files. The values of the user and problem attributes are obtained through dialogue with the user, and can change with each consultation. User attributes are stored and reloaded each time the user consults, with the option of changing any values if the facts about the user have changed since the previous dialogue with the expert system. Resource attributes are instantiated by the resource manager using the knowledge acquisition methodology discussed in the Knowledge Acquisition section of this chapter.

The values of the relevant attributes about users and problems are grouped together in a single definition of the skills, environment, and needs of the user for the current consultation. This group of attributes is then matched against group attributes for each resource defined in the knowledge base. A resource assignment algorithm computes scores for each resource in terms of how well it matches the user's need, screens out inappropriate resources, and of those remaining, places confidence ratings on those whose attribute values most closely match the user/problem attributes. Those resources with

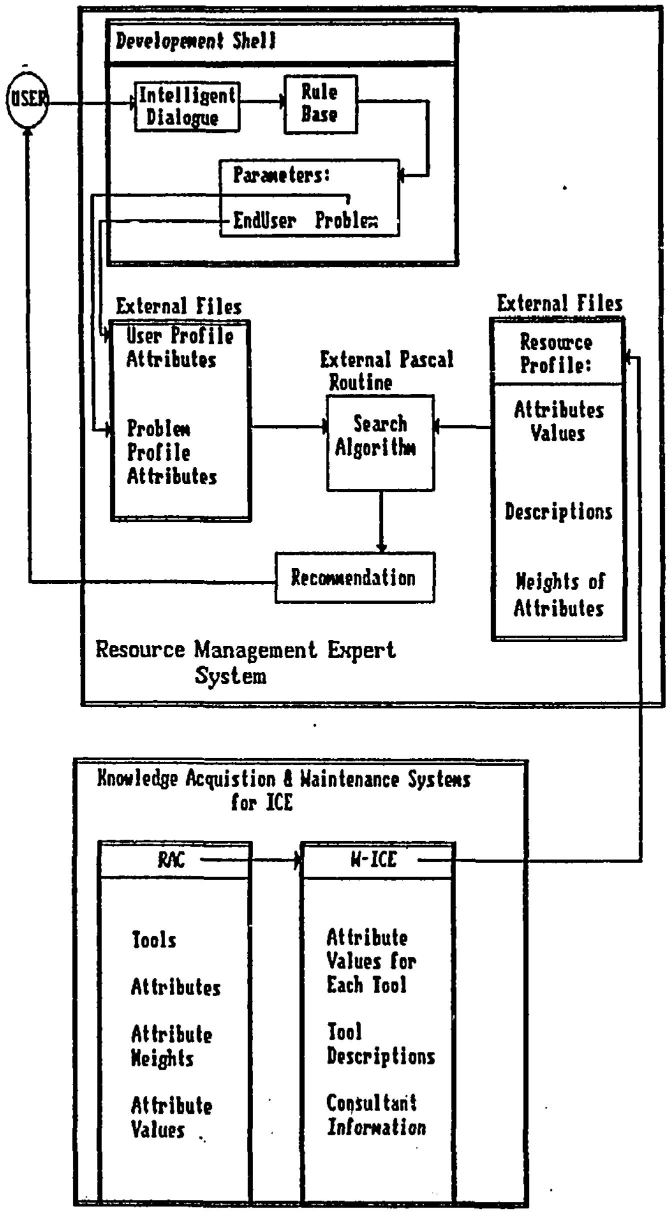

Figure 4.2: Generalized Architecture of Expert Systems for Resource Recommendation

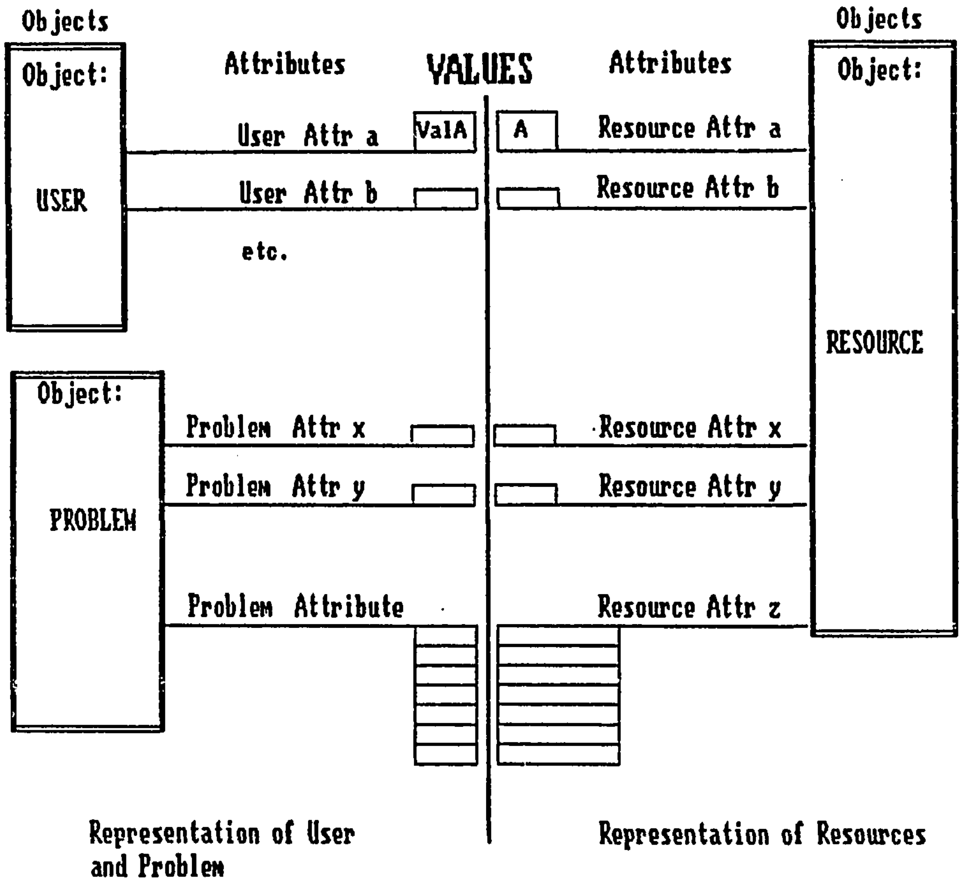

Figure 4.3: Data Structure to Represent Users, Problems and Resources

scores above a specified level of confidence are recommended to the user as solutions to their resource needs.

4.2.3 Deductive Component

In addition to the facts that are represented by object-attribute-value triplets and frame-type structures, rules are needed for specifying how these facts are to be used in reaching a solution. These rules are the deductive component. Standard production rules were created by determining and entering the "if...then.." structures used by the expert in reaching decisions in the subject domain. The rules are of the form "IF *condition* THEN *action*" the *condition* checks the attributes of the user/problem profiles, and *action* sets the values of the corresponding attributes that describe users and their problems. These are then matched against the attributes of the resources.

Each attribute describing the user and the user's problem is referenced by one or more rules, which tell how to compare that value to the corresponding resource attribute. Each rule, or rule set, represents a "chunk" of domain expertise that is communicable to the user and can be added to or deleted from the system's knowledge base. Rules accommodate the knowledge of domain experts in the form they most often communicate it: "If user has X attribute and/or problem has Y attribute, then action is to set Z attribute." These rules accommodate the knowledge of domain experts (resource managers) in the form they most often communicate it: "If user (object 1) has certain characteristics (attribute values A, B, C), and problem (object 2) has certain characteristics (attribute values C, D), then action is to assign resource(s) with these characteristics (attribute values a, b, c, d, e)." The assignment is accomplished through a pattern-matching algorithm which finds the resource(s) most likely to satisfy the user's needs.

The resource selection process is a type of problem solving by pattern matching. In classical pattern recognition, "the problem consists of recognizing class membership, and establishing decision criteria for measuring each class" [Van Gigch 1978]. Problem diagnosis utilizes pattern matching, in that a particular set of attributes is fitted in such a way that we can recognize how close the features of this pattern resemble those of another established pattern. Diagnosis differs from pattern recognition in that the former may involve classes whose attributes may not be completely specified at the time diagnosis (resource assignment) is attempted.

In the present case, the established pattern is the definition of the user and the user's current problem. We match our set of resources to that user-problem set to find the closest resemblance.

4.3 Knowledge Acquisition and Maintenance

A knowledge acquisition methodology for the Information Center Expert (ICE) was developed. It utilizes a series of four charts called Resource Attribute Charts (RAC) to (1) elicit from the expert the conclusions that are to be given by the expert system, (2) gather vocabulary, (3) identify relevant attributes of the resources, and (4) assign relative weights to those attributes. RAC is a methodology for elicitation, combination, and transfer of the expert's knowledge to data structures used by the expert system. It offers assistance in the knowledge engineering process by providing a structure for interviewing the expert, analyzing the information, and producing the resource profile for the knowledge base. RAC is based on Kelly's Repertory Grid process which was described in Chapter 2.

4.3.1 Initial Knowledge Elicitation

RAC-1 (Chart 1) shown in Figure 4.4 first elicits the conclusion items, called elements, from the experts. These elements represent resource conclusions that should be made by the expert system being built. Elements are elicited within major categories of resources.

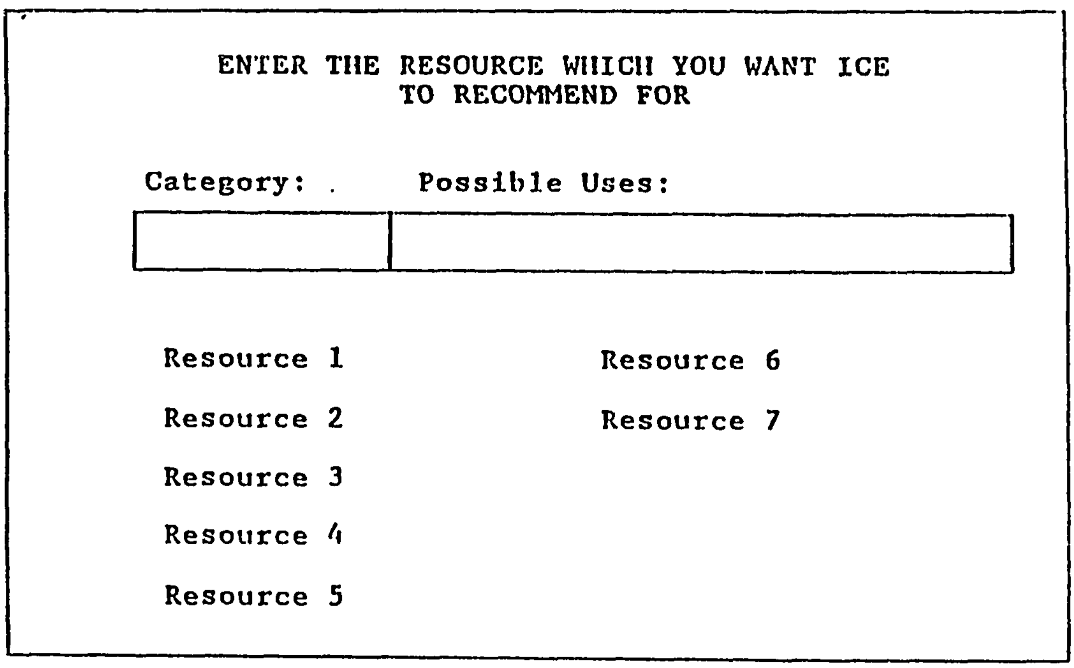

Figure 4.4: Resource Attribute Chart - 1 (RAC-1)

The expert is asked to list all the resources that fall in each category.

4.3.2 Elicitation of Attributes

Next, using RAC-2, the expert compares successive groups of three resources and names an important attribute or trait that distinguishes any two members of this triad from the third one (Figure 4.5). By comparing three elements at a time, the expert must think about both similarities and differences at once. If only two were compared, traits would be elicited that would be either similarities or differences. Comparing more than three at once makes the job much more complex without comparable increase in effective attribute identification [Kelly 1955].

The particular attributes identified depend upon which elements are combined into triads. It is recommended that the combinations are made in the order in which the resource objects were given by the expert on RAC-1, since they are probably the most significant or most crucial elements [Boose 1986a].

This step is performed iteratively until a list of resources within each desired category has been identified, and a list of classification attributes has been elicited for each category. These were all derived from the expert, using her/his own language and terms.

4.3.3 Attribute Weighting

Using RAC-3 (Figure 4.6) the expert determines the relative importance of each attribute in making the selection of the software resource package. This is done by using a rating scale based on Kelly's binary rating method, which has been extended here to include scales, allowing finer shades of distinction than "yes/no" or "on/off". At one extreme, the scale represents "crucial" attributes, and at the other extreme are those which are "not

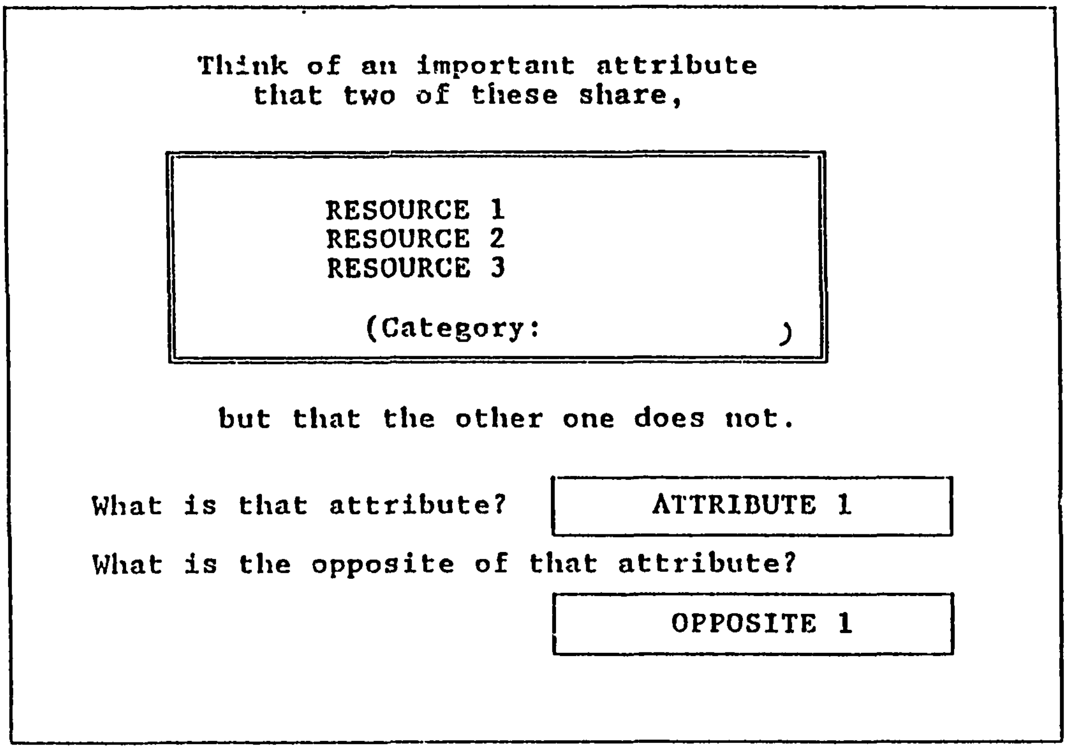

Figure 4.5: Resource Attribute Chart - 2 (RAC-2)

useful". Attribute weights are set from these rating scales.

4.3.4 Attribute Values

The last step of this methodology is a series of charts (RAC-4) used to assign values to all the attributes of each resource element. That is done by placing each element on the appropriate line of each attribute chart (Figure 4.7). The charts allow the experts to easily compare the resources relative to one another.

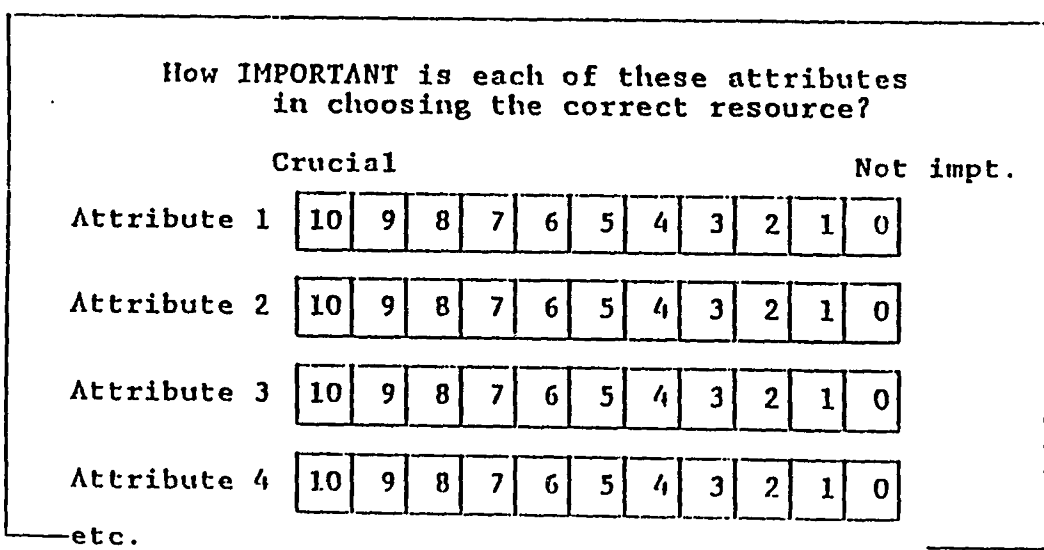

Figure 4.6: Resource Attribute Chart - 3 (RAC-3)

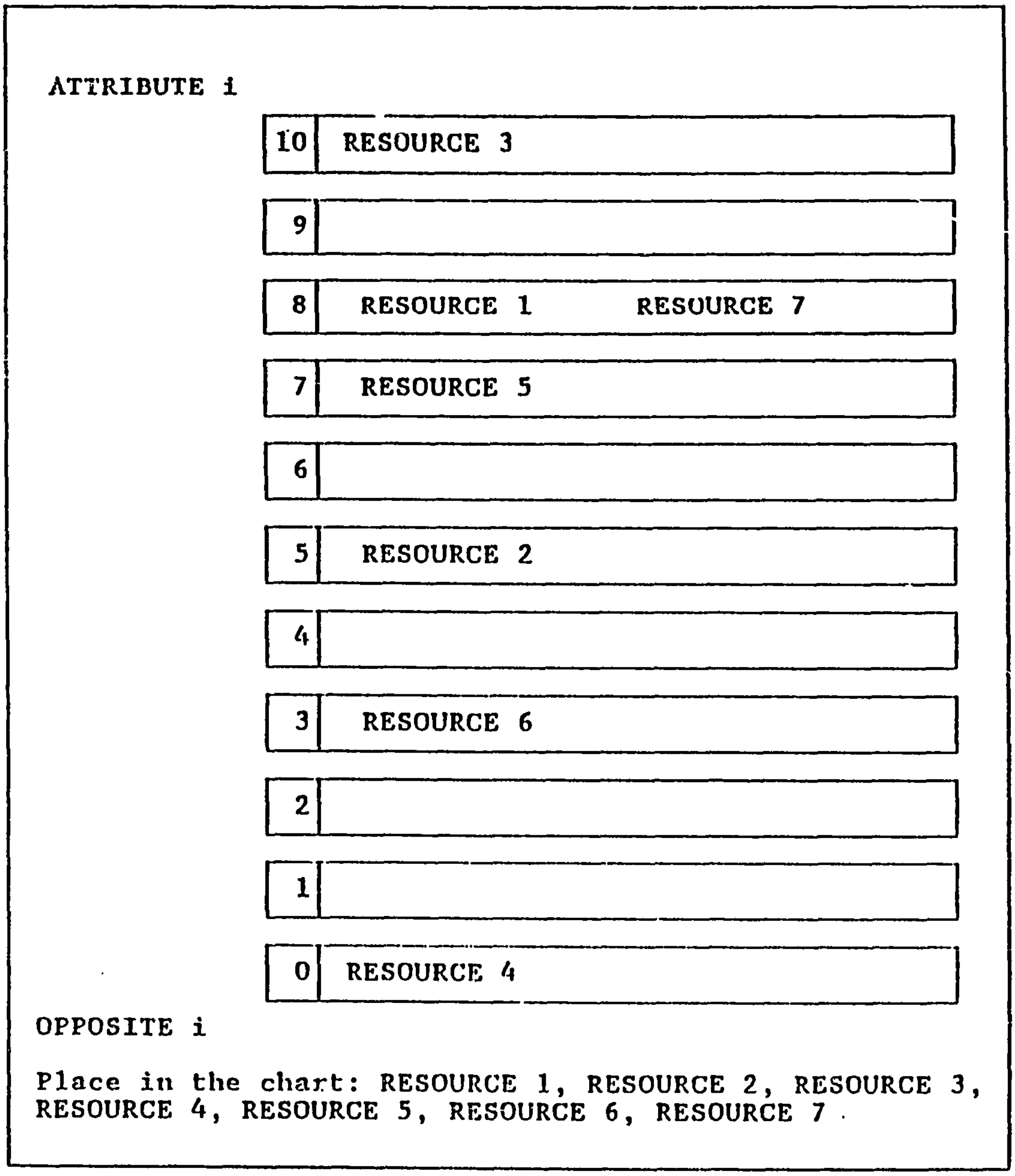

Figure 4.7: Resource Attribute Chart - 4 (RAC-4)

4.4 Summary

The research questions posed in this chapter dealt with (1) defining an appropriate architecture for time- and location-variant data, and (2) developing a useful knowledge acquisition methodology. The expert system architecture allowing for the representation of time– and location-variant data uses a simplified semantic network with object-attribute-value triplets. Each node in the network is a modified frame that defines the attributes of users, problems, and resources. The frames defining the resources are stored in external files to allow for easy creation and maintenance. The knowledge acquisition methodology, Resource Attribute Charts, allowed for the structured elicitation of the resource elements, attributes, attribute weightings, and attribute values.

Knowledge acquisition has been a long-standing problem in artificial intelligence work. The state of the art is "ill-defined and at a stage of experimentation and evolution" [Hart 1986, p. 9]. Often, in the building of expert systems, it has been the knowledge acquisition which was difficult, not the subsequent implementation. That was certainly true for this study. Development of a methodology to structure the process greatly improved the knowledge gathering process.

Chapter 5

CASE STUDY: IMPLEMENTATION OF INFORMATION CENTER EXPERT (ICE)

> To understand something as a specific instance of a more general case–which is what understanding a more fundamental principle or structure means–is to have learned not only a specific thing but also a model for understanding other things like it that one may encounter [Bruner 1960].

Information Center Expert (ICE) is a specific instance of the general architecture of the resource management system presented in Chapter 4. The third research question posed in Chapter 1 was:

> Can the use of an expert system facilitate the selection of resources in a dynamic organizational setting, specifically the setting of the Information Center?

This chapter addresses the question first by describing the actual implementation and installation of ICE in the information centers of two major corporations and in the Center for the Management of Information (CMI) at the University of Arizona, and then by testing the appropriateness of the resource selections mady by ICE in a validation study.

5.1 Installation Site Descriptions

The first corporation is supported by three information centers, each serving clientele with different needs. The Plant Information Center is housed in the Information Systems Department. Clients are end users who work in areas such as control, engineering, finance, and office management. They are usually not programmers. One of the main goals of this site is to make an improvement in the decision making processes of end users. The second center at this site is the Productivity Center, serving application programmers with the goal of looking at global needs of the community rather than needs of individual programmers. Work groups were the unit of attention rather than individual users. The Product Laboratory Center, the third support site, provides complete personal computer hardware and software support, including a PC store that offered inventory acquisition and the expertise on how to use it. The clientele are research and development personnel including managers, programmers, systems analysts, engineers, technicians, computer operators, and secretaries.

The second corporation is supported by an information center much like the Plant Information Center at the first site, serving a wide variety of end users and offering a broad selection of software packages.

The third site, the CMI at the University of Arizona, serves staff, faculty, and students of the College of Business and Public Administration, providing both hardware and software support to users with a variety of skill levels and needs for computer assistance.

5.2 Background

In previous chapters, the research problem was described at the most general level as the matching of users, problems, and resources within the constraints of organizational goals and policies. At the next level, users are specifically identified as end users with business problems requiring computing resources. These resources are assigned according to the policies and regulations of the organization, as enforced by the Information Center.

This chapter describes the specific instance in which the users can be identified as the personnel employed by the first corporation where ICE was installed. They possess specific problems and needs that must be met with computer software resources which will help them achieve their goals. The needs must be fulfilled within the constraints of the goals and policies that govern end user management within that corporation. The process of matching these users, problems and resources is done by building profiles: user profiles, problem profiles, and resource profiles. These profiles consist of attributes which must be weighted and valued by the expert system so that the matching process can take place. The process of identifying and valuing those attributes was described in the previous chapter.

In Figure 1.2, Hayes-Roth's conceptualization of the major components of a knowledge system were presented. They were divided into three environments: development, knowledge system, and operations. In this chapter, ICE is discussed in terms of these environments, as shown in Figure 5.1. Here, the relationships among the ICE subunits are shown. The details of the development and installation environment of ICE are given first, then the knowledge system is described in terms of the profiles built, and its operational environment is described. Finally, the recommendations given by ICE are validated through testing.

5.3 Development Environment

The Information Center Expert (ICE) was built on the IBM 4381 using the IBM shell ESE/VM (Expert System Environment /VM) [IBM 1986]. The development environment of this shell (ESDE/VM) allows for the building of a knowledge base of facts and

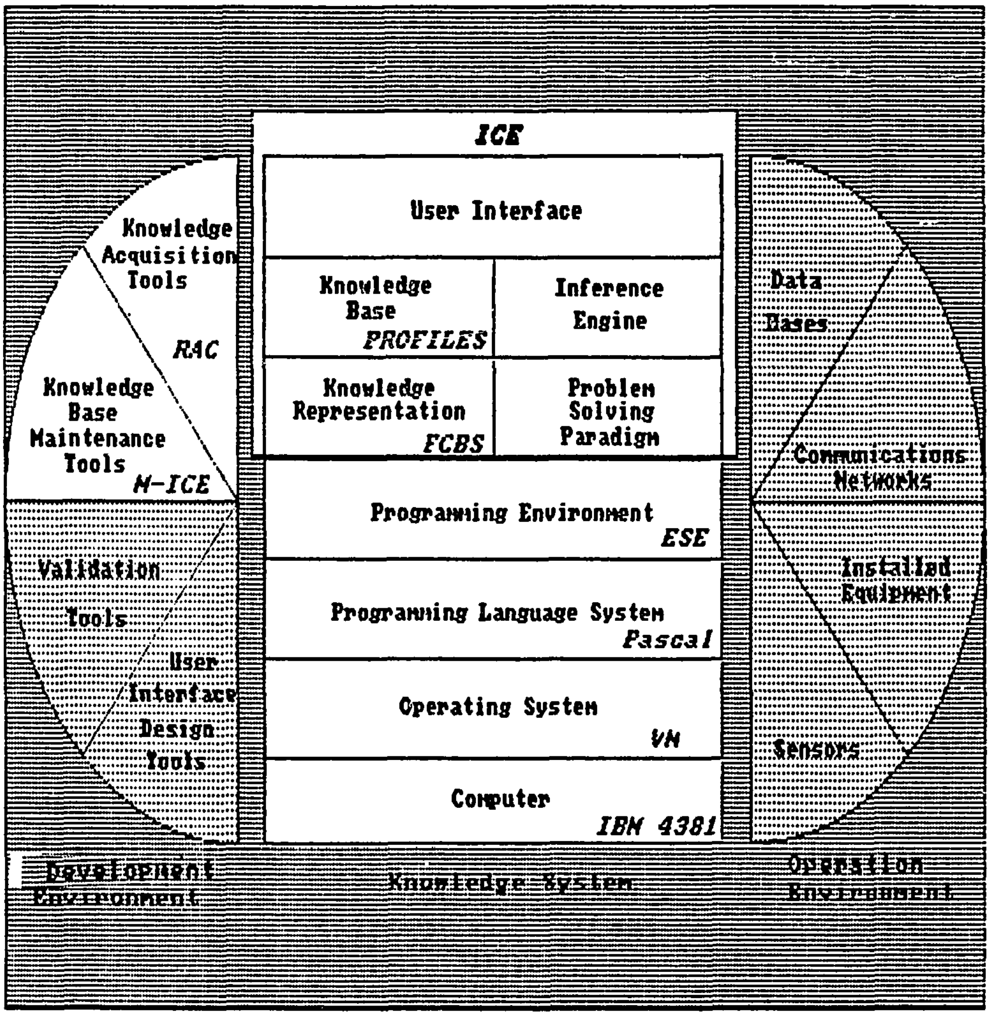

Figure 5.1: Major Components of ICE

relationships about users of the IC resources, current user needs, and definitions of the resources themselves. These facts, called "parameters" in the ESE shell, are grouped logically into profiles of the three parts of the model: the User Profile, the Problem Profile, and the Tool (Resource) Profile. Other facts about the relationships among these profiles are represented as "rules," and "Focus Control Blocks" (FCBs). Control of the parameters and rules is done through FCBs, which are organized as a hierarchy to provide structure to the system. Parameters, rules, and FCBs will be identified and explained in the Knowledge System Environment section.

Because of the flexibility and maintenance issues, part of the ICE knowledge base was stored in files external to the ESE environment, providing for easier update. These files are described in the Maintenace Tool section.

5.3.1 Overview of the ICE Architecture

Figure 5.2 shows the overall structure of the ICE system and its relationship to the external environment. End users approach ICE in need of recommendations for appropriate software for their particular tasks. Through dialogue controlled by the rule base built using the ESE-D (ESE Development) shell, information is collected about the end user's skill and experience (User Profile) and the user's immediate needs (Problem Profile). In the User Profile, the facts about the user are represented as parameters in the data structure called the EU_Vector (EndUser_Vector); the data structure for the Problem Profile is called the P_Vector (Problem_Vector). The relevant parameters from these two vectors are then grouped into a single vector called the PE_Vector (Problem_EndUser_Vector). This PE_Vector describes the skills, past experiences, and needs of the user for the current consultation.

The PE_Vector attribute values are stored in an external file. At the end of the dialogue, the values of this vector are matched against the attributes of the vectors that

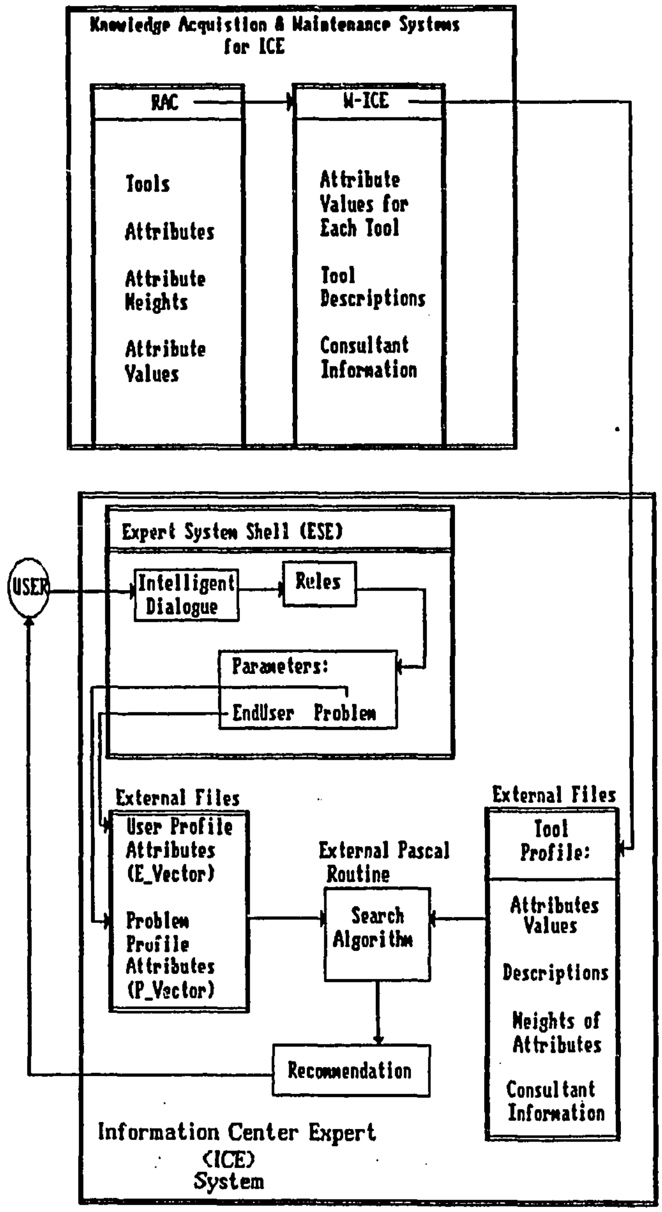

Figure 5.2: Overall Structure of ICE and Its Relationship to the Environment

represent each software tool supported by the Information Center (Tool Profile). The tool selection algorithm screens out inappropriate tools, and of those remaining, places confidence ratings on those whose vectors match the PE_Vector most closely. This recommendation is then given to the user along with information about the tools and the consultants who support those tools.

This section and the next (Knowledge System Environment) discuss in more detail how the content of each of these profiles was determined, how the knowledge was acquired for each, and how the three profiles are matched to determine a software recommendation.

5.3.2 Knowledge Acquisition for ICE

Knowledge for the ICE knowledge base was acquired through interviews, both unstructured and structured, by observation, and by example.

Knowledge of Users and Problems. Information about the consultation process, during which consultants collect data about the end user's background and current need for software, was gathered through extensive interviewing of the consultants as well as by observation of the process. This information was later represented as parameters and rules in User Profile and Problem Profile, discussed in the next major section of this chapter.

Knowledge of Resources. In order to adequately elicit knowledge to build the Tool Profile, an extensive search was conducted of organizational documents and manuals, and structured interviews were held with consultants. It was during the attempt to adequately represent the resource tools in the knowledge base that the need for acquisition and maintenance tools arose. Two tools were developed to support the construction of the ICE knowledge system: Resource Attribute Charts (RAC) for acquiring knowledge about the resources of the Information Center, and Maintenance tool for ICE (M-ICE) to maintain the knowledge base.

The knowledge acquisition charts (RAC) were presented in Chapter 4 as a structured methodology for eliciting knowledge about resources. This knowledge was of four types: elements, attribute identification, weighting of attributes, and assignment of values to attributes. IC consultants participated in the knowledge acquisition process. The data gathered from those structured interviews is now presented.

Elicitation of Elements. It had been determined from formal and informal interviews, and from a search of the literature (Chapter 2), that the major categories of software to be supported in the ICE system were those shown in Figure 13.

Data Management
Data Analysis
Presentation Graphics
Document and Report Preparation
Project Management
Utilities
Programming

Figure 5.3: Major Software Categories for ICE Modules

The experts listed all the software packages that fell in each category for their center. Figure 5.4 (RAC-1) shows the software that was listed for Data Analysis in one of the ICs. These packages are all the packages supported by that center, and given as resource solutions in the ICE system for data analysis needs. Because of nondisclosure agreements, the software is not listed in entirety. Many of the "internal use only" software tools are not revealed in this figure.

Attribute Identification. After identifying the software resources for each cate-

ENTER THE SOFTWARE PACKAGES WHICH YOU WANT
ICE
TO RECOMMEND FOR

Category:	Possible Uses:
Data Analysis	
VM/AS	
SAS	
Lotus 123	
Tiny Calc	
Planning Assistant	
Oxycalc	
APL2	

Figure 5.4: Elicitation of Elements: Software for Data Analysis Category

gory, the experts next used RAC-2 to compare successive groups of three packages and named an important attribute that distinguished any two from the third (see Figure 5.5). The output of this step was a list of classification attributes had been elicited for each category of software. These attributes form the Tool Vector file of the knowledge base. Appendix A contains a complete listing of the Tool Vector Attributes and their definitions, listed for each category of software. Next, priorities must be established among these attributes.

Weighting of Attributes. Next to be determined is the relative importance of each attribute in making the selection of the recommended software package for the user. The assignment of weights on the 1-10 scale is done according judgments of the IC consultants, and is based on their perceptions of which attributes are most important in choosing software tools for users. Each information center can determine its own set of weights, thus reflecting its own preferences on which attributes should contribute most heavily to the software selection. Figure 5.6 (RAC-3) shows how one of the consultants assigned weights to the attributes for data analysis tools.

Values for Attributes. The last step of RAC was to complete a series of charts (RAC-4), one for each software tool, to set the values of each attribute for each package (Figure 5.7). The values assigned must indicate how much better one package is than another with respect to that attribute. Again, each information center decides on its own ratings for the tools, so that its preferences in making recommendations can be simulated by ICE.

The output of each of these four steps (elements, attributes, weights, and attribute values) is entered into M-ICE, the automated maintenance program (described in the next section) that builds the four external files of the Tool (Resource) Profile. This profile is searched during each consultation to match the most appropriate tool to the user's needs and skills. This maintenance utility and the four files are now described.

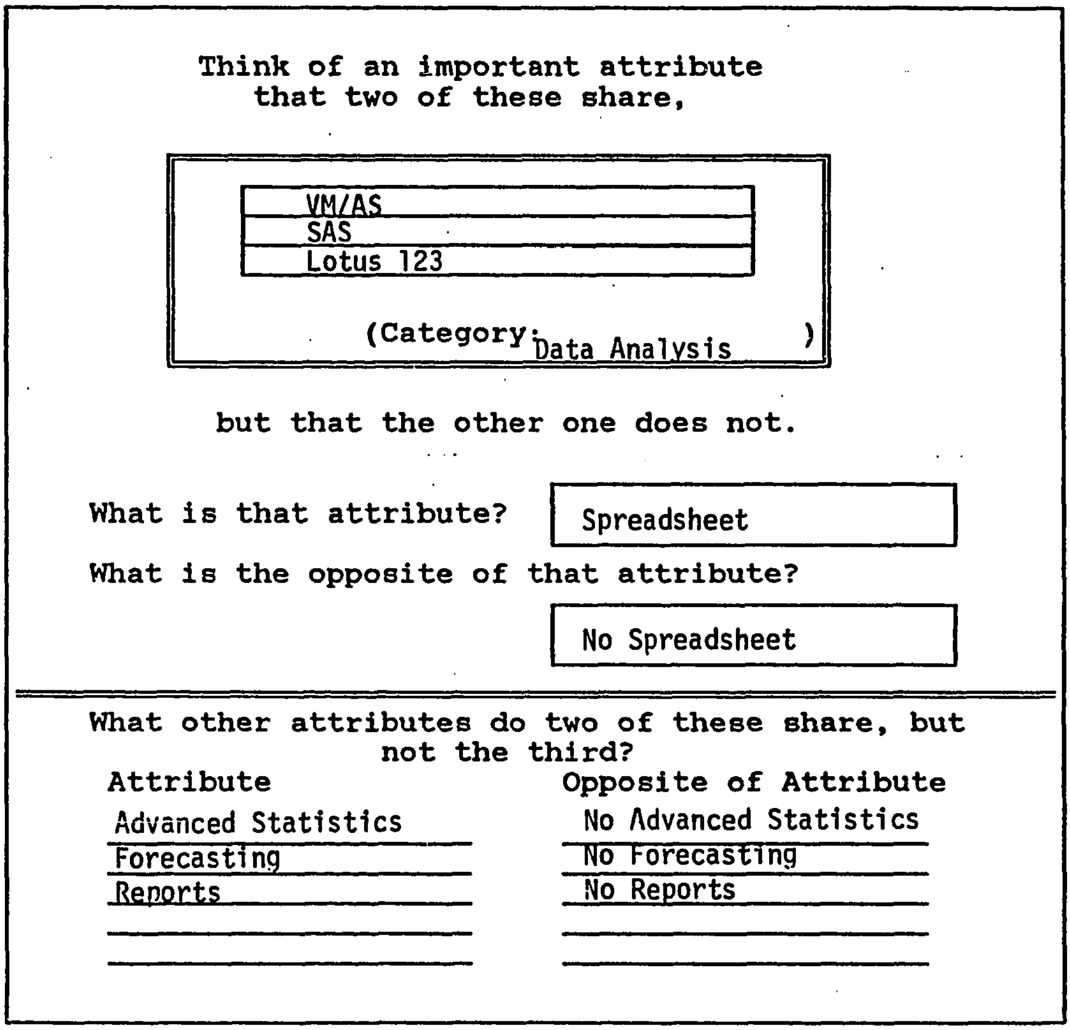

Figure 5.5: Software Attribute Identification (RAC – 2)

How IMPORTANT is each of these attributes in choosing the correct software?

Attribute	Crucial										Not impt.
Advanced Stat	10	9	(8)	7	6	5	4	3	2	1	0
Simple Stat	10	9	8	(7)	6	5	4	3	2	1	0
Spreadsheet	(10)	9	8	7	6	5	4	3	2	1	0
Macro	10	9	(8)	7	6	5	4	3	2	1	0
Graphs	10	9	8	7	6	(5)	4	3	2	1	0
Charts	10	9	8	7	6	(5)	4	3	2	1	0
Reports	10	9	8	7	6	(5)	4	3	2	1	0
Forecasting	10	9	8	7	6	5	4	(3)	2	1	0
Financial	10	9	8	7	6	5	4	(3)	2	1	0
________	10	9	8	7	6	5	4	3	2	1	0
________	10	9	8	7	6	5	4	3	2	1	0
________	10	9	8	7	6	5	4	3	2	1	0
________	10	9	8	7	6	5	4	3	2	1	0
________	10	9	8	7	6	5	4	3	2	1	0

Figure 5.6: Attribute Weighting for Software Selection (RAC – 3)

ATTRIBUTE: Reports

10	SAS VM/AS
9	APL2
8	Planning Assistant
7	
6	Lotus 123
5	
4	
3	
2	Oxycalc Tiny Calc
1	
0	

Opposite of Attribute: No Reports

Place in the chart all software packages from Page 1 of this category.

Figure 5.7: Assignment of Attribute Values to Rate Software Resources (RAC – 4)

5.3.3 Maintenance Tool

M-ICE (Maintenance for ICE) is a PC-based subsystem used primarily to develop and maintain the Resource Profile part of the ICE knowledge base. The profile consists of the four files generated by M-ICE. The contents of these file are accessed by the selection algorithm during the matching process. The files are TOOLFILE.DAT, DESCRIBE.DAT, CONSULT.DAT, WEIGHTS.DAT.

The TOOLFILE.DAT file consists of software tool names and attribute values, in terms of their ability to perform the activities required by the users. The information is gather from RAC-1 and RAC-4 and entered using M-ICE.

DESCRIBE.DAT is a file that allows for 10-line maximum narrative descriptions of the capabilities of each software tool. It is used by ICE after the tool recommendation has been given to the user, and the user elects to find out more about the software.

WEIGHTS.DAT holds all the attributes (and their definitions) elicited from RAC-2, and elicits the assignment of the weights to each attribute. The weights tell the relative importance of the attribute in selecting the resource. The weight assigned is a reflection of the preference of the consultant deciding on the importance of each attribute. The process of weighting attributes is subjective, and for the consultation system to be transportable across ICs, it is necessary to allow the introduction of such subjectivity by each individual IC site that implements the ICE system. This is particulary true because the aim of the IC is not to replace, but to enhance the interaction between end users and consultants. Thus, it is important that the system be a close reflection of the preferences of the specific IC it is supporting.

The file CONSULT.DAT holds the names and phone numbers of the Inforamtion Center consultants. These names can be accessed at the time that ICE makes the software recommendations. Users are encouraged to contact the support personnel of the software selected for them.

M-ICE allows for the easy addition, deletion, and modification of software tools and consultants represented in the knowledge base. It is a very important development tool in the ICE system because it allows maintenance to be performed by the IC personnel through a menu-driven interface. As new software tools are supported by the IC, they can easily be added to the knowledge base without revision of the rules. Tools can be redefined or deleted as IC policy dictates. Names and phone numbers of consultants who support those tools are changed in a similar fashion. M-ICE allows each IC to represent its own resource set without requiring the services of a knowledge engineer to change the rule base.

M-ICE also provides a reporting facility making possible a formatted report of any of the following: (1) All software packages supported by the Information Center, complete with descriptions, (2) the attributes currently being used to describe software categories, (3) information about consultants who support end user activities. Appendix B contains extracts of reports produced by the M-ICE utility.

5.4 Knowledge System Environment

Knowledge about (1) end users, (2) their business problems, and (3) the resources available to them was represented in the knowledge base of the expert system ICE by building profiles of each of these components of the model. Three subsystems perform this process: (1) the User Subsystem builds the User Profile, (2) the Problem Subsystem builds the Problem Profile, and (3) the Resource Subsystem builds the Resource (Tool) Profile.

5.4.1 User Subsystem

The User Profile is gathered through ICE dialogue with the user. First it is determined whether this is the user's first session by trying to match the user's serial number with those stored from previous consultations. If the user has consulted with ICE before, the existing User Profile is displayed, and the user verifies that the data is current, or makes desired changes.

If, however, this is the user's first consultation, the user is engaged in a dialogue to collect background information concerning previous computer experience, skill, and computer hardware availability. This information is collected by a series of questions controlled by a forward chaining mechanism, and stored as attributes (parameters) that form the User Profile, known to the system as E_Vector (EndUser_Vector). This User Profile consists of both the static and dynamic characteristics of users, as shown in Figure 5.8.

5.4.2 Problem Subsystem

After the User Profile has been constructed, a backward chaining inference engine fires rules that carry on further dialogue with the user to determine the characteristics of the immediate problem or activity for which s/he seeks computing resources to help reach a solution.

The category of the user's task is first determined, and control is passed to the appropriate module. The modules built are for those software categories found to be most commonly used by end users: data management, data analysis, presentation graphics, document and report preparation, project management, utility and productivity aids, and programming tools.

1. Static:
    - User Name
    - Identification Number
2. Dynamic:
    - Department Number
    - Phone Number
    - Computer Tasks - description of the tasks which the user most often performs on the computer.
    - Hardware Available - identification of computer hardware which user has available in his/her environment.
    - Operating System Preference - the OS in which the user prefers to work.
    - Skill Level - user's rating of own skills in chosen computer environment.
    - Software Experience - major categories of software with which user has previous experience.
    - Computer Usage - how often person uses a computer to perform job; importance of computer in being able to meet goals.

Figure 5.8: User Profile Attributes: Static and Dynamic Characteristics

The data structure that contains the attributes of the Problem Profile is known to the system as the P_Vector (Problem_Vector). The system stores the attributes by assigning values to the parameters of the Problem Vector. The relevant parameters from the EndUser_Vector are then concatenated to the Problem_Vector, forming a single vector known to the system as the PE_Vector (Problem_EndUser _Vector). This vector then contains attribute values that define the skills, needs, and problem environment of the user for the current consultation.

The attributes of the PE_Vector are then matched against the vectors of attributes in the Resource (Tool) Profile which represent each software tool supported by the Information Center.

5.4.3 Resource Subsystem

The Resource Profile, also referred to as the Tool Profile, characterizes information center resources in such a way that they can be matched with the user skills and needs (Problem and EndUser Profiles). The resources (elements) and the attributes which describe them were acquired using Resource Attribute Charts (RAC) and the maintenance utility M-ICE as described in the development section.

The attributes of each resource are stored in parameters of the Tool Vectors, one vector for each software tool. A selection algorithm matches the attributes of the Problem_EndUser_Vector and the Tool Vectors to determine which tools most closely meet the user's skills and requirements. Figure 5.9 shows conceptually the process of matching the vectors that represent the profiles.

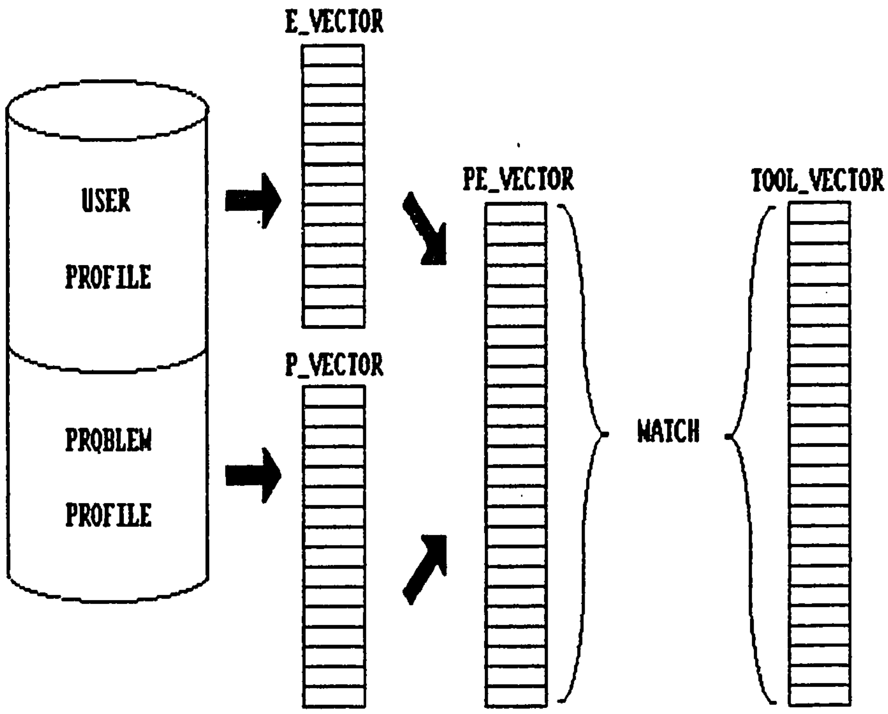

Figure 5.9: Conceptual Matching of User and Problem Attributes With Resource (Tool) Attributes

5.4.4 Focus Control Blocks

Each of the activities involved in acquiring profile information is controlled within its own module called a Focus Control Block (FCB). Figure 5.10 shows the hierarchical structure of the FCBs for the ICE system. The dialogue for a consultation session follows this structure. The user is asked for an identification number. From this number it is determined if the user is making a consultation for the first time, or has previously consulted with ICE. If no previous consultation has been made, user information is gathered to build the User Profile.

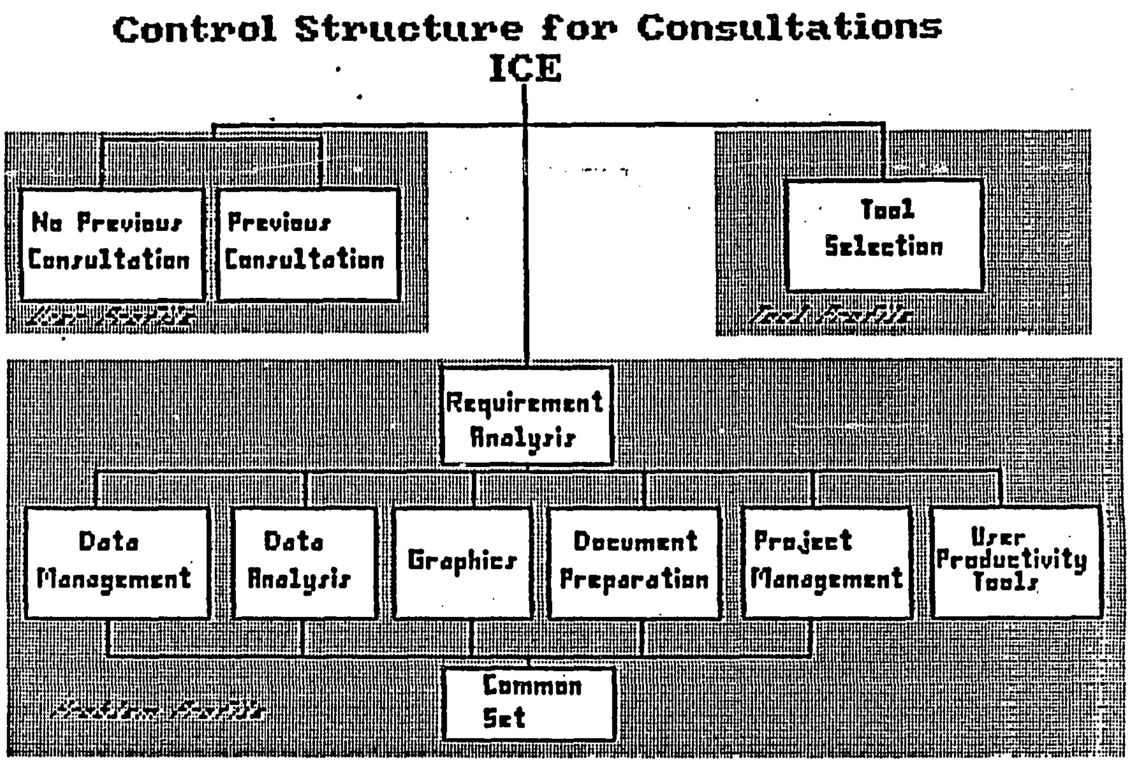

Figure 5.10: Focus Control Blocks

Control is then passed to the Problem Profiling Subsystem controlled by the Requirement Analysis FCB. The user's problem is categorized into one of six categories, and control is passed to the FCB for that category. More detailed information is then gathered about the specific needs within that category. Finally, general questions are asked as controlled by the Common Set FCB. These are questions whose answers are needed for almost all consultations.

Once the user's background and problem are determined, control is passed to the Tool Selection FCB, and the matching operation is performed to determine the recommended set of software.

5.5 Operation Environment

ICE currently is a closed system that interacts only with files created specifically for this program. While it is possible for ICE to access personnel data bases to acquire the User Profile information automatically, the corporation in which the system was installed chose not to allow that interaction. The security level of the personnel files is too high.

IC managers can opt to have ICE produce tracking files which save information about the users of the ICE system. A tracking report gives the users' names, departments, phone numbers, category of software sought, and recommendation made by ICE. Statistics can be generated on how many users had needs that could not be met, those whose needs were partially met, and those who received software recommendations from ICE. These reports can then be used to contact users about further needs or satisfaction level of the recommendation.

When ICE is extended to support the training activities of the IC, it will also interact with the application software packages that it recommends. Users will have the option of experimenting with the software that ICE suggested, and external files will be called from ICE to bring the user into the application programming environment. If the application is a PC package, ICE should allow the users to download the package to their own workstations. These operational features are topics for future research.

5.6 Validation of ICE

Confirmation of the validity of ICE was sought by comparing the results of its operation against the resource recommendations of IC consultants. The procedure used is called "blind" validation and is based upon the work of A. M. Turing [1950] who, when confronted with the issue of whether or not computers and their programs could think, said the testable question is whether a blind observer–that is, one ignorant or blind as to the identity of the source– could distinguish between the product of the machine and the product of the human.

The blind validation of ICE was conducted. The IC manager, serving as validator, was confronted with software resource recommendations for the twenty cases shown in Appendix C. The recommendations came from two different sources, human consultants from the Information Center, and the ICE system. The validator studied evidence that was available to the sources, formed his own conclusion, and then accepted or rejected each recommendation from each source on the basis of his opinion of the appropriateness of the recommendation given. The identity of the source of each recommendation was not made known to the validator until after each had been judged.

The first source–the human consultants–consisted of two IC consultants, one who specialized in personal computer (PC) support, and the other who supported mainframe (MF) computer problems. The PC consultant solved the PC cases, and the MF consultant solved the mainframe cases. The second source consisted of four end users who served as users of the ICE system. The result given by each source (ICE or human consultant) was reported to the validator in random sequence to disguise the source. In ten of the twenty cases (randomly chosen by a BASIC random number generation program), ICE's recommendation was identified as "Source I" and in ten cases as "Source II."

5.6.1 Validator Compares Recommendations

The researcher typed the material of the twenty cases on Case Sheets. Each sheet was accompanied by the Recommendation Sheet on which the recommendations were given for the cases of the two sources, the IC human consultant, and the ICE system. The sources were disguised as to their source, human or machine. Certainty percentages given by ICE were not reported on the Recommendation Sheets, as responses of the human consultants are not often expressed in that form. To include the percentages would have given notice to the validator that ICE was the source of the judgement.

The validator's task was to make an independent judgement as to the correct recommendation. Then, based on that assessment, he was to evaluate the correctness of Source I and Source II on each case by accepting or rejecting each recommendation. The results of this evaluation are summarized in Table 5.1. The details are presented in Table 5.2. In the last column, "A" designates "Accept" and "R" stands for "Reject."

5.6.2 Results

In 17 of the 20 cases, the recommendations of ICE and the consultants were the same, or close enough to be in agreement with the validator. In two cases (#7 and #19) the human consultant was considered correct, but not ICE; in one case (#6), ICE was judged correct but not the human consultant.

A chi-squared test measured the hypotheses for the data summarized in Table 3:

HO:Reaching a valid conclusion is independent of the source of the recommendation.
H1:Source of recommendation and the number of correct recommendations are not independent.

This test suggests that the correct solution is independent of the source of the recommendation, as we fail to reject the null hypothesis. This gives evidence that ICE system provides resource recommendations that are not significantly different from those provided by human Information Center consultants.

Table 5.1: Data for Measuring the Proportion of Correct Recommendations

	Human Consultant	ICE System
Number of Problems	20	20
Number Correct	19	18
Number Incorrect	1	2
Proportion Correct	95%	90%
chi-squared = 0.3603	df = 1	p = 0.45

5.7 Summary

The installation of ICE at a corporate site realized the third research goal: to implement and validate a working resource management system based on the generalized architecture presented in the previous chapter. The system's facility to correctly select software resources for users was demonstrated by the validity experiment. The recommendations of ICE were not significantly different from those of the consultants of the Information Center, supporting the claim that the system can facilitate resource selection in a dynamic environment.

Table 5.2: Validation Data

Case #	ICE: Source Number	Type of Case	Description of Case	Recommendations: Human /	Validation Decision by ICE	Validator
1	II	PC	Data base mgmt/ 1000 inventory records	PC-File+		A
					PC-File3	A
2	I	MF	Data base mgmt/ 2500 inventory records	VMAS		A
					VMAS	A
3	II	MF	Data base mgmt/ 7500 inventory records	VMAS or SQL		A
					VMAS/SQL	A
4	II	PC	Business planning with spreadsheet	Lotus123		A
					Lotus 123 or TinyCalc	A
5	II	PC	Report prep/ with spreadsheet	TinyCalc or Lotus		A
					Lotus or TinyCalc	A
6	I	MF	Complex statistical analysis	VMAS or Lotus		R
					SAS	A
7	II	PC	Budget preparation	Lotus		A
					VMAS	R
8	II	MF	Manipulation and analyses of matrices of data	VMAS		A
					SAS or VMAS	A
9	I	PC	report generation with graphs	Graphing Asst		A
					PC-File or Graph Asst	A
10	I	PC	Presentation material-transparencies	ChartMaster or SignMaster		A
					ChartMaster or SignMaster	A

Table 5.3: Validation Data

11	I	MF	Presentation material-	APGS		A
			transparencies		APGS	A
12	II	PC	Paper graphics with alphanumeric text	SignMaster or PersExpr		
					SignMaster or ChartMstr	A A
13	II	MF	Graphics: fonts,	APGS		A
			color, Xter sizes		APGS	A
14	I	MF	Graphics: library	APGS		A
			predefined symbols		APGS	A
15	II	MF	Document Preparation:	PROFS		A
			columns, spell, merge, embed graphics		PROFS	A
16	I	PC	Document Preparation	DW3/WrdPrf		A
					DW3/DWAsst /WrdPrf	A
17	I	PC	Word Processing & Editing - table of	DW3/ProfEd		A
			contents, footnotes, spell check		DW3/ProfEd/ WrdPrf	A
18	II	PC	Simple memo and document preparation	WrdPrf/WrtgAsst		A
					WrtgAst/WrdPrf	A
19	I	PC	Project management -	HTPM		A
			critical path		VMAS	R
20	I	MF	Project management -	VMAS		A
			critical path		VMAS	A

Chapter 6

TRANSPORTABILITY STUDY

This chapter discusses the results of the experiment testing the expert system's success in meeting the design criteria of transportability.

6.1 Background

It should be recalled that two charactertistics of knowledge important in expert systems development are context dependency and knowledge stability [Krcmar 1984]. Context dependency describes how universal certain knowledge is, and stability is the change over time of the knowledge represented. These criteria for universality and stability have made it difficult in the past to apply expert systems techniques to the Information Center because the knowledge base of IC resources is neither universal nor stable.

Software tools are being introduced into the market at an extremely rapid rate, offering more power and more ease of use. ICs are continually changing the set of available software tools, causing a very dynamic environment that changes quickly over time. The demand for universality is not easily met either, for each IC supports its own unique set of software resources. Often within the same corporation, different branches or sectors prefer different software to solve the same problem. Thus, the challenge in applying the power of expert systems to consultation situations within the IC required that an architecture be designed that could allow for a knowledge base that could be easily changed as the resources changed, and could be changed as required when installed in many different ICs, each with a different resource base. This architecture was the subject of Chapter 5.

Most rule-based expert systems model problem solutions directly into the rules; that is, using IF/THEN statements, recommendations are "hard-coded" into the response portion of the rule. This method, however, is not acceptable in the IC, for it would require the assistance of a knowledge engineer to rewrite and recode the rules with the adoption of each new or different software package. Maintaining the knowledge base so that it is current would be a very difficult and expensive task.

Likewise, the knowledge base must be transportable to different ICs without requiring the intervention of the knowledge engineer. The expert system must rest on an architecture that allows the flexibility for each information center to individualize the system to meet site- specific needs. What is necessary is the division of knowledge into a general core and a local core. The general core consists of knowledge that is common across all ICs; knowledge that is specific to individual sites will go into the local core. An expert system should be adaptable enough to be implemented in different ICs without changing the basic rule structures.

6.2 The Research Question

The research question to test the transportability of the knowledge system between information center sites is:

> Do the knowledge acquisition and maintenance tools, RAC and M-ICE, provide for the adaptation of ICE to different information center locations without changes to the rule base?

The goal is to determine whether the knowledge acquisition and maintenance methodologies (RAC and M-ICE) make possible the transport of the knowledge base from one location to another, allowing each site to identify and define its own set of resources, without changes to the rule structures. If the system can be transported without rule changes, the results support the claim that the knowledge has been successfully divided into a general core, the rules that are common to all information center sites, and an adaptable local core, the definition of the resources unique to each site.

Thus, it would be demonstrated that the expert system can operate in an environment with location-dependent and time-dependent knowledge instead of requiring universal and stable knowledge.

6.3 Description of the Experiment

ICE was installed and tested at the first corporate Information Center in February, 1987. After successful implementation at that site, the system was installed in May at a second corporate site in a different part of the country. The term "original knowledge base" refers to the knowledge base as it was installed at the first IC. The term "adjusted knowledge base" refers to the knowledge base as it was adjusted, using RAC and M-ICE tools, to model the IC resources at the second site. Inspection of the list of software tools supported at the two sites revealed that just over half of them were the same. In some cases, even when the same tool was supported, the conditions of its use differed, so it would not be recommended for the same problems in the two sites. Therefore, its attribute representation in the knowledge base would be different at the two sites.

Participants in the study consisted of 24 users of the ICE system (business people, and students, faculty, and staff of the University of Arizona College of Business and Public Administration), and one "expert" IC consultant (the manager) who judged the results. The ideal participants would have been users from the second site where ICE was installed. Because permission could not be granted to work with anyone other than the IC staff, users from the university site were chosen. It can be argued that the population from which the participants was drawn is very similar to the population of users at the corporate site. The kinds of software resources they need, the computer hardware on which they work, and the problems they need to solve with those resources, are much alike. The users were asked to read sample cases (Appendix C), and assume the user and problem characteristics given by the case. The cases had been judged by the IC manager as representative of all the kinds of activities for which end users need computer support.

The participant users had varying levels of skill and frequencies of computer usage. All had, at a minimum, used the computer for simple word processing or programming, and most had also used the computer as a tool to perform simple activities such as inventory and records management. No effort was made to compare results between user groups, as the intent was to look at the performance of the ICE knowledge base in two different states, its original state as it was built for the first installation, and its adjusted state, changed to meet requirements of the second site.

Using a random number generation program, each of the users was assigned five cases to solve using ICE. Thus, ICE solved each case six times, each time with a different user. Each user first did the same sample case to become familiar with the ICE consultation process. Users were asked to read carefully each of their five cases and to interact with the dialogue of ICE to get a software recommendations for the cases. After each consultation, the entire consultation process was saved in a "store file." The concept of the store file, and the reason for its use, are now explained.

The expert system development shell with which ICE was built (ESE) offers many of the dialogue and explanation facilities common to development environments. During each consultation with a user, ESE stores the answers to the dialogue in a temporary Answer File. The capability exists to save that Answer File beyond the current consultation period. The answers to the consultation can be transferred to a permanent Store File so that the exact same dialogue can be rerun at a later time. Therefore, when a single Store File is rerun, there is no variability in the dialogue or in any of the answers given by the user. If the same knowledge base is used when the Store File is rerun, the expert system solution will be exactly the same. In this experiment, the dialogue with the user was collecting the attributes of users and their problems–the user and problem profiles. Thus, it was these two profiles that were controlled. They were exactly the same when the consultation was done using the original knowledge base as when using the adjusted knowledge base.

The 24 participants used ICE to solve each of their five cases. Each of the 120 consultations was stored in a separate Store File. The recommendations that were made by ICE in each of the 120 sessions was recorded. The knowledge base against which the solutions were drawn was the original KB that represented the software tools at the first site installation. These ICE solutions were then examined by the IC manager at the second site, and were judged "correct" or "incorrect", depending on whether they matched the software solutions as dictated by policy at Site 2. The results of that judgment are given in the next section.

The tools RAC and M-ICE were then used by IC personnel at the second site to adjust the knowledge base so it modelled their resources. No rule changes were made. After that adjustment, the 120 stored consultations were again run again on ICE, this time against the adjusted knowledge base. ICE's recommendations were recorded, and the IC manager again judged the solutions as "correct" or "incorrect."

6.4 Transportability Test Results

Since the IC at the second site supported a set of software resources in which approximately half the tools differed from those of the first site, it was expected that at least half of the software packages recommended on the first run (the original KB) would not be correct according to the policy of the second IC site.

A chi-squared statistic was used to test the hypothesis:

H0:The proportion of cases correctly solved by ICE at the second site is the same when the the original (unadjusted) knowledge base is used as when the adjusted knowledge base is used.
H1:The proportion of cases correctly solved by ICE is different after the knowledge base is adjusted.

Table 6.1: Data for Measuring the Proportion of Correct Solutions

	Original Knowledge Base	Adjusted Knowledge Base
Total Number of Cases	120	120
Total Number Correct	59	104
Total Number Incorrent	61	16
Proportion Correct	0.492	0.867
chi-squared = 38.722	df = 1	p< .005

Test results are given in Table 6.1. The statistic is significant at less than 0.005. The conclusion can be drawn that the proportion of cases correctly solved is different after the knowledge base is adjusted, providing evidence that RAC and M-ICE tools were effective in properly adjusting the knowledge base without changing the rule base.

The previous statistic assumes independent data. Since the data does not strictly conform to the independence assumptions (users did more than one case, and cases were repeated), a Cochran's Q-test was also calculated, giving similar results [Table 6.2]:

Ignoring the cases showing + + (both KB states gave correct answer) or - - (neither KB state gave correct answer), which give no clue as to which state of the knowledge base is superior, we are left with 45 cases. If the original knowledge base produces the

Table 6.2: Data for Calculating Cochran's Q-Test

		Original KB		Total
Adjusted	+	59	45	104
KB	-	0	16	16
		59	61	120
		Chi-squared = 45	df = 1	p < .005

chi-squared adjusted for continuity = 43.022

recommendations as accurately as the adjusted knowledge base, the number of cases in which the original KB is positive (correct) and the adjusted KB is negative (incorrect) should be equal to those in which the original KB is negative (incorrect) and the adjusted KB is positive (correct).

The data is presented in Table 6-2. The chi-square of 45 again supports rejecting the null hypothesis in favor of the alternative hypothesis, giving evidence that the RAC and M-ICE tools were effective in properly adjusting the knowledge base without changing the rule base.

Next, separate analysis was done of each of the twenty cases. Close inspection of the expert's judgement of the correct recommendation for each case reveals that for cases numbered 2, 4, 6, 8, 11, 13, 14, 15, and 20, the second IC recommends the same software tool as the first IC, so the two states of the knowledge base should give equally accurate results.

For cases numbered 1, 3, 5, 7, 9, 10, 12, 16, 17, 18, and 19 the software tools recommended in the first IC are not the same as those recommended in the second IC. Therefore, the recommendations given by ICE from the original knowledge base should not be the same as from the adjusted knowledge base.

The research hypotheses are:

H0: Reaching a valid conclusion is independent of which KB was used, the original KB or the KB adjusted using RAC and M-ICE.
H1:The state of the knowledge base (original or adjusted) and the number of correct conclusions are not independent.

The results of computing the chi-square for each case are given in Table 6.3.

Conclusions drawn from those cases where both IC sites recommend the same software tool are as follows:

Case 2: Fail to reject H0
Case 4: Fail to reject H0
Case 6: Fail to reject H0
Case 8: Fail to reject H0
Case 11: Fail to reject H0
Case 13: Fail to reject H0
Case 14: Fail to reject H0
Case 15: Fail to reject H0
Case 20: Fail to reject H0

These results are as expected.

Conclusions drawn from the results of those cases in which the IC sites recommended a different set of software resources follow:

Case 1: Reject H0

One user of ICE incorrectly analyzed the case as needing a spreadsheet package to handle inventory records. Therefore, ICE produced the wrong recommendation in both the original and adjusted knowledge base for that user. In the other five replications of that case, the original KB produced the wrong answer, the adjusted KB produced the correct answer.

Case 3: Reject H0

Site 1 recommended SQL as its primary tool for this case. Site 2 recommended VM/AS. The adjusted ICE KB made the correct recommendation for Site 2 in all six repetitions

Table 6.3: Chi-square, Measuring the Independence of Correct Recommendations and State of KB

		Original KB	Adjusted KB	Chi-square statistic
Case 1:	Correct	0	5	
	Incorrect	6	1	$(p < 0.005)$
Case 2:	Correct	6	6	0
	Incorrect	0	0	
Case 3:	Correct	0	6	12.0
	Incorrect	6	0	$(p < 0.005)$
Case 4:	Correct	6	6	0
	Incorrect	0	0	
Case 5:	Correct	3	6	4.0
	Incorrect	3	0	$(p = 0.046)$
Case 6:	Correct	5	5	0
	Incorrect	1	1	
Case 7:	Correct	2	4	1.33 *
	Incorrect	4	2	
Case 8:	Correct	6	6	0
	Incorrect	0	0	
Case 9:	Correct	0	2	2.4 *
	Incorrect	6	4	
Case 10:	Correct	0	6	12.0
	Incorrect	6	0	$(p < 0.005)$
Case 11:	Correct	4	4	0
	Incorrect	2	2	
Case 12:	Correct	0	5	8.571
	Incorrect	6	1	$(p < 0.005)$
Case 13:	Correct	5	6	1.091
	Incorrect	1	0	
Case 14:	Correct	6	6	0
	Incorrect	0	0	
Case 15:	Correct	6	6	0
	Incorrect	0	0	
Case 16:	Correct	1	3	1.5 *
	Incorrect	5	3	
Case 17:	Correct	3	6	4.0
	Incorrect	3	0	$(p = 0.0466)$
Case 18:	Correct	0	5	8.571
	Incorrect	6	1	$(p < 0.005)$
Case 19:	Correct	0	5	8.571
	Incorrect	6	1	$(p < 0.005)$
Case 20:	Correct	5	6	1.091
	Incorrect	1	0	

of the case.

Case 5: Reject H0

For this case, Site 1 recommended the use of either Lotus 1-2-3 or Planning Assistant. Site 2 recommended the user of either Lotus 1-2-3 or Tiny Calc. Therefore, the orginial recommendations were judged as being 50% correct, and the adjusted recommendations judged as being 100% correct.

Case 7: Fail to reject H0 at .05 level

In this case, the correct recommendations were judged to be Lotus 1-2-3 and Tiny Calc. Two of the users incorrectly specified in the dialogue that they did not need a spreadsheet, and indicated that they wanted to work in both the PC and mainframe environment. Therefore, both the pre- and post-test ICE sessions produced the wrong recommendations. Of the other four, ICE produced two correct recommendations from the original KB and all four correct for the adjusted KB.

Case 9: Fail to reject H0 at 0.05 level

Case 9 is an example of one time when the software tool added for the adjusted knowledge base was not done correctly. Site 2 specified that Graphing- Assistant be recommended for situations such as this, but it was recommended as a first choice only twice and as a second choice once. The attribute values for this tool must be adjusted again to allow it to be chosen by the search algorithm when relevant. Graphing- Assistant was not ever recommended in the original KB at Site 1.

Case 10: Reject H0

Site 1 recommended Storyboard as its primary tool. Site 2 recommended the ChartMaster/SignMaster series. The adjusted knowledge base reflected this difference by giving the correct answer in all six repetitions of the case.

Case 12: Reject H0

One user emphasized the need to analyze data more strongly than the need to produce presentation materials, as stated in the case. Therefore, the wrong package was recommended for both the original as well as the adjusted knowledge base. For the five remaining repetitions of this case, the original KB produced no correct recommendations, while the adjusted KB produced all five correct.

Case 16: Fail to reject H0 at 0.05 level

Analysis of this case reveals that the dialogue should perhaps be changed for one of the questions asked of ICE users. Three of the six users responded affirmatively when asked if they were using the "Assistant Series" of software, yet nothing in the case indicated that they should do so. Much weight was given to this answer, so ICE in both the pre- and post- runs recommended a word project package associated with the Assistant Series for those three users. The recommendation judged correct by the "expert" was DisplayWrite III with WordProof. In the original knowledge base of ICE, DisplayWrite III was recommended for the other three users, but not WordProof. For the adjusted KB, the other three users were all given the correct recommendation.

Case 17: Reject H0

The "expert" judged that the correct recommendations were either DisplayWrite III or Professional Editor, either in combination with WordProof spelling checker. For the original knowledge base, WordProof was never recommended, so 50% of the answers were judged correct. For the adjusted KB, all six recommendations were correct.

Case 18: Reject H0

Site 1 recommended DW-Assistant for this word processing activity. Site 2 suggested Writing Assistant with Word Proof spell checker. The adjusted knowledge base gave the correct recommendation in five of the six repetitions of the case.

Case 19: Reject H0

One of the users incorrectly indicated in the ICE dialogue that she wanted to use the mainframe for this case. Therefore, the recommendation was incorrect in both the preand post-test runs. Of the other 5 replications, none of the recommendations were judged correct for the original version of the ICE knowledge base, while all five were correct for the adjusted knowledge base.

6.5 Comparison of ICE Recommendations to Consultants' Recommendations

When the system was installed at the second corporate site, the twenty cases described above were also solved by six consultants from that IC. The consultants were chosen, using restricted randomization, by the manager of the information center. Together, their skills covered the entire range of software products supported by their information center, as three PC and three mainframe consultants were randomly selected. Each consultant was given a set of the twenty case studies.

For each case, the consultants were asked to offer the same software recommendations they would make if such a user came to the information center for help. The "answers" were compared with the answer given by the "expert" (IC manager). Those data are summarized in Table 6.4. Chi-squared tests measured these hypotheses:

H0:Receiving a valid recommendation for software is independent of the source of the recommendation.
H1:Receiving a valid recommendation for software is not independent of the source of the recommendation.

The conclusion from these data is to reject the null hypothesis in favor of the alternate hypothesis. It appears that ICE gave significantly more correct answers than the IC consultants. However, if we look closely at the data, we see that it needs to be classified more precisely. The cell values that contribute to the high chi-squared value

Table 6.4: Overall Number of Recommendations Judged Correct

	SOURCE			
	PC CONSULTANTS	HOST CONSULTANTS	ICE SYSTEM	TOTAL
Correct	49	23	104	176
Incorrect	11	37	16	64
Total	60	60	120	240
hline Proportion Correct	.8167	.383	.867	.7333

chi-squared = 50.617 df = 2 p < .005

are those from the host consultants. The PC consultants and the ICE system provided recommendations judged to be approximately 82% and 87% correct, respectively, but the host consultants' recommendations were only 38% correct.

The next table (Table 6.5) shows the cases categorized by environment: PC or Mainframe (MF). Three PC consultants solved 33 PC cases and 27 MF cases. Three MF consultants also solved 33 PC cases and 27 MF cases. This table indicates that when working within their specialized environments, the consultants do very well. PC consultants and MF consultants differ significantly on their ability to help with PC problems. On mainframe problems, MF consultants do only marginally better, and the difference is not significant. The PC consultants have some knowledge of mainframe tools, since all have their own accounts on the system for communications purposes. The MF consultants, on the other hand, rarely work with the personal computing environment.

The percentages of correct solutions from each source (ICE System, PC consultants, MF consultants) in each environment (PC and MF) are reported in Table 6.6. Displaying the data in this manner shows that ICE gives correct answers fairly consistently in both the PC and MF environments. The IC consultants tend to give more correct answers in their own specialized environments. This is particularly true of the MF consultants, who answered only nine percent of the PC cases correctly. The surprisingly low percentage (74 percent) of MF consultants correctly solving MF cases is partially explained by the

Table 6.5: Number of Correct Recommendations Made by IC Consultants Within Their Own Environment of Expertise

	CONSULTANTS					
	PC		MF		ALL	
	PROBLEM TYPE:		PROBLEM TYPE:		PROBLEM TYPE:	
	PC	MF	PC	MF	PC	MF
CORRECT	31	18	3	20	34	38
INCORRECT	2	9	30	7	32	16
	33	27	33	27	66	54

chi-squared = 47.91 df = 2 p < .005

fact that the MF consultants are further specialized in supporting just one or two complex MF tools, and know less about the broad range of MF tools available.

Table 6.6: Summary Table - Percentage of Correct Recommendations:

	PC Cases	MF Cases	All Cases
ICE System	.8636	.9444	.867
Consultants:			
PC	.939	.6667	8167
MF	.0909	.7407	.383

This does not suggest that ICE can replace IC consultants. ICE cannot give answers to specific, complex problems users are having with certain software tools. What ICE does do well is to make available the collected knowledge of the various consultants as to which software tools are to be suggested for the routine, recurring activities performed by the end users.

6.6 Summary

Studies at the second corporate site at which ICE was installed indicated that the development and maintenance tools, RAC and M-ICE, make possible the transport of the knowledge base from one IC location to another without changes to the rule base. A successful implementation of ICE was achieved in an environment with time- and location-variant knowledge.

Chapter 7

CONTRIBUTIONS AND FUTURE RESEARCH

Studies conducted by John Kendrick [1979] indicate that between 1929 and 1978 two of the main sources of productivity improvement in the United States were technological innovation and improved allocation of resources. Kendrick concluded that the main challenge to managers was to find means to ensure the most effective use of the knowledge, information technology, and human potential available to them. This research responded to that challenge by investigating the use of expert systems technology as a support mechanism for managers of organizational computing resources. As end users seek to improve their productivity, they are demanding increasing access to the computing resources. Expert systems techniques described in this dissertation help to organize and store the knowledge about available resources sought by end users.

7.1 Contributions of the Research

The specific contributions of this study were made in two main areas: (1) the design and implementation of a structured methodology (RAC and M-ICE) for acquiring knowledge about reusable resources so they can be located and matched to appropriate users, and (2) the design and implementation of an expert system (ICE) with a flexible architecture that can represent both the general core of knowledge applicable to all locations in which it is installed, as well as the local core of knowledge specific to each location.

7.1.1 RAC and M-ICE

RAC (Resource Attribute Charts) is the methodology developed to elicit, combine, and transfer the expert's knowledge to the data structures used by the expert system. A series of four charts offer assistance to the knowledge engineer by providing a structure for interviewing the expert and analyzing the information. The information gathered on the charts is entered into the knowledge base using a menu-driven tool called M-ICE (Maintenance tool for ICE). M-ICE allows for the creation of the external files that define the resources of each IC location. M-ICE was used successfully by IC consultants to create location-specific knowledge bases.

7.1.2 ICE: Flexible Architecture

The flexible design of ICE's architecture ensures that the knowledge base can be easily transported to sites with differing resource sets, and once installed, can be maintained without the need of a knowledge engineer. The design claim of a flexible architecture was examined in an experiment in which the system was transported from one IC site to another. The outcome of the experiment supports the claim that the data structures designed to represent the users, problems, and resources allow for the matching process to take place in a dynamic environment. Whereas in the past, expert systems have not been flexible enough to be updated quickly and easily, ICE overcomes the formidable update problem of knowledge-based management [Feigenbaum and McCorduck 1984, p. 84].

7.1.3 General Contributions

The purpose of ICE is to support the crucial activities involved in the control and management of EUC resources. In this study, the Information Center was the organizational structure assigned to end user support. Five major responsibilities of information centers were identified by the personnel of the sites into which ICE was installed: (1) consultation with end users, (2) policy formulation and enforcement (3) tracking of users, (4) target marketing of IC resources to users, and (5) training. ICE currently provides support for four of these areas.

7.1.3.1 Consultation With End Users

The major contribution made by ICE in supporting the activities of the IC is its consulting function. Evidence supporting this claim has been provided in Chapters 5 and 6. ICE is not meant to replace the IC personnel, but rather to relieve the consultants from dealing with the more simple, common, and repetitive requests. The users can, through dialogue with ICE, determine a possible set of software to meet their needs. This list can then be brought to the consultant for further information and advice, or the users can try out the software to determine which best fits their requirements.

7.1.3.2 Policy Formulation and Enforcement

As the knowledge base of software tools is built in each IC site, it will necessarily mirror the current corporate policies and recommendations for software resources. As the system accumulates user histories, needs, preferences, and experiences with the software tools available, and as the library of corporate components catalogued by the system is modified by the elimination of some software packages, the modification of others, and the addition of new tools, the sum of that information will itself suggest–perhaps even require–changes in corporate policy, advice, and recommendations about the IC resources.

What began as an enforcer of corporate preferences and capabilities in supporting certain software resources, will migrate to the role of participant in shaping corporate direction. The possibility of this influence should be the subject of further investigation.

7.1.3.3 Tracking of Users

The ICE system is being used in one of the IC sites to collect some of the data necessary in building profiles of end user capabilities and needs within the corporation. At the end of each consultation session, ICE stores in an external file the user's name, phone number, and software recommendation received. IC consultants regularly review these files and contact the users to follow up on user needs and satisfaction. Those consultations that resulted in no recommendation being made because of a lack of appropriate software receive immediate attention. Users are contacted either personally or by computer mail to get further information so that their needs can be met.

Further research in this area of tracking includes implementing an automatic record keeping facility of all users who rely on certain software packages, so that when new releases are obtained, or information comes out concerning those packages, the users can receive immediate notification.

7.1.3.4 Target Marketing of IC Resources to Users

The most successful ICs have been able to identify key end users who develop systems that provide large company payoffs, and have concentrated on helping them choose application approaches, database access, or have provided necessary training to them. Through the tracking capabilities described above, ICE could provide information that would help ICs to differentiate their user population, providing some services to all, and specialized services to certain populations.

7.2 Future Research Ideas

In its current implementation, ICE is limited to only two kinds of resource recommendations: software tools, and/or information about consultants who have expertise in their problem areas. A much more varied and sophisticated role is envisioned for the ICE system. Much can be done to enhance its capabilities, and its architecture can be applied to other resource management issues. Examples of enhancements and further applications are now given.

7.2.1 Training

ICE currently supports the consulting function of IC personnel. But just as IC staff must provide multiple services to end users, other dimensions can be added to ICE to increase its support of IC responsibilities. In addition to recommending software tools, ICE could maintain a schedule of classes or workshops for these tools, and offer on-line information or enrollment to the interested user. A fully developed ICE system will surely include, as part of the interface with users, a demonstration of the capabilities of the tools recommended, or even an on-line instruction session for the selected software.

7.2.2 ICE as a Decision Support Tool for Software Purchasing

ICE can also be used by the IC consultant to help make software purchasing decisions. When new software packages or descriptions arrive for evaluation in the IC, the consultant can enter a consultation with ICE, fitting the attributes of the typical user to those of the new tool. If software already in the system meets those needs, it is possible that the new tool is not needed; at very least, a comparative evaluation should be made. Consultants also should study those stored consultations of users for which no software package could be recommended to the user. If that user's need can be verified as one that should be met, the consultant can search for the necessary software to add to the resources of the Information Center. Research documenting this use of ICE needs to be undertaken in the future.

7.2.3 Further Enhancements for the Maintenance Tools

Much can be done (1) to increase the ease with which the IC consultant interacts with the M-ICE tool for maintaining the knowledge base, and (2) to enhance the versatility of ICE in offering different categories of software depending on the nature of the user population.

Enhancement of M-ICE. In its current form, the maintenance tool M-ICE is a forms-driven utility where the IC consultant "fills in the blank" to enter attribute values for software. The process requires the IC consultant to be familiar with what has previously been entered, so that new software can be defined relative to previous definition. Future research should include rewriting into a rule-based "intelligent" tool that assigns values to software attributes based on dialogue with the IC consultant. This would involve automating the Resource Attribute Charts, and making them a part of M-ICE. This dialogue would allow for comparison of the new tool with software previously entered into the knowledge base. Based on the relative placement of the new tool, attribute values would be automatically assigned, and ratings of existing tools would be adjusted as necessary. Making M-ICE into a small expert system would allow for new software tools to be defined relative to the existing tools without the IC consultant having to pay attention to the exact attribute values of previous tools.

Versatility. The current version of ICE divides the categories of software resources into seven categories. A change in those categories, or a change in the attributes used to define software tools, requires the services of a knowledge engineer in making the necessary changes to the dialogue and the rule base. Future versions of the maintenance utility M-ICE should generate a report detailing the exact changes that need to be made in the ICE dialogue and rules by the knowledge engineer to accommodate new categories of software or new attributes defining that software.

7.2.4 Applications of ICE Architecture to Other Environments

The immediate applications of the flexible architecture of the ICE model to problems of resource identification and selection are enormous. The generality and adequacy of the knowledge representation and structure, and the knowledge elicitation methodology, must be examined in domains beyond the Information Center.

An executive information officer from a major computer company who saw the ICE system has proposed adapting the architecture design to aid in the process of preparing Request for Proposals (RFPs) for customer orders. The RFP contains definable sections, many of which could be reused. Relevant sections could be analyzed using the Resource Attribute Charts to determine important attributes by which to locate and access previous documentation that could be reused. ICE could then offer dialogue to the RFP writer that would identify needs for the current proposal. ICE would identify past proposals that might be searched for relevant modules.

An information systems administrator from the Army is exploring the possibility of using the ICE architecture for managing reusable code modules. ICE would have to capture not only the syntax of the code, but also the semantics and the coupling between modules. Research needs to be done to apply the ICE architecture to a heterogeneous library of codes and high level requirements of programs.

7.3 Summary

The viability of a rule-based consultation system as a mechanism for bringing together knowledge about users, problems, and resources for the purpose of effective resource management has been demonstrated within the context of the Information Center. ICE's flexibility and transportability to sites with different sets of resources has been examined, and those claims substantiated. The desired outcome is an improved approach to formalizing and managing knowledge for an information center. It is possible that the architecture used for ICE could facilitate the assignment of resources to users in other problem environments as well.

Appendix A

MATCHING PROFILE ATTRIBUTES

1. SOFTWARE CATEGORY ATTRIBUTES

   Definition: Determines the main category in which the user needs software.

   - DM_MANAGE: data management software: data base, file management, queries, or reports.
   - DA_ANALYZE_DATA: data analysis, array manipulation, or mathematical manipulation of data.
   - PG_GRAPHICS: software that will produce graphics.
   - D_DOCUMENTS: software is in preparing, proofreading, and/or searching for documents, such as letters, memos, reports, or other written communication.
   - PM_PROJECT_MGT: software that will facilitate project management activities.
   - U_UTILITIES: utility aids, such as file and directory management, disk management, or productivity tools.

2. DATA MANAGEMENT ATTRIBUTES

   Definition: data base, file management, queries, reports

   - DM_GRAPHICS_DATA: Interfaces to other graphics tools.

- DM_CHARTS: Produces charts from the data: bar/ column/ line/ pie, etc.
- DM_EASE_OF_QUERY: Language provides ease of query into data base.
- DM_SHARABILITY: More than one user can access and update the data.
- DM_RESPONSE: High values represent fast response times, low values represent slow response time.
- DM_VIEW_DATA: Built-in facility to view data already stored in the computer.
- DM_SECURITY: Built-in security for data access protection.
- DM_COMPUTATION: Facility for calculation of data.
- DM_SIMPLE_REPORT: Facility for production of simple reports.
- DM_REPORT_COMPLEXITY: Facility for production of more complex reports, with such options as printed charts.
- DM_DATA_SUMMARY: Facility to print summaries of the data, as well as the original data.
- DM_TABLES: Facility to arrange data in tabular format for display.
- DM_SMALL: Facility to handle less than 1000 records.
- DM_MEDIUM: Facility for between 1000-3000 records.
- DM_LARGE: Facility for large quantities of data: over 3000 records.

3. DATA ANALYSIS ATTRIBUTES

Definition: statistics, spreadsheets, number crunching or other mathematical manipulation, reports from data, graphical representation of data

- DA_STATISTICS: Facility for performing statistical analyses on data.
- DA_SIMPLE_STAT: Facility to facilitate simple analyses of statistical data, such as averages, means, standard deviations.

- DA_ADVANCED_STAT: Facility for advanced analyses of statistical data, such as Chi-square test, t-test, canonical correlation, principal components, etc.
- DA_SPREADSHEETS: Spreadsheet facility.
- DA_MACRO: Facility that allows creation of macros.
- DA_DATA_MODELING: Facilities for data modeling, such as curve fitting, time series analysis, correlation, etc.
- DA_QUALITY_CONTROL: Facility to analyze data for quality control, defect analysis.
- DA_GRAPHS: Facility to produce scientific/mathematical graphs from the analyzed data.
- DA_CHARTS: Facility for production of pie, vertical and horizontal bar, star charts etc., from the analyzed data.
- DA_REPORTS: Facility for production of reports.
- DA_FORECASTING: Facility in forecasting: time series, exponential smoothing, autocorrelation.
- DA_FINANCIAL_ANALYSIS: Facility for performance of financial analyses, such as return on investment, compound interest, capital budgeting, etc.

4. GRAPHICS ATTRIBUTES

   Definition: presentation graphics, business graphics, graphics for design, flowcharting, on-line graphics

   - PG_DRAWING: Facility for "free hand" drawing.
   - PG_SYMBOLS: Facility that provides a library of graphic symbols.
   - PG_MECHANICAL_DESIGN: Facility for mechanical design.
   - PG_FACILITY_DESIGN: Facility for design of floor layout, electrical wiring, etc., as related to facility engineering.

- PG_TEXT: Facility to produce text presentation materials.
- PG_CHARTS: Facility to produce charts: pie, bar, etc.
- PG_SCIENTIFIC_GRAPHS: Facility to produce scientific graphics, such as wave charts.
- PG_FLOWCHARTS: Facility to produce flowcharts.
- PG_PAPER: Facility to produce graphics on a paper output form.
- PG_ONLINE: Facility to prepare presentation of online/unattended, CRT screen.
- PG_FONTS: Selection of different font styles.
- PG_CHARACTER_SIZE: Selection of different character sizes.
- PG_COLOR: Facility that supports the production of graphics in different colors.
- PG_INTERFACE: Interface with other common graphics products.

5. DOCUMENT PREPARATION ATTRIBUTES

Definition: text editors, text processors

- D_TOC: Facility for production of table of contents.
- D_INDEX: Facility for production of index.
- D_SUB_SUPERSCRIPT: Facility for production of subscripts and superscripts.
- D_COLUMN: Facility for column-processing.
- D_FOOTNOTES: Facility for production of footnotes.
- D_SPELLER: Spell checker.
- D_OUTLINE: Facility for automatic outlining.
- D_MATH: Facility for insertion of simple mathematical symbols into the text.
- D_MERGE: Facility for merging of files.

- D_MACRO: Facility for user-defined keys or functions, for the purpose of repetitive text processing.
- D_FONTS: Different font styles.
- D_COLOR: Color printing.
- D_ASSISTANT: Tool part of the Assistant Series for data integration.
- D_GRAPHICS: Facility for imbedding graphics into the text.

6. PROJECT MANAGEMENT ATTRIBUTES

   Definition: tracking, resource leveling, reports tasks or less.

   - PM_MEDIUM: Facilitates management of projects with 50 - 200 tasks.
   - PM_LARGE: Facilitates management of projects with over 200 tasks.
   - PM_GRAPHICS: Facility for production of cost graphs, PERT charts, Gantt charts, etc.
   - PM_REPORT: Facility for production of reports – higher values indicate capability for more complex reports.
   - PM_MULTIPLE: Facility to process multiple subprojects at the same time.
   - PM_RISK_ANALYSIS: Facility to do risk analysis on the project plan.
   - PM_TRACKING: Facility for project control activities: the capability to compare planned numbers versus actual numbers, etc.
   - PM_RESOURCE_LEVELING: Facility to help equalize distribution of resources.
   - PM_RESOURCE_ALLOC: Facility to allocate resources automatically, based upon resource availability.
   - PM_TYPE: High rating means tool that supports many different types of resources.
   - PM_NUMBER: Facility handles large total number of resources.

- PM_CALENDAR: Facility to support user-defined calendar.
- PM_CPM: Facility for Critical Path Method.
- PM_PERT: Facility for PERT methodology: slack time analysis, probabilistic analysis.
- PM_COST: Facility for cost analysis on projects.

7. UTILITIES AND PRODUCTIVITY TOOLS ATTRIBUTES

Definition: mail, individual time management, calendars, memos, calculators, data transfer, pc disk utilities, host communication

- U_DIR_FILE: Facility is provided for directory or file management.
- U_DISK: Facility is provided for disk management functions, such as hard disk backup, recapturing erased files, etc.
- U_PRODUCTIVITY: Facility provided in such areas as printer support, keyboard lock, or other productivity tools.

8. COMMON SET ATTRIBUTES

- E_BEGINNING: Software is simple to learn and to use.
- E_INTERMEDIATE: Software is for user who indicates knowledge of common functions of the computer environment, and has had some previous experience working with computer applications.
- E_ADVANCED: Software for user who is highly proficient with computers.
- CM_BATCH: Software will run applications in batch mode.
- CM_ONLINE: Software will run online applications.
- CM_VM: Tool for the VM host environment.
- CM_MVS: Tool for the VM host environment.

- CM_PCDISK: Tool for the PC.
- CM_AS: Data is stored as an AS dataset.
- CM_DB2: Previous use of DB2.
- CM_SAS: Data is stored as a SAS dataset.
- CM_SQL: Previous use of SQL.
- CM_DISK: Data stored on A-disk.
- CM_BARCODE: Information can be input using barcode reader.
- CM_CARDS: Information can be input using cards reader.
- CM_KEYBOARD: Information can be input using keyboard.
- CM_OPTICAL: Information can be input using optical devices.
- CM_PAGE_READER:Information can be input using page-reader.
- CM_TAPE: Information can be input using tape device.

Appendix B

SAMPLE M-ICE REPORTS

ATTRIBUTE REPORT page 1

02/22/87 -- 23:41:53

1 - BEGINNING -- (5.00)

Software is classified as simple to learn and use. Users unfamiliar with computer hardware, or those who rarely use the computer, will still be able to use this with little learning time or frustration.

---

2 - INTERMEDIATE -- (5.00)

Software is recommended by the user with intermediate level skill--one who is familiar with computer hardware and users the computer regularly to perform job.

---

3 - ADVANCED -- (5.00)

Software is recommended for user who is highly proficient with computers and who is dependent on the computer to perform his/her job. Programming experience often necessary to utilize the software.

---

4 - BATCH_MODE -- (1.00)

Software allows for programs to be run in batch mode.

---

5 - ONLINE -- (1.00)

Software allows for online mode only; no batch mode.

---

6 - VM -- (10.00)

Software is supported by the host, VM.

---

7 - MVS -- (10.00)

Software is supported on the host, MVS.

---

8 - PCDOS -- (10.00)

This software is for PCs.

---

9 - AS -- (1.00)

The software interfaces with VM/AS (Application System).

ATTRIBUTE REPORT page 2

02/22/87 -- 23:41:58

10 - DB2 -- (1.00)

The software interfaces with DB2.

---

11 - SAS -- (1.00)

The software interfaces with SAS.

---

12 - SQL -- (1.00)

The software interfaces with SQL.

---

13 - A_DISK -- (1.00)

The software can access data that is stored on the user's "A" disk.

---

14 - BARCODE_READER -- (1.00)

Software accepts data entered with a barcode_reader.

---

15 - COMPUTER_CARDS -- (1.00)

Data stored on computer cards will be read by this software.

---

16 - KEYBOARD -- (1.00)

Software accepts data entered from the keyboard.

---

17 - OPTICAL_READER -- (1.00)

Software accepts data input from an optical reader.

---

18 - PAGE_READER -- (1.00)

Software accepts data input from a page reader.

---

19 - TAPE -- (1.00)

Software accepts data input from a tape.

ATTRIBUTE REPORT page 3

02/22/87 -- 23:42:02

20 - DM_MANAGE_DATA -- (10.00)

Software is classified as a data management tool: data base, file management queries, reports.

---

21 - DM_GRAPHICS -- (1.00)

Software has capability to interface to other graphics tools.

---

22 - DM_CHARTS -- (5.00)

Data can be used to produce charts: bar/column/line/pie, etc.

---

23 - DM_DATA_QUERIES -- (1.00)

Software provides an easy-to-query language.

---

24 - DM_SHARABILITY -- (5.00)

Software allows more than one user to access and update the data.

---

25 - DM_RESPONSE -- (1.00)

The online response time of a query: high values mean fast response time, (7-10), while low values (3-5) mean slower response time.

---

26 - DM_VIEW_DATA -- (5.00)

Software contains built-in facility to view data already stored in the computer.

---

27 - DM_SECURITY -- (1.00)

Software provides built-in data access protection.

---

28 - DM_COMPUTATION -- (3.00)

Facility for calculation is embedded into this data management tool.

---

29 - DM_EASY_REPORT -- (1.00)

Software provides facilities for simple user-defined report format.

ATTRIBUTE REPORT page 4

02/22/87 -- 23:42:06

No complex formatting can be done.

---

30 - DM_COMPLEX_RPTS -- (4.00)

Software offers capabilities for more complex formatting of reports, with such options as printed charts and/or graphics.

---

31 - DM_DATA_SUMMARY -- (1.00)

Software has capability to print summaries of the data as well as the original data.

---

32 - DM_TABLES -- (1.00)

Software has the capability to arrange data in tabular format for display.

---

33 - DM_SMALL -- (5.00)

This software tool is recommended when there are less than 1000 records to manage.

---

34 - DM_MEDIUM -- (5.00)

This data management software handles 3000 records or less.

---

35 - DM_LARGE -- (5.00)

The database tool is capable of handling large quantities of data, to be shared among many users. (LARGE is defined as anything in excess of 3000 records.

---

36 - PM_PROJECT_MGMT -- (10.00)

The main function of this software is to provide support for project management activities.

---

37 - PM_SMALL -- (7.00)

Software recommended for 50 tasks or less.

---

38 - PM_MEDIUM -- (7.00)

ATTRIBUTE REPORT   page 5

02/22/87 -- 23:42:10

Software recommended for between 50 and 200 tasks.

---

39 - PM_LARGE -- (7.00)

Software recommended when number of tasks to manage is over 200 in number.

---

40 - PM_GRAPHICS -- (5.00)

Software provides graphics capabilities to produce such things as cost graphs, PERT charts, or Gantt charts.

---

41 - PM_REPORT -- (1.00)

Complexity of hardcopy reports produced: Rating of 7-10 means that many functions are provided, rating of 3-5 indicates simple reports, 0 indicates no reports.

---

42 - PM_MULTIPLE -- (1.00)

Software can process multiple subprojects/projects.

---

43 - PM_RISK -- (1.00)

Software provides facility to do risk analysis on the project plan.

---

44 - PM_TRACKING -- (1.00)

Software has capability to perform project control; that is, to compare planned numbers versus actual numbers

---

45 - PM_LEVELING -- (1.00)

Software has capability to perform resource leveling.

---

46 - PM_ALLOCATE -- (1.00)

Software has capability to allocate resources automatically, based upon resource availability.

---

47 - PM_TYPE -- (1.00)

High rating here means that the software supports many different types of resources. Low rating means that the software supports only a small number of resource types.

---

48 - PM_NUMBER -- (1.00)

High rating means that the software can handle a large total number of resources.

---

49 - PM_CALENDAR -- (1.00)

Software has capability to support user-defined calendar.

---

50 - PM_CPM -- (1.00)

Software offers capability facilities to do Critical Path Method.

---

51 - PM_PERT -- (1.00)

Software supports the PERT project management methodology; i.e., supports slack time analysis, probabilistic analysis of project schedule, etc.

---

52 - PM_COST -- (1.00)

Software has capability to perform cost analysis.

---

53 - DA_ANALYZE_DATA -- (10.00)

Tool's main function is for data analysis, array manipulation, or mathematical manipulation.

---

54 - DA_STATISTICS -- (8.00)

Tool is mainly for performing statistical analyses.

---

55 - DA_SIMPLE_STAT -- (7.00)

Software will facilitate simple analyses of statistical data, such as averages, means, standard deviations, linear regression, etc.

---

56 - DA_ADVANCE_STAT -- (8.00)

Software will facilitate advanced analyses of statistical data, such as

Chi-square test, t-test, z-test, canonical correlation, trend analysis, time series forecasting, discriminant analysis, etc.

---

57 - DA_SPREADSHEET -- (10.00)

Spreadsheet facility is provided by this software. Rating based on quality of the spreadsheet in terms of offering many different capabilities.

---

58 - DA_MACRO -- (5.00)

Software allows creation of macros by the user.

---

59 - DA_DATA_MODEL -- (1.00)

Software facilitates data modeling, such as curve fitting, time series analysis, correlation etc.

---

60 - DA_QUALITY_CONT -- (1.00)

Software has capability to analyze data for quality control/defect analysis.

---

61 - DA_GRAPHS -- (5.00)

Software has capability to produce scientific/mathematical graphs from the analyzed data.

---

62 - DA_CHARTS -- (5.00)

Software has the capability to produce charts from the analyzed data, such as pie, star, vertical or horizontal bar charts, data plots.

---

63 - DA_REPORTS -- (5.00)

Capability to produce reports. Rating based on quality/flexibility in defining user formats for reports. High rating (8-10) indicates high quality, low (3-5) indicates few reporting capabilities, 0 indicates that no reports can be produced.

---

64 - DA_FORECASTING -- (1.00)

Software facilitates forecasting, such as time series, exponential smoothing, autocorrelation.

ATTRIBUTE REPORT

02/22/87 -- 23:42:24

page 8

65 - DA_FINANCIAL -- (1.00)

Software has capability to perform financial analyses, such as return on investment, compound interest, capital assets, market forecasting, costing, tax analyses, etc.

66 - D_DOCUMENTS -- (10.00)

Software's main functions are in preparing, proofreading, and searching for documents such as letters, memos, or other formal documents.

67 - D_TOC -- (1.00)

Software provides facility for production of table of contents.

68 - D_INDEX -- (1.00)

Software provides facility for production of index.

69 - D_SUB_SUPER -- (1.00)

Software provides facility for production of subscripts and superscripts.

70 - D_COLUMN -- (1.00)

Software allows for column-processing.

71 - D_FOOTNOTES -- (1.00)

Software has facility for production of footnotes.

72 - D_SPELLER -- (1.00)

Software has spell checking facilities.

73 - D_OUTLINE -- (1.00)

Software has facility for automatic outlining.

ATTRIBUTE REPORT page 9

02/22/87 -- 23:42:29

74 - D_MATH -- (1.00)

Simple mathematical symbols are available for insertion into the text.

---

75 - D_MERGE -- (1.00)

Capability is provided to easily merge files.

---

76 - D_MACRO -- (1.00)

Capability for user-defined keys or functions for the purpose of repetitive text processing.

---

77 - D_FONTS -- (5.00)

Tool offers different font styles for printing.

---

78 - D_COLOR -- (1.00)

Tool provides facility for color printing.

---

79 - D_ASSISTANT -- (7.00)

Software is part of the Assistant Series, and thus can be integrated with graphics and spreadsheet data from other Assistant packages.

---

80 - D_GRAPHICS -- (4.00)

Facilities for imbedding graphics into the text are provided.

---

81 - PG_GRAPHICS -- (10.00)

The main functions provided by this software are for the production of graphics.

---

82 - PG_DRAWING -- (8.00)

Tool allows for "free hand" drawing.

---

83 - PG_SYMBOLS -- (8.00)

Tool provides a library of graphic symbols.

---

84 - PG_MECHANICS -- (9.00)

Tool provides mechanical design facilities.

---

85 - PG_FACILITY -- (9.00)

Tool provides facilities to design floor layout, electrical wiring, etc; as related to facility engineering

---

86 - PG_TEXT -- (8.00)

Presentation materials can be prepared in textual format only.

---

87 - PG_CHARTS -- (8.00)

Tool is good for preparation of charts, such as pie charts, bar charts, etc.

---

88 - PG_SCIENTIFIC -- (1.00)

Scientific graphics can be produced with this tool, such as wave charts.

---

89 - PG_FLOWCHART -- (1.00)

Software facilitates the production of flowcharts.

---

90 - PG_PAPER -- (1.00)

Software supports the production of graphics for paper output.

---

91 - PG_ONLINE -- (8.00)

Online/unattended presentation on a CRT screen is accommodated by this software.

---

92 - PG_FONTS -- (5.00)

Tool provides selection of different font styles.

---

93 - PG_CHARACTER --, (5.00)

Tool provides choices of different character sizes.

---

94 - PG_COLOR -- (1.00)

Tool supports the production of graphics in different colors.

---

95 - PG_INTERFACE -- (1.00)

Tool interfaces with other common graphics products.

---

96 - U_UTILITIES -- (10.00)

Software falls into the general category of utility aids, such as file and directory management, disk management, productivity aids for menu and printers.

---

97 - U_DIR_FILE -- (1.00)

Software is categorized as a utility aid that falls into the category of directory or file management.

---

98 - U_DISK -- (1.00)

Software is a utility aid that performs disk management functions, such as hard disk backup, recapturing erased files, clears files from a disk, etc.

---

99 - U_PRODUCTIVITY -- (1.00)

Software is a utility aid that does not fit into the category is file/disk management, but performs some other task which enhances productivity, such as printer support or output enhancement.

CONSULTANT REPORT page 1

02/22/87 -- 23:43:37

ID NUMBER	CONSULTANT NAME	PHONE NUMBER
1	Kendall Cliff	111-1111
2	George Walters	222-2222
3	Betty Evans	333-3333
4	Irene Chen	444-4444
5	Belinda Wong	555-5555

TOOL REPORT page 1

02/22/87 -- 23:31:55

1 - D1 -- PERSONAL EDITOR

IBM PC Personal Editor is a high performance, full screen editor for any desired text file: memos, letters, documents, program files, etc. It is very fast, scrolling almost instantly and searching and changing text in seconds. Personal Editors provides full support for inserting, deleting, moving, and copying text. It allows for redefinition of the keyboard. Up to 20 files at be edited at one time in memory, and it can edit files larger than the memory size. It is an extremely powerful editor for the PC. Any user of Personal Editor should also use Word Proof II, a package that does spelling and synonym checks on 125,000 words.

CONSULTANT #1: 1, CONSULTANT #2:

BEGINNING	(5.00)	:	8.00		INTERMEDIATE	(5.00)	:	10.00
ADVANCED	(5.00)	:	10.00		BATCH_MODE	(1.00)	:	0.00
ONLINE	(1.00)	:	10.00		VM	(10.00)	:	0.00
MVS	(10.00)	:	0.00		PCDOS	(10.00)	:	10.00
AS	(1.00)	:	0.00		DB2	(1.00)	:	0.00
SAS	(1.00)	:	0.00		SQL	(1.00)	:	0.00
A_DISK	(1.00)	:	0.00		BARCODE_READER	(1.00)	:	0.00
COMPUTER_CARDS	(1.00)	:	0.00		KEYBOARD	(1.00)	:	10.00
OPTICAL_READER	(1.00)	:	0.00		PAGE_READER	(1.00)	:	0.00
TAPE	(1.00)	:	0.00		DM_MANAGE_DATA	(10.00)	:	0.00
DM_GRAPHICS	(1.00)	:	0.00		DM_CHARTS	(5.00)	:	0.00
DM_DATA_QUERIES	(1.00)	:	0.00		DM_SHARABILITY	(5.00)	:	0.00
DM_RESPONSE	(1.00)	:	0.00		DM_VIEW_DATA	(5.00)	:	0.00
DM_SECURITY	(1.00)	:	0.00		DM_COMPUTATION	(3.00)	:	0.00
DM_EASY_REPORT	(1.00)	:	0.00		DM_COMPLEX_RPTS	(4.00)	:	0.00
DM_DATA_SUMMARY	(1.00)	:	0.00		DM_TABLES	(1.00)	:	0.00
DM_SMALL	(5.00)	:	0.00		DM_MEDIUM	(5.00)	:	0.00
DM_LARGE	(5.00)	:	0.00		PM_PROJECT_MGMT	(10.00)	:	0.00
PM_SMALL	(7.00)	:	0.00		PM_MEDIUM	(7.00)	:	0.00
PM_LARGE	(7.00)	:	0.00		PM_GRAPHICS	(5.00)	:	0.00
PM_REPORT	(1.00)	:	0.00		PM_MULTIPLE	(1.00)	:	0.00
PM_RISK	(1.00)	:	0.00		PM_TRACKING	(1.00)	:	0.00
PM_LEVELING	(1.00)	:	0.00		PM_ALLOCATE	(1.00)	:	0.00
PM_TYPE	(1.00)	:	0.00		PM_NUMBER	(1.00)	:	0.00
PM_CALENDAR	(1.00)	:	0.00		PM_CPM	(1.00)	:	0.00
PM_PERT	(1.00)	:	0.00		PM_COST	(1.00)	:	0.00
DA_ANALYZE_DATA	(10.00)	:	0.00		DA_STATISTICS	(8.00)	:	0.00
DA_SIMPLE_STAT	(7.00)	:	0.00		DA_ADVANCE_STAT	(8.00)	:	0.00
DA_SPREADSHEET	(10.00)	:	0.00		DA_MACRO	(5.00)	:	0.00
DA_DATA_MODEL	(1.00)	:	0.00		DA_QUALITY_CONT	(1.00)	:	0.00
DA_GRAPHS	(5.00)	:	0.00		DA_CHARTS	(5.00)	:	0.00
DA_REPORTS	(5.00)	:	0.00		DA_FORECASTING	(1.00)	:	0.00
DA_FINANCIAL	(1.00)	:	0.00		D_DOCUMENTS	(10.00)	:	10.00
D_TOC	(1.00)	:	0.00		D_INDEX	(1.00)	:	0.00
D_SUB_SUPER	(1.00)	:	0.00		D_COLUMN	(1.00)	:	0.00
D_FOOTNOTES	(1.00)	:	0.00		D_SPELLER	(1.00)	:	1.00
D_OUTLINE	(1.00)	:	0.00		D_MATH	(1.00)	:	0.00

TOOL REPORT

02/22/87 -- 23:32:03

D_MERGE	(1.00) : 10.00		D_MACRO	(1.00) : 0.00	
D_FONTS	(5.00) : 0.00		D_COLOR	(1.00) : 10.00	
D_ASSISTANT	(7.00) : 0.00		D_GRAPHICS	(4.00) : 0.00	
PG_GRAPHICS	(10.00) : 0.00		PG_DRAWING	(8.00) : 0.00	
PG_SYMBOLS	(8.00) : 0.00		PG_MECHANICS	(9.00) : 0.00	
PG_FACILITY	(9.00) : 0.00		PG_TEXT	(8.00) : 0.00	
PG_CHARTS	(8.00) : 0.00		PG_SCIENTIFIC	(1.00) : 0.00	
PG_FLOWCHART	(1.00) : 0.00		PG_PAPER	(1.00) : 0.00	
PG_ONLINE	(8.00) : 0.00		PG_FONTS	(5.00) : 0.00	
PG_CHARACTER	(5.00) : 0.00		PG_COLOR	(1.00) : 0.00	
PG_INTERFACE	(1.00) : 0.00		U_UTILITIES	(10.00) : 0.00	
U_DIR_FILE	(1.00) : 0.00		U_DISK	(1.00) : 0.00	
U_PRODUCTIVITY	(1.00) : 0.00				

2 - D2 -- E2

E2 is a text editor that allows full screen editing. It is useful for large files, and up to 20 files can be edited from different subdirectories. E2 provides flexible block marking and movement as well as search and replace options. Keys can be redefined to user's choice. Color displays are also supported. Since there is not a built-in spell checker, it is advised that anyone choosing E2 should also use Word Proof II, a software tool with a 125,000 word dictionary that not only checks spelling but also finds synonyms.

CONSULTANT #1: , CONSULTANT #2:

BEGINNING	(5.00)	: 8.00	INTERMEDIATE	(5.00)	: 10.00
ADVANCED	(5.00)	: 0.00	BATCH_MODE	(1.00)	: 0.00
ONLINE	(1.00)	: 10.00	VM	(10.00)	: 0.00
MVS	(10.00)	: 0.00	PCDOS	(10.00)	: 10.00
AS	(1.00)	: 0.00	DB2	(1.00)	: 0.00
SAS	(1.00)	: 0.00	SQL	(1.00)	: 0.00
A_DISK	(1.00)	: 0.00	BARCODE_READER	(1.00)	: 0.00
COMPUTER_CARDS	(1.00)	: 0.00	KEYBOARD	(1.00)	: 10.00
OPTICAL_READER	(1.00)	: 0.00	PAGE_READER	(1.00)	: 0.00
TAPE	(1.00)	: 0.00	DM_MANAGE_DATA	(10.00)	: 0.00
DM_GRAPHICS	(1.00)	: 0.00	DM_CHARTS	(5.00)	: 0.00
DM_DATA_QUERIES	(1.00)	: 0.00	DM_SHARABILITY	(5.00)	: 0.00
DM_RESPONSE	(1.00)	: 0.00	DM_VIEW_DATA	(5.00)	: 0.00
DM_SECURITY	(1.00)	: 0.00	DM_COMPUTATION	(3.00)	: 0.00
DM_EASY_REPORT	(1.00)	: 0.00	DM_COMPLEX_RPTS	(4.00)	: 0.00
DM_DATA_SUMMARY	(1.00)	: 0.00	DM_TABLES	(1.00)	: 0.00
DM_SMALL	(5.00)	: 0.00	DM_MEDIUM	(5.00)	: 0.00
DM_LARGE	(5.00)	: 0.00	PM_PROJECT_MGMT	(10.00)	: 0.00
PM_SMALL	(7.00)	: 0.00	PM_MEDIUM	(7.00)	: 0.00
PM_LARGE	(7.00)	: 0.00	PM_GRAPHICS	(5.00)	: 0.00
PM_REPORT	(1.00)	: 0.00	PM_MULTIPLE	(1.00)	: 0.00
PM_RISK	(1.00)	: 0.00	PM_TRACKING	(1.00)	: 0.00
PM_LEVELING	(1.00)	: 0.00	PM_ALLOCATE	(1.00)	: 0.00

TOOL REPORT

02/22/87 -- 23:32:09

PM_TYPE	(1.00)	0.00	PM_NUMBER	(1.00)	0.00
PM_CALENDAR	(1.00)	0.00	PM_CPM	(1.00)	0.00
PM_PERT	(1.00)	0.00	PM_COST	(1.00)	0.00
DA_ANALYZE_DATA	(10.00)	0.00	DA_STATISTICS	(8.00)	0.00
DA_SIMPLE_STAT	(7.00)	0.00	DA_ADVANCE_STAT	(8.00)	0.00
DA_SPREADSHEET	(10.00)	0.00	DA_MACRO	(5.00)	0.00
DA_DATA_MODEL	(1.00)	0.00	DA_QUALITY_CONT	(1.00)	0.00
DA_GRAPHS	(5.00)	0.00	DA_CHARTS	(5.00)	0.00
DA_REPORTS	(5.00)	0.00	DA_FORECASTING	(1.00)	0.00
DA_FINANCIAL	(1.00)	0.00	D_DOCUMENTS	(10.00)	10.00
D_TOC	(1.00)	0.00	D_INDEX	(1.00)	0.00
D_SUB_SUPER	(1.00)	0.00	D_COLUMN	(1.00)	0.00
D_FOOTNOTES	(1.00)	0.00	D_SPELLER	(1.00)	5.00
D_OUTLINE	(1.00)	0.00	D_MATH	(1.00)	0.00
D_MERGE	(1.00)	0.00	D_MACRO	(1.00)	0.00
D_FONTS	(5.00)	0.00	D_COLOR	(1.00)	10.00
D_ASSISTANT	(7.00)	0.00	D_GRAPHICS	(4.00)	0.00
PG_GRAPHICS	(10.00)	0.00	PG_DRAWING	(8.00)	0.00
PG_SYMBOLS	(8.00)	0.00	PG_MECHANICS	(9.00)	0.00
PG_FACILITY	(9.00)	0.00	PG_TEXT	(8.00)	0.00
PC_CHARTS	(8.00)	0.00	PG_SCIENTIFIC	(1.00)	0.00
PG_FLOWCHART	(1.00)	0.00	PG_PAPER	(2.00)	0.00
PG_ONLINE	(8.00)	0.00	PG_FONTS	(5.00)	0.00
PG_CHARACTER	(5.00)	0.00	PG_COLOR	(1.00)	0.00
PG_INTERFACE	(1.00)	0.00	U_UTILITIES	(10.00)	0.00
U_DIR_FILE	(1.00)	0.00	U_DISK	(1.00)	0.00
U_PRODUCTIVITY	(1.00)	0.00			

3 - D3 -- PROFESSIONAL ED

IBM PC Professional Editor is a high performance, full screen editor. It is similar to Personal Editor, except that it provides an 8-column prefix area to the left of each line for entering line commands (for copying text, etc.). This software tool has much in common with Host SPF. It provides programmable function keys, data macro capabilities, and merge file options. Word Proof II should be used in conjunction with this package; Word Proof is a spelling and synonym checker for over 125,000 words.

CONSULTANT #1: 2, CONSULTANT #2:

BEGINNING	(5.00)	7.00	INTERMEDIATE	(5.00)	10.00
ADVANCED	(5.00)	7.00	BATCH_MODE	(1.00)	0.00
ONLINE	(1.00)	10.00	VM	(10.00)	0.00
MVS	(10.00)	0.00	PCDOS	(10.00)	10.00
AS	(1.00)	0.00	DB2	(1.00)	0.00
SAS	(1.00)	0.00	SQL	(1.00)	0.00
A_DISK	(1.00)	0.00	BARCODE_READER	(1.00)	0.00
COMPUTER_CARDS	(1.00)	0.00	KEYBOARD	(1.00)	10.00
OPTICAL_READER	(1.00)	0.00	PAGE_READER	(1.00)	0.00

TOOL REPORT page 4

02/22/87 -- 23:32:16

TAPE	(1.00)	: 0.00	DM_MANAGE_DATA	(10.00)	: 0.00
DM_GRAPHICS	(1.00)	: 0.00	DM_CHARTS	(5.00)	: 0.00
DM_DATA_QUERIES	(1.00)	: 0.00	DM_SHARABILITY	(5.00)	: 0.00
DM_RESPONSE	(1.00)	: 0.00	DM_VIEW_DATA	(5.00)	: 0.00
DM_SECURITY	(1.00)	: 0.00	DM_COMPUTATION	(3.00)	: 0.00
DM_EASY_REPORT	(1.00)	: 0.00	DM_COMPLEX_RPTS	(4.00)	: 0.00
DM_DATA_SUMMARY	(1.00)	: 0.00	DM_TABLES	(1.00)	: 0.00
DM_SMALL	(5.00)	: 0.00	DM_MEDIUM	(5.00)	: 0.00
DM_LARGE	(5.00)	: 0.00	PM_PROJECT_MGMT	(10.00)	: 0.00
PM_SMALL	(7.00)	: 0.00	PM_MEDIUM	(7.00)	: 0.00
PM_LARGE	(7.00)	: 0.00	PM_GRAPHICS	(5.00)	: 0.00
PM_REPORT	(1.00)	: 0.00	PM_MULTIPLE	(1.00)	: 0.00
PM_RISK	(1.00)	: 0.00	PM_TRACKING	(1.00)	: 0.00
PM_LEVELING	(1.00)	: 0.00	PM_ALLOCATE	(1.00)	: 0.00
PM_TYPE	(1.00)	: 0.00	PM_NUMBER	(1.00)	: 0.00
PM_CALENDAR	(1.00)	: 0.00	PM_CPM	(1.00)	: 0.00
PM_PERT	(1.00)	: 0.00	PM_COST	(1.00)	: 0.00
DA_ANALYZE_DATA	(10.00)	: 0.00	DA_STATISTICS	(8.00)	: 0.00
DA_SIMPLE_STAT	(7.00)	: 0.00	DA_ADVANCE_STAT	(8.00)	: 0.00
DA_SPREADSHEET	(10.00)	: 0.00	DA_MACRO	(5.00)	: 0.00
DA_DATA_MODEL	(1.00)	: 0.00	DA_QUALITY_CONT	(1.00)	: 0.00
DA_GRAPHS	(5.00)	: 0.00	DA_CHARTS	(5.00)	: 0.00
DA_REPORTS	(5.00)	: 0.00	DA_FORECASTING	(1.00)	: 0.00
DA_FINANCIAL	(1.00)	: 0.00	D_DOCUMENTS	(10.00)	: 10.00
D_TOC	(1.00)	: 0.00	D_INDEX	(1.00)	: 0.00
D_SUB_SUPER	(1.00)	: 0.00	D_COLUMN	(1.00)	: 0.00
D_FOOTNOTES	(1.00)	: 0.00	D_SPELLER	(1.00)	: 1.00
D_OUTLINE	(1.00)	: 0.00	D_MATH	(1.00)	: 0.00
D_MERGE	(1.00)	: 10.00	D_MACRO	(1.00)	: 10.00
D_FONTS	(5.00)	: 0.00	D_COLOR	(1.00)	: 10.00
D_ASSISTANT	(7.00)	: 0.00	D_GRAPHICS	(4.00)	: 0.00
PG_GRAPHICS	(10.00)	: 0.00	PG_DRAWING	(8.00)	: 0.00
PG_SYMBOLS	(8.00)	: 0.00	PG_MECHANICS	(9.00)	: 0.00
PG_FACILITY	(9.00)	: 0.00	PG_TEXT	(8.00)	: 0.00
PG_CHARTS	(8.00)	: 0.00	PG_SCIENTIFIC	(1.00)	: 0.00
PG_FLOWCHART	(1.00)	: 0.00	PG_PAPER	(1.00)	: 0.00
PG_ONLINE	(8.00)	: 0.00	PG_FONTS	(5.00)	: 0.00
PG_CHARACTER	(5.00)	: 0.00	PG_COLOR	(1.00)	: 0.00
PG_INTERFACE	(1.00)	: 0.00	U_UTILITIES	(10.00)	: 0.00
U_DIR_FILE	(1.00)	: 0.00	U_DISK	(1.00)	: 0.00
U_PRODUCTIVITY	(1.00)	: 0.00			

4 - D4 -- DISPLAYWRITE 3

IBM DisplayWrite 3 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 Ppell 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 3 provides

TOOL REPORT

page 5

02/22/87 -- 23:32:23

4 function math and technical writing support.

CONSULTANT #1: 1, CONSULTANT #2:

BEGINNING	(5.00)	: 8.00		INTERMEDIATE	(5.00)	: 10.00
ADVANCED	(5.00)	: 10.00		BATCH_MODE	(1.00)	: 0.00
ONLINE	(1.00)	: 10.00		VM	(10.00)	: 0.00
MVS	(10.00)	: 0.00		PCDOS	(10.00)	: 10.00
AS	(1.00)	: 0.00		DB2	(1.00)	: 0.00
SAS	(1.00)	: 0.00		SQL	(1.00)	: 0.00
A_DISK	(1.00)	: 0.00		BARCODE_READER	(1.00)	: 0.00
COMPUTER_CARDS	(1.00)	: 0.00		KEYBOARD	(1.00)	: 10.00
OPTICAL_READER	(1.00)	: 0.00		PAGE_READER	(1.00)	: 0.CO
TAPE	(1.00)	: 0.00		DM_MANAGE_DATA	(10.00)	: 0.00
DM_GRAPHICS	(1.00)	: 0.00		DM_CHARTS	(5.00)	: 0.00
DM_DATA_QUERIES	(1.00)	: 0.00		DM_SHARABILITY	(5.00)	: 0.00
DM_RESPONSE	(1.00)	: 0.00		DM_VIEW_DATA	(5.00)	: 0.00
DM_SECURITY	(1.00)	: 0.00		DM_COMPUTATION	(3.00)	: 0.00
DM_EASY_REPORT	(1.00)	: 0.00		DM_COMPLEX_RPTS	(4.00)	: 0.00
DM_DATA_SUMMARY	(1.00)	: 0.00		DM_TABLES	(1.00)	: 0.00
DM_SMALL	(5.00)	: 0.00		DM_MEDIUM	(5.00)	: 0.00
DM_LARGE	(5.00)	: 0.00		PM_PROJECT_MGMT	(10.00)	: 0.00
PM_SMALL	(7.00)	: 0.00		PM_MEDIUM	(7.00)	: 0.00
PM_LARGE	(7.00)	: 0.00		PM_GRAPHICS	(5.00)	: 0.00
PM_REPORT	(1.00)	: 0.00		PM_MULTIPLE	(1.00)	: 0.00
PM_RISK	(1.00)	: 0.00		PM_TRACKING	(1.00)	: 0.00
PM_LEVELING	(1.00)	: 0.00		PM_ALLOCATE	(1.00)	: 0.00
PM_TYPE	(1.00)	: 0.00		PM_NUMBER	(1.00)	: 0.00
PM_CALENDAR	(1.00)	: 0.00		PM_CPM	(1.00)	: 0.00
PM_PERT	(1.00)	: 0.00		PM_COST	(1.00)	: 0.00
DA_ANALYZE_DATA	(10.00)	: 0.00		DA_STATISTICS	(8.00)	: 0.00
DA_SIMPLE_STAT	(7.00)	: 0.00		DA_ADVANCE_STAT	(8.00)	: 0.00
DA_SPREADSHEET	(10.00)	: 0.00		DA_MACRO	(5.00)	: 0.00
DA_DATA_MODEL	(1.00)	: 0.00		DA_QUALITY_CONT	(1.00)	: 0.00
DA_GRAPHS	(5.00)	: 0.00		DA_CHARTS	(5.00)	: 0.00
DA_REPORTS	(5.00)	: 0.00		DA_FORECASTING	(1.00)	: 0.00
DA_FINANCIAL	(1.00)	: 0.00		D_DOCUMENTS	(10.00)	: 10.00
D_TOC	(1.00)	: 0.00		D_INDEX	(1.00)	: 0.00
D_SUB_SUPER	(1.00)	: 10.00		D_COLUMN	(1.00)	: 0.00
D_FOOTNOTES	(1.00)	: 10.00		D_SPELLER	(1.00)	: 10.00
D_OUTLINE	(1.00)	: 10.00		D_MATH	(1.00)	: 10.00
D_MERGE	(1.00)	: 10.00		D_MACRO	(1.00)	: 0.00
D_FONTS	(5.00)	: 0.00		D_COLOR	(1.00)	: 0.00
D_ASSISTANT	(7.00)	: 0.00		D_GRAPHICS	(4.00)	: 10.00
PG_GRAPHICS	(10.00)	: 0.00		PG_DRAWING	(8.00)	: 0.0C
PG_SYMBOLS	(8.00)	: 0.00		PG_MECHANICS	(9.00)	: 0.00
PG_FACILITY	(9.00)	: 0.00		PG_TEXT	(8.00)	: 0.00
PG_CHARTS	(8.00)	: 0.00		PG_SCIENTIFIC	(1.00)	: 0.00
PG_FLOWCHART	(1.00)	: 0.00		PG_PAPER	(1.00)	: 0.00
PG_ONLINE	(8.00)	: 0.00		PG_FONTS	(5.00)	: 0.00
PG_CHARACTER	(5.00)	: 0.00		PG_COLOR	(1.00)	: 0.00

PG_INTERFACE	(1.00)	:	0.00	U_UTILITIES	(10.00)	:	0.00
U_DIR_FILE	(1.00)	:	0.00	U_DISK	(1.00)	:	0.00
U_PRODUCTIVITY	(1.00)	:	0.00				

5 - D5 -- WRITING ASSISTA

IBM Writing Assistant is part of the Assistant series, and accepts data from all other Assistant series programs. It is easy to learn and use, can print to an ASCII file, is uploadable to the host, and is useable with Display-Write 3. It contains a Wordproof spell checker, but does not check for synonyms. This software can automatically add headings, footings, page numbers to each printed page. Writing Assistant can print read, green and blue text on a color printer, and can print envelopes from the address on a letter.

CONSULTANT #1: 2 , CONSULTANT #2: 1

BEGINNING	(5.00)	:	10.00	INTERMEDIATE	(5.00)	:	7.00
ADVANCED	(5.00)	:	5.00	BATCH_MODE	(1.00)	:	0.00
ONLINE	(1.00)	:	10.00	VM	(10.00)	:	0.00
MVS	(10.00)	:	0.00	PCDOS	(10.00)	:	10.00
AS	(1.00)	:	0.00	DB2	(1.00)	:	0.00
SAS	(1.00)	:	0.00	SQL	(1.00)	:	0.00
A_DISK	(1.00)	:	0.00	BARCODE_READER	(1.00)	:	0.00
COMPUTER_CARDS	(1.00)	:	0.00	KEYBOARD	(1.00)	:	10.00
OPTICAL_READER	(1.00)	:	0.00	PAGE_READER	(1.00)	:	0.00
TAPE	(1.00)	:	0.00	DM_MANAGE_DATA	(10.00)	:	0.00
DM_GRAPHICS	(1.00)	:	0.00	DM_CHARTS	(5.00)	:	0.00
DM_DATA_QUERIES	(1.00)	:	0.00	DM_SHARABILITY	(5.00)	:	0.00
DM_RESPONSE	(1.00)	:	0.00	DM_VIEW_DATA	(5.00)	:	0.00
DM_SECURITY	(1.00)	:	0.00	DM_COMPUTATION	(3.00)	:	0.00
DM_EASY_REPORT	(1.00)	:	0.00	DM_COMPLEX_RPTS	(4.00)	:	0.00
DM_DATA_SUMMARY	(1.00)	:	0.00	DM_TABLES	(1.00)	:	0.00
DM_SMALL	(5.00)	:	0.00	DM_MEDIUM	(5.00)	:	0.00
DM_LARGE	(5.00)	:	0.00	PM_PROJECT_MGMT	(10.00)	:	0.00
PM_SMALL	(7.00)	:	0.00	PM_MEDIUM	(7.00)	:	0.00
PM_LARGE	(7.00)	:	0.00	PM_GRAPHICS	(5.00)	:	0.00
PM_REPORT	(1.00)	:	0.00	PM_MULTIPLE	(1.00)	:	0.00
PM_RISK	(1.00)	:	0.00	PM_TRACKING	(1.00)	:	0.00
PM_LEVELING	(1.00)	:	0.00	PM_ALLOCATE	(1.00)	:	0.00
PM_TYPE	(1.00)	:	0.00	PM_NUMBER	(1.00)	:	0.00
PM_CALENDAR	(1.00)	:	0.00	PM_CPM	(1.00)	:	0.00
PM_PERT	(1.00)	:	0.00	PM_COST	(1.00)	:	0.00
DA_ANALYZE_DATA	(10.00)	:	0.00	DA_STATISTICS	(8.00)	:	0.00
DA_SIMPLE_STAT	(7.00)	:	0.00	DA_ADVANCE_STAT	(8.00)	:	0.00
DA_SPREADSHEET	(10.00)	:	0.00	DA_MACRO	(5.00)	:	0.00
DA_DATA_MODEL	(1.00)	:	0.00	DA_QUALITY_CONT	(1.00)	:	0.00
DA_GRAPHS	(5.00)	:	0.00	DA_CHARTS	(5.00)	:	0.00
DA_REPORTS	(5.00)	:	0.00	DA_FORECASTING	(1.00)	:	0.00
DA_FINANCIAL	(1.00)	:	0.00	D_DOCUMENTS	(10.00)	:	10.00

TOOL REPORT page 7

02/22/87 -- 23:32:38

D_TOC	(1.00)	: 0.00		D_INDEX	(1.00)	: 0.00
D_SUB_SUPER	(1.00)	: 0.00		D_COLUMN	(1.00)	: 0.00
D_FOOTNOTES	(1.00)	: 0.00		D_SPELLER	(1.00)	: 8.00
D_OUTLINE	(1.00)	: 0.00		D_MATH	(1.00)	: 0.00
D_MERGE	(1.00)	: 0.00		D_MACRO	(1.00)	: 0.00
D_FONTS	(5.00)	: 0.00		D_COLOR	(1.00)	: 10.00
D_ASSISTANT	(7.00)	: 10.00		D_GRAPHICS	(4.00)	: 7.00
PG_GRAPHICS	(10.00)	: 0.00		PG_DRAWING	(8.00)	: 0.00
PG_SYMBOLS	(8.00)	: 0.00		PG_MECHANICS	(9.00)	: 0.00
PG_FACILITY	(9.00)	: 0.00		PG_TEXT	(8.00)	: 0.00
PG_CHARTS	(8.00)	: 0.00		PG_SCIENTIFIC	(1.00)	: 0.00
PG_FLOWCHART	(1.00)	: 0.00		PG_PAPER	(1.00)	: 0.00
PG_ONLINE	(8.00)	: 0.00		PG_FONTS	(5.00)	: 0.00
PG_CHARACTER	(5.00)	: 0.00		PG_COLOR	(1.00)	: 0.00
PG_INTERFACE	(1.00)	: 0.00		U_UTILITIES	(10.00)	: 0.00
U_DIR_FILE	(1.00)	: 0.00		U_DISK	(1.00)	: 0.00
U_PRODUCTIVITY	(1.00)	: 0.00				

---

6 - D6 -- DWSCRIPT

DWScript is very useful for manuscript preparation and allows many functions for page makeup, composition, and printing. It is very similar to host Script, providing for user specified formats, and advanced formatting capabilities (produces index, appendix, table of contents, headings). A tool named DWMennew is a full screen interface to DWScript.

CONSULTANT #1: 1, CONSULTANT #2:

BEGINNING	(5.00)	: 6.00		INTERMEDIATE	(5.00)	: 10.00
ADVANCED	(5.00)	: 10.00		BATCH_MODE	(1.00)	: 0.00
ONLINE	(1.00)	: 10.00		VM	(10.00)	: 0.00
MVS	(10.00)	: 0.00		PCDOS	(10.00)	: 0.00
AS	(1.00)	: 0.00		DB2	(1.00)	: 0.00
SAS	(1.00)	: 0.00		SQL	(1.00)	: 0.00
A_DISK	(1.00)	: 0.00		BARCODE_READER	(1.00)	: 0.00
COMPUTER_CARDS	(1.00)	: 0.00		KEYBOARD	(1.00)	: 10.00
OPTICAL_READER	(1.00)	: 0.00		PAGE_READER	(1.00)	: 0.00
TAPE	(1.00)	: 0.00		DM_MANAGE_DATA	(10.00)	: 0.00
DM_GRAPHICS	(1.00)	: 0.00		DM_CHARTS	(5.00)	: 0.00
DM_DATA_QUERIES	(1.00)	: 0.00		DM_SHARABILITY	(5.00)	: 0.00
DM_RESPONSE	(1.00)	: 0.00		DM_VIEW_DATA	(5.00)	: 0.00
DM_SECURITY	(1.00)	: 0.00		DM_COMPUTATION	(3.00)	: 0.00
DM_EASY_REPORT	(1.00)	: 0.00		DM_COMPLEX_RPTS	(4.00)	: 0.00
DM_DATA_SUMMARY	(1.00)	: 0.00		DM_TABLES	(1.00)	: 0.00
DM_SMALL	(5.00)	: 0.00		DM_MEDIUM	(5.00)	: 0.00
DM_LARGE	(5.00)	: 0.00		PM_PROJECT_MGMT	(10.00)	: 0.00
PM_SMALL	(7.00)	: 0.00		PM_MEDIUM	(7.00)	: 0.00
PM_LARGE	(7.00)	: 0.00		PM_GRAPHICS	(5.00)	: 0.00
PM_REPORT	(1.00)	: 0.00		PM_MULTIPLE	(1.00)	: 0.00
PM_RISK	(1.00)	: 0.00		PM_TRACKING	(1.00)	: 0.00

TOOL REPORT page 8

02/22/87 -- 23:32:45

PM_LEVELING	(1.00) : 0.00		PM_ALLOCATE	(1.00) : 0.00	
PM_TYPE	(1.00) : 0.00		PM_NUMBER	(1.00) : 0.00	
PM_CALENDAR	(1.00) : 0.00		PM_CPM	(1.00) : 0.00	
PM_PERT	(1.00) : 0.00		PM_COST	(1.00) : 0.00	
DA_ANALYZE_DATA	(10.00) : 0.00		DA_STATISTICS	(8.00) : 0.00	
DA_SIMPLE_STAT	(7.00) : 0.00		DA_ADVANCE_STAT	(8.00) : 0.00	
DA_SPREADSHEET	(10.00) : 0.00		DA_MACRO	(5.00) : 0.00	
DA_DATA_MODEL	(1.00) : 0.00		DA_QUALITY_CONT	(1.00) : 0.00	
DA_GRAPHS	(5.00) : 0.00		DA_CHARTS	(5.00) : 0.00	
DA_REPORTS	(5.00) : 0.00		DA_FORECASTING	(1.00) : 0.00	
DA_FINANCIAL	(1.00) : 0.00		D_DOCUMENTS	(10.00) : 10.00	
D_TOC	(1.00) : 10.00		D_INDEX	(1.00) : 10.00	
D_SUB_SUPER	(1.00) : 0.00		D_COLUMN	(1.00) : 10.00	
D_FOOTNOTES	(1.00) : 0.00		D_SPELLER	(1.00) : 0.00	
D_OUTLINE	(1.00) : 10.00		D_MATH	(1.00) : 0.00	
D_MERGE	(1.00) : 10.00		D_MACRO	(1.00) : 10.00	
D_FONTS	(5.00) : 10.00		D_COLOR	(1.00) : 0.00	
D_ASSISTANT	(7.00) : 0.00		D_GRAPHICS	(4.00) : 0.00	
PG_GRAPHICS	(10.00) : 0.00		PG_DRAWING	(8.00) : 0.00	
PG_SYMBOLS	(8.00) : 0.00		PG_MECHANICS	(9.00) : 0.00	
PG_FACILITY	(9.00) : 0.00		PG_TEXT	(8.00) : 0.00	
PG_CHARTS	(8.00) : 0.00		PG_SCIENTIFIC	(1.00) : 0.00	
PG_FLOWCHART	(1.00) : 0.00		PG_PAPER	(1.00) : 0.00	
PG_ONLINE	(8.00) : 0.00		PG_FONTS	(5.00) : 0.00	
PG_CHARACTER	(5.00) : 0.00		PG_COLOR	(1.00) : 0.00	
PG_INTERFACE	(1.00) : 0.00		U_UTILITIES	(10.00) : 0.00	
U_DIR_FILE	(1.00) : 0.00		U_DISK	(1.00) : 0.00	
U_PRODUCTIVITY	(1.00) : 0.00				

7 - D7 -- DW ASSISTANT

IBM DisplayWrite Assistant a word processing program you can use to create, edit, save, and print documents. It is part of the Assistant Series. DisplayWrite Assistant has the following features:
* Online Help
* Typing Frame resembling a piece of paper; the area in which you enter the text looks like the printed version of the document.
* Spelling Checker and Synonym Lists
* Cursor Draw
* Merge capabilities
* All typical word processing and document formatting fuctions.

CONSULTANT #1: 1 , CONSULTANT #2: 2

BEGINNING	(5.00) : 10.00	INTERMEDIATE	(5.00) : 10.00
ADVANCED	(5.00) : 7.00	BATCH_MODE	(1.00) : 0.00
ONLINE	(1.00) : 10.00	VM	(10.00) : 0.00
MVS	(10.00) : 0.00	PCDOS	(10.00) : 10.00
AS	(1.00) : 0.00	DB2	(1.00) : 0.00
SAS	(1.00) : 0.00	SQL	(1.00) : 0.00

TOOL REPORT page 9

02/22/87 -- 23:32:52

A_DISK	(1.00)	:	0.00	BARCODE_READER	(1.00)	:	0.00
COMPUTER_CARDS	(1.00)	:	0.00	KEYBOARD	(1.00)	:	10.00
OPTICAL_READER	(1.00)	:	0.00	PAGE_READER	(1.00)	:	0.00
TAPE	(1.00)	:	0.00	DM_MANAGE_DATA	(10.00)	:	0.00
DM_GRAPHICS	(1.00)	:	0.00	DM_CHARTS	(5.00)	:	0.00
DM_DATA_QUERIES	(1.00)	:	0.00	DM_SHARABILITY	(5.00)	:	0.00
DM_RESPONSE	(1.00)	:	0.00	DM_VIEW_DATA	(5.00)	:	0.00
DM_SECURITY	(1.00)	:	0.00	DM_COMPUTATION	(3.00)	:	0.00
DM_EASY_REPORT	(1.00)	:	0.00	DM_COMPLEX_RPTS	(4.00)	:	0.00
DM_DATA_SUMMARY	(1.00)	:	0.00	DM_TABLES	(1.00)	:	0.00
DM_SMALL	(5.00)	:	0.00	DM_MEDIUM	(5.00)	:	0.00
DM_LARGE	(5.00)	:	0.00	PM_PROJECT_MGMT	(10.00)	:	0.00
PM_SMALL	(7.00)	:	0.00	PM_MEDIUM	(7.00)	:	0.00
PM_LARGE	(7.00)	:	0.00	PM_GRAPHICS	(5.00)	:	0.00
PM_REPORT	(1.00)	:	0.00	PM_MULTIPLE	(1.00)	:	0.00
PM_RISK	(1.00)	:	0.00	PM_TRACKING	(1.00)	:	0.00
PM_LEVELING	(1.00)	:	0.00	PM_ALLOCATE	(1.00)	:	0.00
PM_TYPE	(1.00)	:	0.00	PM_NUMBER	(1.00)	:	0.00
PM_CALENDAR	(1.00)	:	0.00	PM_CPM	(1.00)	:	0.00
PM_PERT	(1.00)	:	0.00	PM_COST	(1.00)	:	0.00
DA_ANALYZE_DATA	(10.00)	:	0.00	DA_STATISTICS	(8.00)	:	0.00
DA_SIMPLE_STAT	(7.00)	:	0.00	DA_ADVANCE_STAT	(8.00)	:	0.00
DA_SPREADSHEET	(10.00)	:	0.00	DA_MACRO	(5.00)	:	0.00
DA_DATA_MODEL	(1.00)	:	0.00	DA_QUALITY_CONT	(1.00)	:	0.00
DA_GRAPHS	(5.00)	:	0.00	DA_CHARTS	(5.00)	:	0.00
DA_REPORTS	(5.00)	:	0.00	DA_FORECASTING	(1.00)	:	0.00
DA_FINANCIAL	(1.00)	:	0.00	D_DOCUMENTS	(10.00)	:	10.00
D_TOC	(1.00)	:	0.00	D_INDEX	(1.00)	:	0.00
D_SUB_SUPER	(1.00)	:	10.00	D_COLUMN	(1.00)	:	0.00
D_FOOTNOTES	(1.00)	:	0.00	D_SPELLER	(1.00)	:	10.00
D_OUTLINE	(1.00)	:	0.00	D_MATH	(1.00)	:	0.00
D_MERGE	(1.00)	:	10.00	D_MACRO	(1.00)	:	0.00
D_FONTS	(5.00)	:	0.00	D_COLOR	(1.00)	:	10.00
D_ASSISTANT	(7.00)	:	10.00	D_GRAPHICS	(4.00)	:	9.00
PG_GRAPHICS	(10.00)	:	0.00	PG_DRAWING	(8.00)	:	0.00
PG_SYMBOLS	(8.00)	:	0.00	PG_MECHANICS	(9.00)	:	0.00
PG_FACILITY	(9.00)	:	0.00	PG_TEXT	(8.00)	:	0.00
PG_CHARTS	(8.00)	:	0.00	PG_SCIENTIFIC	(1.00)	:	0.00
PG_FLOWCHART	(1.00)	:	0.00	PG_PAPER	(1.00)	:	0.00
PG_ONLINE	(8.00)	:	0.00	PG_FONTS	(5.00)	:	0.00
PG_CHARACTER	(5.00)	:	0.00	PG_COLOR	(1.00)	:	0.00
PG_INTERFACE	(1.00)	:	0.00	U_UTILITIES	(10.00)	:	0.00
U_DIR_FILE	(1.00)	:	0.00	U_DISK	(1.00)	:	0.00
U_PRODUCTIVITY	(1.00)	:	0.00				

8 - D8 -- PROFS

PROFS stands for Professional OFfice System. It offers a wide range of automated office functions:

* document preparation and proofreading, including capabilities to enter,

change, and correct documents, spelling and synonym checking, an aid to correct confusing word usage, phrase checking.
* electronic search and retrieval.
* facilities-scheduling and people-scheduling: personal calendar to schedule appointments and coordinate vacation times.
* electronic reminders
* electronic mail

CONSULTANT #1: 1, CONSULTANT #2:

BEGINNING	(5.00)	:	7.00		INTERMEDIATE	(5.00)	:	10.00
ADVANCED	(5.00)	:	10.00		BATCH_MODE	(1.00)	:	0.00
ONLINE	(1.00)	:	10.00		VM	(10.00)	:	10.00
MVS	(10.00)	:	10.00		PCDOS	(10.00)	:	0.00
AS	(1.00)	:	0.00		DB2	(1.00)	:	0.00
SAS	(1.00)	:	0.00		SQL	(1.00)	:	0.00
A_DISK	(1.00)	:	7.00		BARCODE_READER	(1.00)	:	0.00
COMPUTER_CARDS	(1.00)	:	0.00		KEYBOARD	(1.00)	:	10.00
OPTICAL_READER	(1.00)	:	0.00		PAGE_READER	(1.00)	:	0.00
TAPE	(1.00)	:	0.00		DM_MANAGE_DATA	(10.00)	:	0.00
DM_GRAPHICS	(1.00)	:	0.00		DM_CHARTS	(5.00)	:	0.00
DM_DATA_QUERIES	(1.00)	:	0.00		DM_SHARABILITY	(5.00)	:	0.00
DM_RESPONSE	(1.00)	:	0.00		DM_VIEW_DATA	(5.00)	:	0.00
DM_SECURITY	(1.00)	:	0.00		DM_COMPUTATION	(3.00)	:	0.00
DM_EASY_REPORT	(1.00)	:	0.00		DM_COMPLEX_RPTS	(4.00)	:	0.00
DM_DATA_SUMMARY	(1.00)	:	0.00		DM_TABLES	(1.00)	:	0.00
DM_SMALL	(5.00)	:	0.00		DM_MEDIUM	(5.00)	:	0.00
DM_LARGE	(5.00)	:	0.00		PM_PROJECT_MGMT	(10.00)	:	0.00
PM_SMALL	(7.00)	:	0.00		PM_MEDIUM	(7.00)	:	0.00
PM_LARGE	(7.00)	:	0.00		PM_GRAPHICS	(5.00)	:	0.00
PM_REPORT	(1.00)	:	0.00		PM_MULTIPLE	(1.00)	:	0.00
PM_RISK	(1.00)	:	0.00		PM_TRACKING	(1.00)	:	0.00
PM_LEVELING	(1.00)	:	0.00		PM_ALLOCATE	(1.00)	:	0.00
PM_TYPE	(1.00)	:	0.00		PM_NUMBER	(1.00)	:	0.00
PM_CALENDAR	(1.00)	:	0.00		PM_CPM	(1.00)	:	0.00
PM_PERT	(1.00)	:	0.00		PM_COST	(1.00)	:	0.00
DA_ANALYZE_DATA	(10.00)	:	0.00		DA_STATISTICS	(8.00)	:	0.00
DA_SIMPLE_STAT	(7.00)	:	0.00		DA_ADVANCE_STAT	(8.00)	:	0.00
DA_SPREADSHEET	(10.00)	:	0.00		DA_MACRO	(5.00)	:	0.00
DA_DATA_MODEL	(1.00)	:	0.00		DA_QUALITY_CONT	(1.00)	:	0.00
DA_GRAPHS	(5.00)	:	0.00		DA_CHARTS	(5.00)	:	0.00
DA_REPORTS	(5.00)	:	0.00		DA_FORECASTING	(1.00)	:	0.00
DA_FINANCIAL	(1.00)	:	0.00		D_DOCUMENTS	(10.00)	:	10.00
D_TOC	(1.00)	:	10.00		D_INDEX	(1.00)	:	0.00
D_SUB_SUPER	(1.00)	:	0.00		D_COLUMN	(1.00)	:	0.00
D_FOOTNOTES	(1.00)	:	0.00		D_SPELLER	(1.00)	:	10.00
D_OUTLINE	(1.00)	:	0.00		D_MATH	(1.00)	:	0.00
D_MERGE	(1.00)	:	10.00		D_MACRO	(1.00)	:	0.00
D_FONTS	(5.00)	:	10.00		D_COLOR	(1.00)	:	10.00
D_ASSISTANT	(7.00)	:	0.00		D_GRAPHICS	(4.00)	:	0.00
PG_GRAPHICS	(10.00)	:	0.00		PG_DRAWING	(8.00)	:	0.00
PG_SYMBOLS	(8.00)	:	0.00		PG_MECHANICS	(9.00)	:	0.00
PG_FACILITY	(9.00)	:	0.00		PG_TEXT	(8.00)	:	0.00
PG_CHARTS	(8.00)	:	0.00		PG_SCIENTIFIC	(1.00)	:	0.00
PG_FLOWCHART	(1.00)	:	0.00		PG_PAPER	(1.00)	:	0.00
PG_ONLINE	(8.00)	:	0.00		PG_FONTS	(5.00)	:	0.00
PG_CHARACTER	(5.00)	:	0.00		PG_COLOR	(1.00)	:	0.00
PG_INTERFACE	(1.00)	:	0.00		U_UTILITIES	(10.00)	:	0.00
U_DIR_FILE	(1.00)	:	0.00		U_DISK	(1.00)	:	0.00
U_PRODUCTIVITY	(1.00)	:	0.00					

Appendix C

CASES FOR EXPERIMENTAL STUDIES

Case 1:

Joe has a PC, using it both as a stand-alone PC and as a connection to the host. He 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.

Recommendation:

Source I	Source II	
PC-File+	PC-File3	
	(Consultant)	(ICE)
Accept	Accept	

Case 2:

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

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 package to accomplish this task, and wants to work in the VM environment.

Recommendation:

Source I	Source II
VM/AS	VM/AS
(ICE)	(Consultant)
Accept	Accept

Case 3:

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

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.

Recommendation:

Source I	Source II
VM/AS, SQL	VM/AS, SQL
(Consultant)	(ICE)
Accept	Accept

Case 4:

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 finan- cial 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.

Recommendation:

Source I	Source II
Lotus 123	Lotus 123, Tiny Calc
(Consultant)	(ICE)
Accept	Accept

Case 5:

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 process- ing 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 finan- cial issues: to perform "what-if" analyses.

When the analyses are finished, she must 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.

Recommendation:

Source I	Source II
Tiny Calc, Lotus 123	Lotus 123, Tiny Calc
(Consultant)	(ICE)
Accept	Accept

Case 6:

Tom uses a PC, both as a stand-alone and as a connection to the host machine. He has recently been assigned to a work group with the marketing division, where he must compile and summarize the statistics from data collected about customer satisfaction.

The majority of Tom's tasks involve the following: * Fairly complex analyses, including principal com- ponents analysis, multiple analysis of variance (MANOVA), multiple regression, and factor analysis. * Integration of the results into a report that includes graphical representation of the data.

Because of the complexity of the analyses, and the huge volume of data involved, Tom has been advised to work in a batch mode environment so the jobs can be submitted to the host during the evening hours when there is less demand from other users.

Recommendation:

Source I	Source II
SAS	VM/AS, Lotus 123
(ICE)	(Consultant)
Accept	Reject

Case 7:

Nathan has been asked to conduct a study where the final goal is to make recommendations on budget priorities. Individuals in this department of ten people have been asked to submit budget proposals for the coming year. The proposals must include a breakdown of required monies in 5 specified areas.

Nathan is to analyze these budgets, asking "what-if" questions to see how the total budget should best be allocated. The proposals are weighted according to value to the company, and each of the 5 areas of the individual proposals are examined in appropriating the funding. He is not sure if the analysis would be made easier if he could write his own macros.

He prefers to use the PC environment, but is most interested in having the best tool for the job, so will be willing to work in either one. His work is all on-line. He has used the PC for simple programming and report writing in the past, and rates his skills as average.

Recommendation:

Source I	Source II
Lotus 123	VM/AS
(Consultant)	(ICE)
Accept	Reject

Case 8:

Lyle has been a VM hacker since he joined the firm. He has in the past been involved primarily at the assembly language level.

His current need is for a tool that will allow manipula- tion and analyses of matrices of data – extremely LARGE volumes of data. He also needs to be able to write routines that will display this data in tables and charts.

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 soft- ware.

Recommendation:

Source I	Source II
VM/AS	SAS, VM/AS
(Consultant)	(ICE)
Accept	Accept

Case 9:

Judy needs to prepare monthly reports of sales data. The data is brought to her on PC disks from 20 people who use IBM Filing Assistant.

The reports go to the manager, who requires that certain graphs be made each month to accompany the textual summaries of the data.

Judy's previous experience with computers is limited to the PROFS environment (to take care of daily schedules and mail) and to document preparation, for which she has experi- mented with various PC software available.

Recommendation:

Source I	Source II
PC-File+, Graphing Assistant	Graphing Assistant
(ICE)	(Consultant)
Accept	Accept

Case 10:

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.

Recommendation:

Source I	Source II
ChartMaster, SignMaster	ChartMaster, SignMaster
(ICE)	(Consultant)
Accept	Accept

Case 11:

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.

Recommendation:

Source I	Source II
APGS	APGS
(ICE)	(Consultant)
Accept	Accept

Case 12:

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.

Recommendation:

Source I	Source II
SignMaster,	SignMaster,
PersonalExpression	ChartMaster
(Consultant)	(ICE)
Accept	Accept

Case 13:

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 sizes

* 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.

Recommendation:

Source I	Source II
APGS	APGS
(Consultant)	(ICE)
Accept	Accept

Case 14:

Sue began using the computer two years ago to prepare documents with PROFS. She has a PC on her desk, with a connec- tion 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 is includes both alphanumeric text and graphics, so she requires the following capabilities:

* different fonts
* varying character sizes
* color
* library of predefined 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.

Recommendation:

Source I	Source II
APGS	APGS
(ICE)	(Consultant)
Accept	Accept

Case 15:

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 process- ing.

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.

Recommendation:

Source I	Source II
PROGS	PROFS
(Consultant)	(ICE)
Accept	Accept

Case 16:

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 as much time as needed learning the package, and must work in the PC environment.

Recommendation:

Source I	Source II
DW3/DWAssistant,	DW3, WordProof
	WordProof
(ICE)	(Consultant)
Accept	Accept

Case 17:

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.

Recommendation:

Source I	Source II
DW3/ProfesEdit	DW3, WordProof
	WordProof
(ICE)	(Consultant)
Accept	Accept

Case 18:

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.

Recommendation:

Source I	Source II
WordProof,	Writing Assistant,
Writing Assistant	WordProof
(Consultant)	(ICE)
Accept	Accept

Case 19:

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 a 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.

Recommendation:

Source I	Source II
VM/AS	Harvard Total Project Manager
(ICE)	(Consultant)
Reject	Accept

Case 20:

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.

Recommendation:

Source I	Source II
VM/AS	VM/AS
(ICE)	(Consultant)
Accept	Accept

LIST OF REFERENCES

1. Atre, Shaku. "The Information Center Holds," *PC World*, August 1986, 156-163.

2. Barber, G. "Supporting Organizational Problem Solving with a Work Station." *ACM Transactions on Office Information Systems, 1* (1983) 1, pp. 45-67.

3. Bender, Paul S. *Resource Management, An Alternative View of the Management Process*. NY: John Wiley and Sons, 1983.

4. Benjamin, R. I. "Information Technology in the 1990s: A Long-Range Planning Scenario," *MIS Quarterly*, June 1982, pp. 11-31.

5. Biggerstaff, Ted and Charles Richter. "Reusability Frame- work, Assessment, and Directions," *IEEE Software*, March 1987, pp. 41-49.

6. Bimson, Kent and Linda Boehm Burris. "The Craft of Engineering Knowledge: Some Practical Insights," *Proceedings of the Twentieth Annual Hawaii International Conference on System Sciences, 1987*, pp. 460-469.

7. Blanning, Robert. "Expert System for Management: Possible Application Areas," in *Transactions of Fourth International Conference on Decision Support Sciences*, Dallas, Texas, April 1-4, 1984, pp. 69-77.

8. Blanning, Robert, "Sensitivity Analysis in Expert Systems for Management," *AFIPS Proceedings of the National Computer Conference* (Vol. 53), 1984, pp. 489-495.

9. Blanning, Robert W. "Issues in the Design of Expert Systems for Management." Working Paper: Owen Graduate School of Management, Vanderbilt University, Nashville, Tennessee, 1984a.

10. Blanning, Robert W. "Management Applications of Expert Systems," *Information and Management*, Volume 7, No. 6, December, 1984b, pp. 311-316.

11. Bohanek, M., I. Bratko, and V. Rajkovic. "An Expert System for Decision Making." In H. G. Sol (ed.) *Processes and Tools for Decision Support.* Amsterdam: North-Holland, 1983, pp. 235-248.

12. Bohl, Don Lee, (ed.) The AMA Report on Information Centers, American Management Association Membership Publications Division, New York, 1986.

13. Boose, John H. "ETS: A System for the Transfer of Human Expertise," in *Knowledge Based Problem Solving*, Janusz S. Kowalik (ed.), Prentice-Hall, 1986a, pp. 68-111.

14. Boose, John H. *Expertise Transfer for Expert System Design.* NY: Elsevier, 1986b.

15. Boose, John H. and J. M. Bradshaw. "A Knowledge Acquisition Workbench for Eliciting Decision Knowledge," *Twentieth Annual Hawaii International Conference on System Sciences*, 1987, pp. 460-469.

16. Bouwman, M.J. "Human Diagnostic Reasoning by Computer: An Illustration from Financial Analysis," *Management Science*, 29 (1983) 6, pp. 653-672.

17. Brancheau, James C., Douglas R. Vogel, and James C. Wetherbe. "An Investigation of the Information Center from the User's Perspective," *Data Base* (Vol. 17, No. 1), Fall, 1985, pp. 4-17.

18. Bruner, Jerome. *The Process of Education.* 1960, Cambridge: Harvard University Press.

19. Buchanan, B. G., D. Barstow, R. Bechtel, J. Bennet, W. Clancey, C. Kulikowski, T. M. Mitchell, and D. A. Waterman. "Constructing an Expert System,' in *Building Expert Systems*, (Eds. Hayes-Roth, Waterman and Lenat), Reading, MA: Addison-Wesley Publishing Co., 1983, pp. 127-167.

20. Butler, Keith, and James E. Corter. "Use of Psychometric Tools for Knowledge Acquisition: A Case Study," *Tools for Knowledge Acquisition*, pp. 295-319.

21. Canning, Barbara, (ed.) "Creating an Information Center Strategy," *EDP Analyzer*, Volume 25, Number 2, February 1987.

22. Canning, R. G., (ed.) "Supporting the End User Programmer," *EDP Analyzer*, Volume 19, Number 6, June 1981.

23. Canning, R. G. (ed.) "Coping With End User Computing," *EDP Analyzer*, Volume 22, Number 2, February, 1984.

24. Canning, Richard G. (ed.) "What's Happening With Expert Systems?" *EDP Analyzer*, December 1985, Vol. 23, No. 12.

25. Carlson, Walter M. "Business Information Analysis and Integration Technique (BIAIT) - The New Horizon," *Data Base*, Vol. 10, No. 4, Spring 1979, pp. 3-19.

26. Carr, Houston H. "Information Centers: The IBM Model vs. Practice," *MIS Quarterly*, Vol. 11, No. 3, September 1987, p. 325-338

27. Clancey, W. J. "Classification Problem Solving," *Proceedings of the National Conference on AI, AAAI, 1984.*

28. Clarkson, G. P. E. "A Model of the Trust Investment Process" in E. A. Feigenbaum and J. Feldman (eds.), *Computers and Thought.* NY: McGraw-Hill, 1963, pp. 347-371.

29. Clifford, Jim, Matthias Jarke, and Yannis Vassiliou. "A Short Introduction to Expert Systems," Working Paper: Computer Applications and Information Systems Area, Graduate School of Business Administration, New York University, 1982.

30. Cohen, P. and M. D. Lieberman. "A Report on FOLIO: An Expert Assistant for Portfolio Managers," *Proceedings IJCAI- 83*, 1983, pp. 212-214.

31. Cohen, P. R.; Feigenbaum, E. A. *The Handbook of Artificial Intelligence*, Vol. 3, Pitman, 1982.

32. Coombs, M. J. and J. L. Alty. "Expert Systems: An Alternative Paradigm," *International Journal of Man-Machine Studies*, 1984, 20, p. 21-43.

33. Davis, R. "Interactive Transfer of Expertise: Acquisition of New Inference Rules," *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, pp. 321-328.

34. Dickson, Gary W., Robert L. Leitheiser, Mal Nechis, and James C. Wetherbe. "Key Information Systems Issues for the 1980's," *MIS Quarterly*, Volume 8, Number 3, September 1985, pp. 135-148.

35. Duda, Richard and Edward Shortliffe. "Expert Systems Research," *Science*, Vol. 220, No. 4594, April 15, 1983, pp.

36. Dungan, C. W. "A Model of Audit Judgement in the Form of an Expert System." Ph.D. Dissertation, University of Illinois, 1983.

37. Eshelman, Larry and John McDermott. "MOLE: A Knowledge Acquisition Tool That Uses Its Head," *Proceedings of 5th AAAI Conference*, AAAI, Aug 11-15, 1986, pp. 950-955.

38. Feigenbaum, E. A. and McCorduck, P. *The Fifth Generation Pan Books*, London, 1984.

39. Fellers, Jack. "Key Factors in Knowledge Acquisition," Working Paper, Operations and Systems Management, Graduate School of Business, Indiana University, Bloomington, IN, March 1987.

40. Fikes, R. E. "Odyssey: A Knowledge-Based Assistant," *Artificial Intelligence, 16* (1981) 3, pp. 331-361.

41. Gaschnig, John G. *Annotated Bibliography about Propsector*, Menlo Park, CA: SRI International, Feb. 1981.

42. Goldstein, Ira P. and Roberts, Bruce. "Using Frames in Scheduling," in *Artificial Intelligence: An MIT Perspective*, Patrick Henry Whinston and Richard Henry Brown (Eds.), MIT Press, Cambridge, (1982), Vol. 1, pp. 257-284.

43. Gulden, G. K. and E. S. Arkush. "Developing a Strategy Profile for Management Support Systems," *Proceedings of the 1983 National Computer Conference*, AFIPS Press, p. 415-420

44. Hammond, L. W. "Management Considerations for an Information Center," *IBM Systems Journal*, Vol. 21, No. 2, 1982, pp. 131-161.

45. Harmon, P. and King, D. *Expert Systems: Artificial Intelligence in Business*, NY: Wiley, 1985.

46. Hart, Anna. "The Role of Induction in Knowledge Elicitation,'; *Expert Systems*, January 1985, Vol. 2, No. 1, pp. 24-28.

47. Hart, Anna. *Knowledge Acquisition for Expert Systems*, Kogan-Page, 1986.

48. Hayes-Roth, Frederick. "The Knowledge-Based Expert System: A Tutorial," *Computer*, September 1984, pp. 11-28.

49. Heltne, Mari, Ajay S. Vinze, Benn Konsynski, J. F. Nunamaker, Jr. "A Consultation System for Information Center Resource Allocation," *Proceedings of the 1987 ACM SIGBDP- SIGCPR Conference*, Elias Awad (ed.) March 5-6, 1987, pp. 20-44.

50. IBM. Expert System Consultation Environment and Expert System Development Environment: Reference Manual, IBM, Order Number SH20-9609-2, 1986.

51. Kelly, George A. *The Psychology of Personal Constructs*, Norton, NY, 1955.

52. Kendrick, John W. "Productivity Trends and the Recent Showdown: Historical Perspective, Causal Factors, and Policy Options," in *Contemporary Economic Problems*. Washington, D.C., American Enterprise Institute, 1979.

53. Konsynski, Benn. "Advances in Information Systems Design," *Journal of Management Information Systems*, Winter 1984-85, Vol. 1, No. 3, pp. 5-32.

54. Krcmar, Helmut A. O. "Enterprise-Wide Information Management: Expert Systems for Information Management." *IBM Los Angeles Scientific Center*, Report Number G320-2767, July 1985.

55. Leitheiser, Robert and Wetherbe, James. "The Successful Information Center: What Does It Take to Make One?" Working Paper Series, MISRC-WP-85-14, January 1985, Graduate School of Management, University of Minnesota, Minneapolis, MN.

56. Lodge, Lorne K. "How IBM Uses Technology To Manage Produc- tivity," *Business Quarterly*, Winter 1983, pp. 159-162.

57. Luconi, Fred L., Thomas W. Malone and Michael S. Scott Morton. 11Expert Systems: The Challenge for Managers," *Sloan Management Review*, Summer 1986, pp. 3-14.

58. McFarlan, F. Warren, (ed.) The Information Systems Research Challenge, *Harvard Business School Press*, Boston, MA, 1984.

59. McKenney, James L., Warren McFarlan, and Philip Pyburn. "The Information Archipelago–Maps and Bridges," *Harvard Business Review*, Sept-Oct., 1982; p 109ff.

60. Michaelsen, Robert H., Donald Michie, and Albert Boulanger. "The Technology of Expert Systems," *Byte*, April, 1985, pp. 303-312.

61. Michalski, R. S., J. Carbonell, and T. Mitchell, Eds. *Machine Learning: An Artificial Intelligence Approach*. Los Altos, CA., Tioga Publishing Co., 1983.

62. Michie, D.; Johnston, R. *The Creative Computer*, Pelican, England, 1985.

63. Nau, Dana S. "Expert Computer Systems," *Computer*, February 1983, pp. 63-83.

64. Nolan, R. L. and C. F. Gibson. "Managing the Four Stages of EDP Growth," *Harvard Business Review*, January-February 1974, pp. 76-88.

65. Prieto-Diaz, Ruben and Peter Freeman. "Classifying Software for Reusability," *IEEE Software*, January 1987, pp. 6-16.

66. Reggia, James A., Nau, Dana S. and Wang, Pearl Y. "Diagnostic Expert Systems Based on a Set Covering Model," in *Developments in Expert Systems*, edited by M. J. Coombs, Academic Press Inc., Orlando, FL., 1984 pp. 35-58.

67. Robinson, Lewis. Speech Transcript, International Conference on Information Systems, Tucson, Arizona, November 1984.

68. Rockart, John and Flannery, Lauren. "The Management of End User Computing," *Communications of the ACM*, October 1983, Vol. 26, No. 10, pp. 776-784.

69. Rowe, Alan, Richard O. Mason, and Karl Dickel. *Strategic Management and Business Policy*, Addison-Wesley, 1985.

70. Royksund, Conrad, Mari Heltne, and J. F. Nunamaker, Jr. "Critical Success Factors in Information Center Stragegy," *Proceedings of the 1988 ACM-SIGCPR International Conference*, April 1988.

71. Santarelli, Mary-Beth. "Waiting for A.I." *Information Center*, Sept. 1985, pp. 23-27.

72. Scott Morton, Michael S. "The State of the Art Research," in *The Information Systems Research Challenge*, edited by F. Warren McFarlan, Harvard Business School Press, Boston, MA., 1984, pp. 13-41.

73. Shaw, M. L. G., (ed.) *Recent Advances in Personal Construct Technology*. NY: Academic Press, 1981.

74. Shaw, Mildred L. G. "Conversational Heuristics for Eliciting Shared Understanding," from *Recent Advances in Personal Construct Technology*, edited by Mildred L. G. Shaw, NY:Academic Press, 1981, p. 31-44.

75. Shortliffe, E. H. *Computer-Based Medical Consultations: MYCIN*. NY: Elsevier, 1976.

76. Slagle, James R. and Henry Hamburger. "An Expert System for a Resource Allocation Problem," *Communications of the ACM*, September 1985, Vol. 28, No. 9, pp. 994-1004.

77. Stearns, E. W. "The Information Center: The Best Answer?" *Infosystems*, June 1984, pp. 46-48.

78. Sviokla, John. "Business Implications of Knowledge-Based Systems," *Data Base*, Fall 1986, Vol. 18, No. 1, pp. 5-16.

79. Tanimoto, Steven L. *The Elements of Artificial Intelligence*. Rochville, MD: Computer Science Press, 1987.

80. Turing, A. M. "Computing Machinery and Intelligence," MIND, October 1950, reprinted in *Creative Computing*, Vol. 6, No. 1, January 1980, pp. 44-53.

81. Van Gigch, John P. *Applied General Systems Theory*, 2nd edition NY: Harper and Row, 1978.

82. Vinze, Ajay S., Mari Heltne, Minder Chen, J. F. Nunamaker, Benn Konsynski. "A Knowledge-Based Approach to Resource Management," *Proceedings of the Eighth International Conference on Information Systems*, December 6-9, 1987.

83. Wallace, Fred. *Information Center*, IBM Corporation, Tucson, AZ. Interview with author.

84. Waterman, Donald A. *A Guide to Expert Systems*, Reading, MA: Addison-Wesley Publishing, 1986.

85. Webster, Robin and Leslie Miner. "Expert Systems: Programming Problem-Solving," *Technology*, January/February, 1982, pp. 62-73.

86. Wicklund, Gary and Roth, Roberta. "Expert Systems in Insurance Underwriting: Model Development and Application," *Proceedings of the 1987 ACM SIGBDP-SIGCPR Conference*, March 5-6, 1987, pp. 129-139.

87. Wright, J. R., Miller, F.D., Otto, G.U.E, Siegfried, E.M., Vesonder, G.T., and Zielinski, J. E. "ACE: Going From Prototype to Product with an Expert System," *ACM Conference Proceedings*, October 1985.