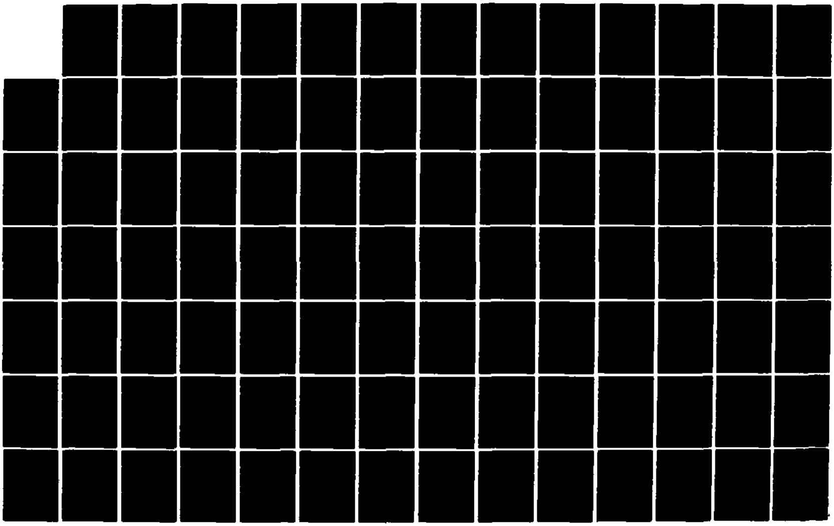
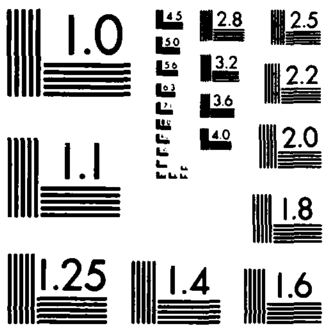
AD-A132 998 THE DETERMINATION OF USER INFORMATION REQUIREMENTS DURING THE DEVELOPMENT OF MANAGEMENT INFORMATION SYSTEMS(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA P R GARDELLA JUN 83 1/2

UNCLASSIFIED F/G 5/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

THE DETERMINATION OF USER INFORMATION REQUIREMENTS
DURING THE DEVELOPMENT OF
MANAGEMENT INFORMATION SYSTEMS

by

Paul Richard Gardella, Jr.

June, 1983

Thesis Advisor: Gordon C. Howell

83

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A132998	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) The Determination of User Information Requirements During the Development of Management Information Systems		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis June, 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Paul Richard Gardella, Jr.		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June, 1983
		13. NUMBER OF PAGES 136
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release, Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

User Information Requirements, Information Requirements, User Requirements, User Needs, Information Needs

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

One of the major causes for the failure of Management Information Systems (MIS) is that they do not satisfy the users' information requirements This, in turn, is most often caused by the fact that those requirements are difficult to obtain accurately and completely. Simply "asking" the user what he needs is inadequate. This thesis reviews the Information Requirements Analysis (IRA) literature, briefly describing some of the techniques available for determining the users' information requirements. It then reports on a survey which attempted to investigate the degree to which the

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ABSTRACT (Continued) Block # 20

extenxive MIS literature involving information requirements determination has had practical impact on the way in which MIS's are actually developed.

The Determination of User Information Requirements During the Development of Management Information Systems

by

Paul Richard Gardella, Jr.
Lieutenant, United States Navy
B.S., Villanova University, 1978

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
June 1983

Author: Paul R. Gardella Jr.

Approved by: ______________________
Thesis Advisor

____
Second Reader

____
Chairman, Department of Administrative Sciences

____
Dean of Information and Policy Sciences

ABSTRACT

One of the major causes for the failure of Management Information Systems (MIS) is that ~~they~~ these do not satisfy the users' information requirements. This, in turn, is most often caused by the fact that those requirements are difficult to obtain accurately and completely. Simply "asking" the user what he needs is inadequate. This thesis reviews the Information Requirements Analysis ~~(IRA)~~ literature, briefly describing some of the techniques available for determining the users' information requirements. It then reports on a survey which attempted to investigate the degree to which the extensive MIS literature involving information requirements determination has had practical impact on the way in which MIS's are actually developed.

TABLE OF CONTENTS

PART I INTRODUCTION, DESCRIPTION OF PROBLEM, AND TECHNIQUES ---- 9

I. INTRODUCTION ---- 10

II. THE INFORMATION REQUIREMENTS DETERMINATION PROBLEM ---- 18

A. COMPLEX PATTERNS OF INTERACTION ---- 19

B. VARIETY AND COMPLEXITY OF INFORMATION REQUIREMENTS ---- 21

C. CONSTRAINTS ON HUMANS AS INFORMATION PROCESSORS ---- 23

III. THE BASICS ---- 29

A. INTERVIEWING ---- 29

B. QUESTIONNAIRES ---- 32

C. DIRECT OBSERVATION ---- 33

D. DOCUMENT EXAMINATION ---- 34

E. MEASUREMENT ---- 36

F. TOP-DOWN VS. BOTTOM-UP APPROACH TO REQUIREMENTS ANALYSIS ---- 36

G. SUMMARY ---- 38

IV. EARLY TECHNIQUES ---- 40

A. ASK AND ANALYZE ---- 40

B. FUNCTIONAL ANALYSIS ---- 41

V. INFORMATION ANALYSIS ---- 47

A. DATA ANALYSIS ---- 47

B. DECISION ANALYSIS ---- 48

	C. DATA VS. DECISION ANALYSIS	49
	D. PROTOCOL ANALYSIS	51
	E. INFORMATION ENVIRONMENTAL ANALYSIS	51
VI.	GROUP TECHNIQUES	53
	A. BRAINSTORMING	54
	B. DELPHI METHOD	55
	C. NOMINAL GROUP TECHNIQUE	59
VII.	OTHER APPROACHES	62
	A. CRITICAL SUCCESS FACTORS APPROACH	62
	B. SIMULATION	65
	1. Paper Simulation	65
	2. Human Simulation	66
	3. System Simulation	67
	C. DEFINEPAC	69
	D. CRITICAL INCIDENT TECHNIQUE	72
	E. DATA BASE APPROACH	73
VIII.	ITERATIVE DESIGN TECHNIQUES	75
	A. PROTOTYPING	76
	B. HEURISTIC DEVELOPMENT	77
	C. EVALUATION OF ITERATIVE DESIGN	78
IX.	SELECTION METHODS	82
	A. SITUATIONAL PERSPECTIVE	82
	B. CONTINGENCY APPROACH	87
X.	USER SELF-DETERMINATION OF NEEDS	92
	A. USER PROJECT TEAMS	92

	B.	"TROJAN HORSE" STRATEGY	93
	C.	INFORMATION CENTER	94
PART II		IRD TECHNIQUE SURVEY AND CONCLUSIONS	97
XI.		PRACTICAL APPLICATIONS	98
	A.	RESULTS OF THE INTERVIEWS	99
XII.		EVALUATION OF THE SURVEY	108
	A.	COMMUNICATIONS	108
	B.	PARTICIPANTS	111
	C.	SYSTEM TYPES	112
	D.	RESEARCH PRACTICES	112
	E.	ALTERNATE METHODS	113
	F.	SUMMARY	115
XIII.		CONCLUSIONS	116
APPENDIX A		ASPECTS OF TAGGART AND THARP'S IRA TECHNIQUE FRAMEWORK	119
APPENDIX B		CHARACTERISTICS OF PARTICIPANTS	121
APPENDIX C		INTERVIEW QUESTIONS	123
APPENDIX D		TECHNIQUES DESCRIBED IN INTERVIEWS	125
APPENDIX E		TABULATION OF RESPONSES TO SURVEY QUESTION #14	127
LIST OF REFERENCES			129
INITIAL DISTRIBUTION LIST			135

LIST OF FIGURES

FIGURE	7-1	DEFINEPAC Framework for Decision Modeling	70
FIGURE	9-1	MIRA Technique Ratings	83
FIGURE	9-2	Steps in Selection of a MIRA Technique	84
FIGURE	9-3	Selection of an IRD Strategy	91

PART I

INTRODUCTION, DESCRIPTION OF PROBLEM, AND TECHNIQUES

I. INTRODUCTION

When computers first came on the scene, they were used almost solely to perform clerical tasks, for example, tabulating a census, performing complex scientific calculations, processing sales orders, and logging transactions such as those associated with accounts receivable and accounts payable. As the technology evolved, it became evident that the computer had the ability to do more than just perform such clerical tasks; it could extract *information* from data and present this information to managers in such a way as to assist these managers in performing their jobs. Hence, the birth of what are often called *Management Information Systems* (MIS). The MIS concept created quite a stir in data processing circles at first because of the fantastic potential it held for revolutionizing the way business was done and decisions were made. When the initial smoke cleared, however, it became sadly apparent that MIS had not achieved its potential. [Refs. 1,2] The managers which these information systems were designed to serve just did not find their outputs as useful as had once been expected.

What went wrong? Most authorities on the subject describe causes which essentially fall into one of three basic categories:

(1) Managers simply expected too much initially because they did not really understand the capabilities and limitations of computers. These expectations were undoubtedly spawned, at least in part, by over-enthusiastic data processing (DP) professionals who went overboard in describing the "amazing things" their machines could do.

(2) In the course of trying to ensure that the manager had all the information he needed, and possibly to justify their own existence, DP personnel flooded the manager with so much data that he had not the time nor the patience to sift through it all in search of the small amount of relevent information. [Refs. 1,3] This led to managerial frustration and disgust with MIS.

(3) Perhaps the most commonly accepted cause for this "MIS potential-realization gap" [Ref. 2: p.231] is that not enough attention was paid to the proper content of the information system during the development process. [Ref. 4] In other words, the systems were simply not providing the managers with the information they really needed. Taggart and Tharp discuss a national survey conducted by researchers at Colorado State University in 1975 which pointed out that the identification of information needs of management can be considered the most critical factor associated with successful MIS implementation second only to the definition of system objectives. [Ref. 2: p.231] Dhar and Davis charge that the information provided to managers was often

incorrect, inadequate, inconsistent, ambiguous, or unavailable. [Ref. 5: p.191] Dr. Gordon B. Davis of the University of Minnesota and the Management Information Systems Research Center, one of the foremost figures in the field, agrees. "The analysis of information needs has always been one of the most significant problems in information systems design and implementation...." [Ref. 6: p.41] Numerous examples of MIS development and implementation efforts which have failed due to an inability to meet the users' needs are present in the literature. [Refs. 7,8]

What can be done about this pervasive problem? The fields of Information Requirements Analysis (IRA) and, more specifically, Information Requirements Determination (IRD) have arisen to attempt to answer this question. (In practice, these two terms are used interchangeably, and will be used that way in this paper, also.) IRA seeks to discover the nature of the information needs problem and to develop techniques or methodologies for overcoming it.

Before delving too deeply into IRA, it would be useful to clarify some of the terminology which will be encountered during any discussion of this area of research.

Despite the fact that MIS's have been around for over two decades, there is still no clear agreement on the answer to the question "What is a Management Information System?" Each author advances his own definition at the start of his writing to clarify <u>his</u> use of the term, so I shall do the same. In this paper, a Management Information System (MIS) shall refer to any system designed to provide one or more managers, at any level in the organization, with information to support managerial decision making. Strictly speaking, an MIS could be manual or computer-based, but in this paper the latter flavor is assumed. Any business data processing activity which is not an MIS is a TRANSACTION PROCESSING SYSTEM which performs a clerical or recordkeeping task rather than providing information. Payroll, accounts receivable, sales order processing, and similar activities are examples of Transaction Processing Systems. The reader will undoubtedly notice that the MIS concept defined here encompasses a huge area of data processing. For this reason, it is helpful to categorize MIS, using Robert N. Anthony's framework [Ref. 9], into information systems supporting (1) Operational level decisions, such as those encountered in a manufacturing environment where the manager is concerned with control of the efficiency and effectiveness with which a task is accomplished (I shall refer to systems in this category as OPERATIONAL MIS); (2) Management level decisions, somethimes referred to as "tactical" planning or control, such as those made by managers when allocating or monitoring the status and use of organizational resources (I shall refer to systems in this category as TACTICAL MIS); and (3) Strategic planning decisions, which are generally made in connection with organizational objectives as well as the resources used to attain these objectives and the policies that are to govern the acquisition, use, and disposition of these resources [Ref. 9: p.24] (this will be called STRATEGIC MIS). The boundaries between these categories are fuzzy, but they are nonetheless useful.

The decisions which managers must make at these three levels can be either STRUCTURED or UNSTRUCTURED. Herbert A. Simon first discussed this concept in 1960. [Ref. 10] Using slightly different terminology, Simon describes structured decisions as those that are repetitive and routine, to the extent that a definite procedure has been worked out for handling them so that they don't have to be treated <u>de nova</u> each time they occur. Unstructured decisions, on the other hand are those for which there is no cut-and-dried method of handling the problem because it hasn't arisen before, or because its precise nature and structure are elusive or complex, or because it is so important that it deserves a custom tailored treatment. Further, an unstructured decision problem calls for a response where the system has no specific procedure to deal with situations like the one at hand, but must fall back on whatever <u>general</u> capacity it has for intelligent, adaptive, problem-oriented action. [Ref. 10: pp.5-6] The distinction between structured and unstructured decisions is necessary because the type of information required for each is different and the proper matching of information type with decision type is an essential ingredient to the success of an MIS. [Ref. 11]

The emphasis of this paper is on the USER, but who in fact are the users of an MIS? Quite simply, the users of a management information system are those managers in the organization to whom the outputs of the system are directed for decision-making purposes. [Ref. 12: p.131] Since only managers are considered users of an MIS, I shall often use the terms "manager" and "user" interchangeably.

USER INFORMATION REQUIREMENTS refer to any and all elements of information required or desired by the manager in fulfilling his managerial tasks, expressed in terms of the content, scope, quality, accuracy, and timeliness of the information required. [Ref. 12: p.132] I would hasten to add "format" to this list. Some authors, notably Ackoff [Ref. 1], point out that the manager does not really need all the information he wants; much of that requested by managers is desired because they do not understand what variables actually affect the outcome of the situation being considered, so they want everything. But they do not know precisely which information they really need and neither does anyone else, so, until a satisfactory model of the decision situation is developed, the manager does in effect need everything he wants.

Equivalent terms used in place of User Information Requirements are INFORMATION REQUIREMENTS, USER REQUIREMENTS, USER NEEDS, and INFORMATION NEEDS.

In an attempt to keep this paper from trespassing into the technical realm of systems analysis and software engineering, it is necessary to distinguish between user requirements and <u>system specifications</u>. Systems specifications are the technical translation of information requirements into required output and minimum standards of performance for the software used to implement the MIS. The difference is also one of perspective: User requirements are written with the user in mind while system specifications consider the programmer.

As mentioned previously, Information Requirements Analysis is the field of research through which information system specialists hope to gain an understanding of the information requirements determination problem thereby enabling the construction and successful implementation of techniques for accurately eliciting the information needs of managers. Although no techniques have been conclusively proven effective, several have been developed and successfully implemented under experimental conditions.

In light of the results of IRA research, this thesis has the broad objective of investigating the degree to which the extensive MIS literature involving information requirements determination (IRD) has had practical impact on the way in which MIS's are actually developed.

More specifically, the study will:

(1) present and discuss the IRD problem and techniques borne of the IRA research which has been conducted up to the present,

(2) identify the techniques and methods of IRD currently being used by MIS professionals in industry, and

(3) attempt to draw some conclusions about the relationship between IRA research and application.

To achieve these objectives, the paper will be divided into two parts. Part I, to which this chapter belongs, deals with the introduction to and description of the IRD problem. Following that, some of the IRD techniques proposed in the literature are presented. Part II discusses the results of a fifteen-organization survey which attempted to identify the practical application of these IRD techniques and draws some conclusions. It must be pointed out that due to time and resource constraints, the survey of industry is not adequate for a valid statistical analysis but rather, was intended to provide a learning experience for the student and a "rough" indication of the state of the art in current MIS development practices with respect to information requirements determination.

II. THE INFORMATION REQUIREMENTS DETERMINATION PROBLEM

It was mentioned in the last chapter that the main reason many MIS's fail to perform as expected is that they are not meeting the needs of their users. The intuitive solution to this problem is to revise the system so it does meet those needs. Unfortunately, this is not as easy as it seems for two reasons. First, once the system is up and running, it is very expensive, in terms of both money and personnel effort, to change a system; second, many DP managers either do not know or refuse to accept the fact that the users are dissatisfied. The best way to deal with the problem is to ensure that it is not permitted to exist in the first place. The logical way to determine what information a manager needs would seem to be to simply ask him. But herein lies the heart of the IRD problem; one cannot always ACCURATELY determine what information a manager needs simply by asking him. This fact is one of the few, if not only, issues upon which everyone in the field of IRA agrees. The question now is, "Why is asking a user what he needs insufficient?"

Davis proposes three basic reasons:

1. the constraints on humans as information processors and problem solvers;

2. the variety and complexity of information requirements; and

3. the complex patterns of interaction among users and analysts in defining requirements. [Ref. 13: p.5]

I shall next explain each of these in reverse order.

A. COMPLEX PATTERNS OF INTERACTION

Specifically, several sub-problems fall into this category. First, it often happens that the user "experts" are (or think they are) too busy to get deeply involved in the MIS development project so they assign less qualified surrogate experts [Ref. 14: p.4], who do not understand the task or its requirements nearly as well as the principal user, to work with the systems analyst. Worse yet, the user may just completely refuse to make more than a token contribution to the effort. This generally prevents him from developing any sort of commitment to the project as well as denying him an understanding of what the computer can do for him.

Second, when the analyst interviews the manager to collect information on the requirements of the system, the manager may feel threatened. Often, managers consider the criteria they use for making decisions to be privileged information or "not for public consumption" and do not want it documented for all to see. This may lead the manager to make omissions or exaggerate, or to provide requirements which are inaccurate, vague, or nonspecific. [Ref. 12] In extreme cases, the user may intentionally give invalid requirements in an effort to sabotage the system. [Ref. 15]

Third, even though he is trying to be honest and helpful, the user may provide the analyst with erroneous information which represents opinion but not fact. Also, he may omit crucial details or very rare but significant exceptions. [Ref. 16: p.15]

Fourth, managers may mistakenly assume that the analyst understands more than he really does about the manager's job. The analyst himself may think he understands the manager's job when, in fact, he does not. [Ref. 16]

Fifth, users generally express their needs in natural language (English) but natural language is not sufficiently precise for stating requirements. [Ref. 14: p.4] This presents another problem when the systems people try to translate those requirements into "computer language." In doing this, they often use their own interpretation of the requirements which is colored by their idea of the solution. [Ref. 17] Further, when they check back with the users to make sure they have obtained the correct requirements, the users do not understand what they are reading, if they bother to read the analysts' documentation at all.

Sixth, the systems "experts" too often get wrapped up in the technical aspects of the MIS, for example, devices, languages, record formats, etc., and lose sight of the overall problem to be solved. [Ref. 14]

Finally, there are too many communication links through which the users' requirements must pass, and at each stop those requirements can be misunderstood, filtered, and passed on incorrectly. To illustrate, the user expresses his needs to the systems analyst who passes them to the systems designer. From there, they continue on to the program designer and finally to the programmers. [Ref. 14]

B. VARIETY AND COMPLEXITY OF INFORMATION REQUIREMENTS

Again, we encounter numerous sub-problems. First, managers are usually experts in performing their jobs but not necessarily in describing them. In the "content-given" world of transaction processing systems and information systems supporting structured decisions, the task to be performed is usually fairly well defined. Procedures, methods, and models already exist; thus the requirements tend to be black-and-white, relatively easily understood, and relatively easily determined. But in the "content-undetermined" world of tactical and strategic MIS which must deal with unstructured decisions, the manager may be incapable of articulating (or even knowing) requirements with the specificity that designers require in the application of traditional design methodologies. [Ref. 18: p.2] More likely, he has only vague, general ideas of what it is he needs. After all, unstructured, high-level decisions are such because there is no model or set process for making the decision. So almost by definition the manager will have difficulty explaining exactly what he needs and, indeed, he may not even know what he needs. In fact, in extreme cases, the manager may not even know what decisions he should be making or how to make them. [Ref. 19]

Second, managers often make unanticipated, emergent decisions; hence, it is difficult to determine in advance just what information will be needed. [Ref. 20]

Third, the procedures, rules, and regulations of an organization can become internalized by a manager when he has been working there for a sufficiently long period of time. They eventually are thought of almost as "customs" of the organization and are applied, without very much consideration as to their applicability, in all situations. This may be contributing to the problem which the MIS is designed to solve, but when asked what he needs, the manager unconsciously requests an information system which supports those same procedures, rules, and regulations. Without some level of objectivity, the user's analysis of his own problem is likely to be distorted. This difficulty is more often encountered when designing operational level or structured-decision MIS.

Fourth, along somewhat the same vein, the way some structured decisions are <u>supposed</u> to be made and the way they are <u>actually</u> made is often different. When questioned by a systems analyst, the user will generally describe the way decisions are supposed to be made in an effort to disguise the fact that procedures are not being followed. The resulting MIS will then not properly support the user's actual decision process. Further, bottlenecks and distortion in the information flow which may exist in the actual decision process are not identified and corrected. [Ref. 15]

Finally, the systems people may also fail to understand the problem due to its complexity despite an honest attempt by the user to provide a straightforward description. [Ref. 14]

C. CONSTRAINTS ON HUMANS AS INFORMATION PROCESSORS

This is the most "scientific" of the three categories and is supported by a good deal of psychological research conducted over the last twenty to thirty years. Davis [Ref. 13] is one of the few IRA researchers to explore this aspect of the IRD problem so all of the following sub-problems, except the first, reflect his ideas.

The first item, proposed by Lederer, basically states that preconceptions and prejudices on the part of the users affect their ability to accurately describe their needs. They may think the computer can do more than it really can or may be bitter about some bad experience in the past. "Science fiction-like stories cause them to attribute too much intelligence to the computer and to underestimate the importance of their careful explanations." [Ref. 16: p.15]

Davis' first explanation incorporates a theory discussed by Newell and Simon. [Ref. 21] The human information processor uses three memories -- external, long-term, and short-term. External memory is any object or device upon which data is recorded or displayed, such as a piece of paper, a chalkboard, or any sort of visual display device. The human brain has a capability for both long- and short-term memory. Long-term memory has essentially unlimited capacity. It requires only a few hundred milliseconds to read (recall) from it, but the write time (commit to memory) is fairly long. [Ref. 13: p.8] Anyone who has studied for an examination or memorized a poem for a high school English class should be familiar with long-term memory. Short-term memory, on the other hand, is human processor memory. It is very fast, but small in capacity. Its limitations may affect human ability to define requirements. [Ref. 13: p.8] To use a computer analogy, short-term memory has been compared to a register or cache memory.

Psychological researchers believe that the capacity of short-term memory is "seven plus or minus two." [Ref.22] In other words, the brain can store from five to nine individual characters, page numbers, words, or even visual images. For example, a telephone number may completely fill short-term memory while dialing. This affects the determination of information requirements in the following way:

> The limits of short-term memory affect the information requirements obtained whenever the process being used to elicit requirements uses only short-term memory (such as an interview unaided by external storage). The user being interviewed cannot hold a large number of items in short-term memory for discussion or analysis purposes and is therefore limited in processing responses. The short-term memory limitation may also affect the number of requirements that users define as important. In various processing activities using short-term memory, the user may have selectively emphasized a few items of information and recorded these in long-term memory as being the most important. These few may be the only ones recalled when a question is asked. [Ref. 13: p.9]

There are two ways to offset these limitations. First, the user can utilize external memory to document his needs as he thinks of them <u>before</u> the interview and, second, by using iterative IRD techniques that elicit small amounts of data at a time and immediately record them.

Third, humans are biased in their selection and use of data. There are four types of bias that affect the information requirements determination process:

(1) Anchoring and adjustment--judgments and decisions are often made by applying adjustments to an anchor point; in other words, the decision-maker will start from a known value, which is the information he is currently receiving, and make modifications from that base to arrive at a new set of requirements. This prevents new requirements, which are unrelated to anything currently being received, from being revealed.

(2) Concreteness--humans tend not to search for information beyond that which they already have. They tend to use what they have in the form it is presented rather than transforming or manipulating it. Consequently, the requirements defined tend to be based on information the users already have about their requirements. They are hesitant to delve deeper into examining what they need beyond what they already know they need.

(3) Recency--humans are more influenced by events which occurred recently than by those which occurred in the past. Therefore, needs discovered through a past event will tend to be overshadowed by needs discovered recently.

(4) Intuitive Statistical Analysis--I shall refer to Davis' explanation:

> Humans are not good as intuitive statisticians. For example, humans do not intuitively understand the effect of sample size on variance and therefore draw unwarranted conclusions from small samples or a small number of occurrences. This is an important limitation because many organizational phenomena occur at a fairly low rate. Also, there is a tendency to identify causality with joint occurrence and assign cause where none exists. These limits of humans in processing low-occurrence data and in identifying causality may result in misjudging the need for information. [Ref. 13: p.10]

To sum up the effect of human bias on the IRD process, we can say that the user is likely to provide inaccurate requirements which are based on "current procedures, currently available information, recent events, and inferences from small samples of events." [Ref. 13: p.9]

Finally, the IRD process is complicated by human problem solving behavior. Two useful concepts here are "task environment" and "problem space." The task environment represents the actual problem to be solved and includes the interrelationships of all variables which influence the environment. The problem space represents the problem as viewed by the problem-solver for purposes of working on a solution. The problem space is thus of a more limited scope than the task environment. In the IRD process, the task environment is the IRD problem itself. The problem space is how a particular analyst or user represents this problem for purposes of determining the requirements for an MIS. Training, prejudice, custom, attitude, and bounded rationality are what create the limitations on the problem space. Of these, only bounded rationality requires an explanation.

Problems are often too complex to be dealt with directly, so the problem-solver must create a model or a simplification of the problem. Rationality is thus bounded by this model which may or may not correspond to the actual problem. The accuracy of the solution, then, depends on how well the model represents the actual problem. A poor model or an oversimplification can lead to a problem space which is so limited that the solution is invalid. This is what often happens in IRD and it is an affliction that can affect analysts and users alike.

In summary, all of these individual difficulties with getting accurate and complete information requirements can be grouped under the three main categories mentioned at the beginning of the chapter (listed here in the order in which they were discussed):

1. The complex patterns of interactions among users and analysts in defining requirements.
2. The variety and complexity of information requirements.
3. The constraints on humans as information processors and problem solvers.

All three of these must be overcome to permit the definition of truly reliable information requirements. The next seven chapters discuss some techniques or methodologies for determining user needs which attempt to overcome some or all of these limitations.

III. THE BASICS

There are certain activities in the IRD process which may be considered as "ground level" or basic; they are the "primitives" of the IRD process. In other words, they cannot be decomposed into sub-activities. These basic activities may be used by themselves but more often are a part of a larger, more comprehensive requirements definition methodology. Interviewing, use of questionnaires, direct observation, document examination, and measurement, all perhaps more accurately described as data gathering techniques, are the subject of this chapter. Additionally, two approaches to applying these techniques, top-down and bottom-up, are covered.

A. INTERVIEWING

This is the most common method for gathering data relating to information needs. It is very effective for obtaining opinions and insights concerning the effects certain policies, programs, and systems have on other acitvities, as well as obtaining evaluations of the performance of existing information systems. Interviewing is also useful in collecting data that cannot otherwise be observed, for example the operation of the "informal organization," and it will sometimes aid in the discovery of sources of resistance to the proposed system. That same interview can then be used to dispel misconceptions and apprehension, thereby dissolving that resistance. Perhaps the main appeal of this data collection tool is that it is one of the simplest and quickest ways to accomplish these tasks. Even in cases where the weaknesses of interviewing hamper its success, it reamins a good a starting point for the systems analyst.

The effectiveness of an interview is hindered for several reasons, many of which are reflected in the IRD problem discussed in Chapter 2. But there are others. First, success of any interview is heavily dependent on the skill of the interviewer; i.e., how he handles himself, to what extent he dominates the conversation, the types of questions he asks, etc. "The interviewer's risk of being 'sandbagged' with erroneous and incomplete information is in direct proportion to his dominance of the interview situation." [Ref. 15: p.27] The environment must be carefully planned beforehand to ensure it is conducive to good dialog. One example of an ill-planned interview is one which is conducted just prior to lunch or quitting time. The interviewee is anxious to get out of the office and a poor dialog is virtually assured.

Second, the interviewee's reponses will be heavily biased by his goals, attitudes, beliefs, and motives. The interviewer must understand these factors so that he can view the responses in the proper perspective.

Third, the interviewee may provide information which is not totally accurate or complete. This may be due to his inability to understand the process he is describing or the question that was asked or to say what he really means; that is, to articulate his needs in a way the analyst will understand. If the respondent has some objection to the proposed system, he may intentionally inject invalid information in an attempt to have the project scrapped before implementation or fail afterwards. Such "counter-implementation" techniques can be very sophisticated and very difficult for the analyst or project team to overcome. [Ref. 23]

Fourth are the ever present communication problems between two humans attempting to pass information. Misunderstanding, misinterpretation, filtration, and related difficulties take their toll. This and similar problems were introduced in the previous chapter as "complex patterns of interaction." Finally, interviewing is simply not practical in situations where there is a great number of individuals to be interviewed or where these people are geographically dispersed.

B. QUESTIONNAIRES

The latter two situations mentioned as weaknesses of interviewing are the forte of questionnaires. That is, they are most useful for collecting data from a large number of individuals or those who are geographically dispersed. A key point in evaluating the utility of a questionnaire is "How committed is the user to solving the problem we are attempting to solve with the use of this questionnaire?" If the users, who are assumed to be the respondents, have a stake in the succussful completion of the project they are more likely to provide more and better information via the questionnaire. Even with user commitment, however, it is difficult to collect small details and to get the respondent to elaborate on certain items he has mentioned. Also, without the personal contact to stimulate the user it is likely that less well thought out answers will be received. In the absence of user commitment, it is wise to contact the proposed respondent ahead of time and attempt to obtain from him some sort of consent to complete and return the questionnaire. This places him under a pseudo-obligation, in a sense, to cooperate. Even so, people generally object to long questionnaires or those that require long or involved answers; multiple choice or yes/no type questions are the most successful. In any event, the project team can expect long response times. Finally, it is often a good idea to distribute a sample questionnaire to a relatively small group and then analyze the results. This enables an evaluation of its effectiveness in eliciting the desired responses. The questionnaire may then be modified, or cancelled, before the actual study begins.

There is significant disenchantment with questionnaires for determining information requirements for many of the reasons cited, but their successful use has been reported in the literature. [Refs. 24,25]

C. DIRECT OBSERVATION

As the name implies, direct observation involves observing the process that the MIS is designed to support. The analyst can see how documents are handled and how policies and procedures are followed under different conditions. Further, this allows him to discover information gaps in the system as well as bottlenecks and other information flow problems. It is the most effective way to learn about the existing system, how successful it is, and how it is affected by external activities or events.

There are two approaches that can be taken. First, the analyst can be an <u>outside observer</u>. That is, he keeps his distance from the activity and just watches. The strong point about this technique is its objectivity. Second, the analyst may choose to be a <u>participant observer</u> in which case he will actually perform the activity which he is observing.

This reveals to the analyst any subtle rivalries, attitudes, or political impacts which may not be apparent to an outsider.

There are three main disadvantages of direct observation. First of all, the results may be biased by the Hawthorne Effect which basically says that people will behave differently than normal when they know they are being observed. The second problem is that the process of observing and drawing the correct conclusions is very difficult. James A. Senn points out:

> The ability to view a series of activities and continually focus on the proper aspects of them without distortion or distraction is a special skill. It is not something that can be easily learned. [Ref. 26: p.476]

Third, the technique works better for clerical tasks and operational level structured decisions than for higher level unstructured decisions. In the latter case, the cognitive process of the decision maker is extremely difficult, if not impossible to observe.

A technique based on direct observation is "task analysis" [Ref. 16: p.17], also called "functional analysis," which is described more fully in the next chapter.

D. DOCUMENT EXAMINATION

The best way to obtain an overall picture of the organization or functional area, according to Guerrieri [Ref. 15: p.27], is through document review, or document examination. Using this method, the analyst looks at reports, memoranda, letters, policy statements, standard operating procedure manuals, and reports of previous investigations and analyses. The object is to see what information is currently being transmitted to whom or requested by whom, how the organization operates, what types of tasks are being done and how, etc. Additionally, computer listings, ledgers, catalogs, and records used in a process should be reviewed to see what information is currently available and in what form.

The problem with this method is that an analyst cannot always trust the documentation because organizations do not always operate the way they say they do. Also, changes in policies and procedures may not be reflected in the organization's documentation for several months or even years. Last, but most important, is that the information reflected in the documents may be extraneous and not even used in reality, while some very important information may travel via informal routes, such as notes or verbal exchanges between managers.

Document examination can still be a valuable tool; the point is that it must be used in conjunction with one or more other techniques. These other techniques can be used to validate the requirements generated from the document review or vice versa.

E. MEASUREMENT

The value of this technique is more or less limited to operational level MIS. Also called <u>sampling</u>, it is used to approximate, within reasonable and workable limits, the frequency with which certain events occur in normal work activities. [Ref. 26: p.477] The data thus gathered can be used to identify and classify exceptions for use in exception reporting, for example. Also, information on certain activites may be needed only if those activities take place with significant frequency.

F. TOP-DOWN VS. BOTTOM-UP APPROACH TO REQUIREMENTS ANALYSIS

Two of the most commonly heard buzzwords in systems development are "top-down" and "bottom-up". These terms relate to the progression through the managerial levels of an organization followed by analysts in determining information requirements.

Using the top-down approach, the higher levels of management are consulted first, followed by progressively lower-level managers until the entire targeted user community has defined their needs. In contrast, the bottom-up approach involves obtaining the needs of the lowest level managers first then progressing up to top management.

The theory behind the top-down approach is that the top level managers have the "big picture"; they define their needs in terms of the overall corporate strategy and objectives. The requirements of the lower level managers should, ideally, all fall into place within the top manager's framework, each forming a piece of the total "MIS mosaic." [Ref. 27: p.78] These lower level managerial needs are a translation of top management's strategy and policies into action-oriented terms. There are three significant advantages with this approach.

(1) Top management is more keenly aware of what is and what is not really important to the organization and can pass this along to the analyst, enabling him to focus on the really relevant information.

(2) This approach avoids the patchwork effect of lower level requirements which may be unrelated to the overall goals of the organization and which subsequently fail to support progress toward achieving those goals.

(3) Often, if lower level management's efforts are moving in a direction away from top management's objectives or are failing to support them, the top-down approach will detect this, enabling the situation to be investigated and corrected before going any further. If the situation went undetected, any MIS implemented in an organization with such a problem will almost surely fail.

Proponents of the bottom-up approach point out that using their method enables the analyst to already understand the operations and needs of lower level managers before entering into discussions with top management. The benefits of this are twofold. First, it provides an opportunity to "sell" top management on the need for an MIS, and second, it serves to bring top management up-to-date on current business problems. However, Krauss points out that a key weakness of starting out at the "ground level" is that

> ...the fragments of information gathered may not fit into a mosaic of any kind and as a result may produce misunderstandings and confusion. The absence of a central or unifying theme, with at least some guideposts along the way, may well get a negative reaction from top managers when MIS discussion finally gets to them. [Ref. 27: p.79]

The implication is that top-down is the preferred approach although bottom-up may apply in certain situations.

G. SUMMARY

Sometimes the tools discussed in this chapter form an IRD technique of their own, but more often they form the foundation for the techniques presented in subsequent chapters. Individual tools, or combinations of them, are components of the more sophisticated methodologies. In addition to being used to complement each other, one may be used to validate requirements determined primarily by another. The top-down and bottom-up concepts are relevant because many of the techniques presented later may be applied in a top-down or bottom-up fashion.

The remainder of Part I discusses some of the IRD techniques that have appeared in the literature from 1963 to the present and organizes them into the following groups:

Early Techniques (Chapter 4)

Information Analysis (Chapter 5)

Group Techniques (Chapter 6)

Other Approaches (Chapter 7)

Iterative Design Techniques (Chapter 8)

Selection Methods (Chapter 9)

User self-determination of needs (Chapter 10)

There is a considerable gray area and overlap between many of these groups so the classification of individual techniques was made subjectively; however, such a framework is still believed helpful in organizing and understanding the concepts presented.

IV. EARLY TECHNIQUES

Two techniques fall into this category: Ask and Analyze (my own terminology), and Functional Analysis (sometimes called Task Analysis). These are not described as "early" because they were used only in the early days of MIS, but because they were the first techniques to be used; they are still in use today. In fact, as we shall see in Part II, they are still the most commonly used techniques.

A. ASK AND ANALYZE

Due to the poor results historically obtained from "asking" users what their needs are, there is little, if any, research currently being conducted in this area and there is very little mention of it at all in the recent literature. I include it in this survey of techniques, however, because it is still so widely used.

Heany [Ref. 28] writes of a requirements determination process which primarily emphasizes the Ask and Analyze technique. The "asking" phase consists of the operating manager meeting with the system designer to explain what is needed. [Ref. 28: p.50] The designer must then analyze the requirements which resulted from the meeting to determine what the true requirements are, since the needs as described by the user cannot be accepted at face value.

More specifically, he must analyze the wording of the specification, its source, and the business situation out of which it arose. [Ref. 28: p.51] The goal is to discover and weed out contradictions and nuances. Cliches, slang, and shoptalk are notoriously poor ways of conveying information and should also be removed along with generalizations and overly technical language. Basically, the requirements should be couched in very precise terms understood by all involved.

The requirements thus generated must then be validated. The method proposed to accomplish this is for the designer to discuss the requirements with users in various levels of the organization and elicit their comments. Since perspectives change with the organizational level, an agreement on the requirements thus obtained should assure their validity.

This is a relatively quick and simple way of determining information needs but suffers from most of the problems discussed in Chapter Two. The validation process may help somewhat but not enough to produce a truly accurate set of requirements.

B. FUNCTIONAL ANALYSIS

This is an extremely popular technique whereby the requirements for information are seen to be related to the functions or the objectives of the organization and, hence, are derived from them. Most implementations of this technique call for decomposing the functions into individual tasks or component activities which must be performed in order to accomplish the function or meet the organizational objectives. These tasks are then analyzed in terms of what they entail, when and how often they are performed, and any additional considerations believed relevant. Information requirements are then derived from each of these activities.

An application of this technique was described by Sisson during the development of an MIS for U.S. Naval shipyards. [Ref. 29] In this case, the functions of the shipyards were identified and decomposed into second and then third level functional units. The third level functional units were examined to determine the management processes to be supported and criteria necessary for the execution of those processes, and then the information needed from a management information system to execute those management processes was identified. [Ref. 29: p.237]

Miller [Ref. 4] provides a more detailed description of the technique. The procedure is composed of five steps:

1. identify key operations,
2. list input/output and suboperations,
3. identify managerial actions,
4. list effects of managerial actions, and
5. derive information requirements.

The analysis is conducted as follows. First, the analyst, perhaps in conjunction with the manager, must identify the key operations that the enterprise must accomplish in order to continue to function. [Ref. 4: p.611] Second, these operations should be further classified by listing the input and output for each, as well as a description of the "suboperations" that compose the major operation.

Upon completion of these two steps, the analyst will have assembled a conceptual model of what the firm, as a whole, does. [Ref. 4: p.613] The next step involves building a similar model to describe the functions of management. Miller describes this step:

> We have an adequate statement of what the company does, but we must now decide how management manipulates the things that the company does in order to make it successful or unsuccessful. The basic question can be simply stated as 'How are the operations managed?' [Ref. 4: p.613]

The implementation of this step, then, involves identifying the significant managerial actions taken to influence each operation.

To round out the managerial conceptual model, the effects of each of these managerial actions are listed and associated with the action that caused their occurrence. This relationship is called the "action/result model," and a close examination of it reveals that specific information requirements may be derived from each element.

By way of example, let us assume the firm is a wholesaling business. One of the operations that can be identified in step 1 is "Get Orders." One of the many managerial actions which may be taken to influence this operation is "Adjust frequency of salesmen's order solicitation." Miller explains:

> This automatically suggests the question, 'How often do salesmen solicit orders?' The simplest answer to that question is the total number of calls made by all salesmen in a month. Of course, we might want a finer breakdown--number of calls made by each salesman, and number of calls made on each customer....we might want to turn it around and look at it from the customers' point of view. How many solicitations does the average customer receive in a month? [Ref. 4: p.618]

We might then want to measure the result of this action. Looking at the action/result model shows us that one of the expected results is a change in "salesmen's travel time," so that would define a requirement to track and report the time each salesman, or an average salesman, spends traveling per unit of time.

The strength of this technique is that it provides a structured method of determining which information is most likely to have a bearing on the operations of the organization. It seems, however, that it would generate an overwhelming number of requirements. Also, neither of the two cases presented emphasized much user involvement or user-analyst interaction, which raises the possibility that many of the requirements identified will be considered irrelevant by the user or, just as bad, some requirements which are relevant and necessary will fail to be uncovered. Even with user participation, the systematic and logical progression of the process may lull him into accepting the outputs without taking the time to analyze them and decide if they meet his needs or not.

In an attempt to avoid many of the irrelevant requirements generated using functional analysis, Chapman et al. [Ref. 30], proposed a variant. Their method involves first identifying the <u>demands</u> placed on the system by both external entities (e.g., government or a parent organization) and internal entities (e.g., top management). Users are then interviewed to obtain an initial set of requirements which are "bounced" against the demands on the system and the organization's objectives. Any requirement which does not support the satisfaction of a demand or an objective is discarded, even though an individual manager may sincerely wish to receive that element of information. Chapman and collegues report that:

> ...no requirement can be retained for any reason except that it is necessary to meet the valid demands made on the system.
>
> The analyst must probe and question until he knows why the information flowing from a given requirement is needed, how it is used and where, whether used elsewhere or filed, and if so, why, until he knows every use and disposition of the information being generated by each requirement. In order to do this he must learn the content of specific jobs in depth and the purpose each is intended to serve. [Ref. 30: p.38]

They further state, "Requirements are determined on the basis of actual need rather than on desire without any demonstrable reason." [Ref. 30: p.38] Despite its good intentions, I have serious doubts about the effectiveness of such a technique. Their intent is laudable, but this puts the analyst in a position of making the decision as to which requirements justifications are satisfactory and which are not, a decision which he is doubtfully qualified to make. Further, management is likely to resist making such extensive justifications to the analyst. The system thus runs the risk of being unresponsive to management's needs.

For the interested reader, Langefors [Ref. 31], Krauss [Ref. 27], Hartman, Matthes, and Proeme [Ref. 32], and Levinton and Dunning [Ref. 33] also discuss techniques and concepts which could be classified as Functional Analysis.

V. INFORMATION ANALYSIS

The term "information analysis" generally refers to the techniques of "data analysis" and "decision analysis" but I shall stretch it here to also include "protocol analysis," and "information environmental analysis."

A. DATA ANALYSIS

This method is sometimes called the "traditional" or "bottom-up" approach to determining information requirements (not to be confused with "bottom-up" as used in Chapter 3). [Ref. 34] Working together closely, the analyst and manager identify all sources of information currently being received by the manager and drawn upon for decision making. This is more than a simple document examinatiion; all sources of information whether formal (e.g., reports) or informal (e.g., personal notes or discussions between managers) are analyzed. With the manager's assistance, the analyst attempts to determine how the information is used and to establish its relevancy, resulting in the elimination of unnecessary information. Next, the analyst and manager discuss needs which are currently unsatisfied in an attempt to identify what new information is required.

The advantage of this method is that it starts from a concrete, known base--currently received information. This accomodates the "anchor and adjust" tendency of users (as mentioned in Chapter 2) when defining their information requirements. But herein lies its weakness. This "anchor and adjust" tendency inhibits the discovery of the true information requirements. The data analysis technique also fails to link those requirements to the decision process actually used by the manager. Even so, this approach is believed to work reasonably well with structured decisions. [Ref. 35]

B. DECISION ANALYSIS

Sometimes refered to as the "top-down" approach (again, not to be confused with the usage of that same term in Chapter 3), decision analysis requires that the analyst and the manager identify all the decisions which the latter makes, or should make. Then each decision is analyzed in an attempt to build a model of the process used to reach that decision. The information used at each step along the way is examined as is information which should or could be used if it were available. The result of this activity is the set of information required to make each decision. This would then be implemented in the information system.

The strength of this approach is that it forces the manager to think about how he makes his decisions and what information he really uses. This in turn increases the likelihood that the needs which are defined will be accurate and complete.

Opponents of the method claim that, for unstructured decisions at least, the manager is unable to identify the process he follows. Proponents respond that, while this is true in many cases, the act of forcing the manager to analyze what he does may serve to clarify some previously poorly understood processes.

C. DATA VS. DECISION ANALYSIS

Munro and Davis conducted an experiment designed to compare the two methods. [Ref. 34] Expectations prior to research were that (1) both decision analysis and data analysis would perform better on structured decisions than on unstructured decisions; (2) both methods would perform equally well on highly structured decisions; (3) neither approach would provide accurate results when used with highly unstructured decisions; and (4) with relatively unstructured decisions, decision analysis would perform better than data analysis.

The findings of the experiment were somewhat surprising. Greatly simplified and summarized, the results indicated that: (1) the overall performance of the two methods was not significantly different, (2) both performed better on unstructured decisions than on structured decisions; (3) data analysis performed poorly on structured decisions relative to either decision analysis or data analysis on unstructured decisions; and (4) the effectiveness of decision analysis on unstructured decisions was only slightly greater than that on structured decisions. The indication, then, is that (1) decision analysis should be used on structured decisions and that (2) either method could be used on unstructured decisions with approximately equal results.

Finally, the most interesting finding was that there was little difference in practice between the two methods. Munro and Davis explain:

> The researchers observed, for the entire set of decisions, that use of the two techniques seemed to result in similar interviews. In fact, it often seemed impossible to discuss information needs (data analysis) without discussing decision procedures (decision analysis) and vice-versa. It became evident that many of the steps in the decision procedure were actually the acquisition and analysis of particular items of information. The only manner in which the techniques seemed to differ was in the analytical stage following the interview. While data analysis involved an analysis of the data, decision analysis involved the modeling (flowcharting) of the decision procedure....[Ref. 34: p.64]

Unfortunately, in the six years that have passed since these findings were reported, no evidence of further research on this topic has been found.

Other authors discussing data and/or decision analysis include Zani [Ref. 36] (decision analysis), Penney [Ref. 37] (decision analysis), King and Cleland [Ref. 38] (decision analysis), Courtney [Ref. 25] (data and decision analysis), and Ein-Dor and Segev [Ref. 19] (data and decision analysis).

D. PROTOCOL ANALYSIS

Little has been written on the use of protocol analysis in MIS development, but it is nevertheless an interesting technique. An analyst using protocol analysis will ask the user to "think aloud" as he performs an actual or simulated task. The analyst records what the user says and from this, information requirements are derived. Alternatively, the user may simply jot down his thoughts as he performs a task without requiring the analyst to be present. As the reader has undoubtedly noticed, this technique is quite similar to decision analysis, and it shares many of the latter's advantages. However, much of the benefit of the analysis of the decision process by the user (in decision analysis) is lost because the user-analyst interaction is omitted. A disadvantage which it shares with decision analysis is that it causes the analyst to focus on the usual; unfortunately, unusual circumstances and exceptions result in substantial problems for information systems analysts. [Ref. 16: p.17]

E. INFORMATION ENVIRONMENTAL ANALYSIS

A very interesting variant of the decision/data analysis techniques is one used by Willoughby and Gardner [Ref. 39] during the design of an energy related information retrieval system. Referred to as "information environmental analysis," the method principally revolved around the concept of an "information tour." The analysts believed that any analysis of the type of information used, how it was used, how often it was used and how it was stored and retrieved was best conducted in the users' normal work environment. The authors explain:

> Factors such as the content and organization of file cabinets and bookcases may seem incidental but were in fact, important indicators of user perceived relationships among information types and users' priorities for accessing information. For these reasons, the participants were asked to provide an 'information tour' of their workspace and to describe day-to-day activities which utilized and generated information. [Ref. 39: p.516]

Four advantages of this process were identified. First, it would provide more information than the users were likely to think of in a straight interview process; second, it aided in revealing the type of information that users would find useful in the performance of their jobs; third, it gave the analyst an idea of what the users were looking for in a computerized system to support the task under study; and fourth, it drew the users into the systems design process.

VI. GROUP TECHNIQUES

This category includes all the methods which involve some sort of group interaction as the primary focus of the technique. Discussed in this chapter are Brainstorming, the Nominal Group Technique, and the Delphi Method.

All of these methods share the common advantages and disadvantages of group processes. The advantages include:

1. The knowledge and information of the total group is greater than that of any one individual in the group.

2. The misinformation of one member may be cancelled by the true information of another.

3. The number of factors that can be considered by a group is much greater than that of any one member.

4. Groups are generally more willing to take risks.

Some of the disadvantages are:

1. If related but incorrect information is held by two or more members, each one's misinformation may tend to support and amplify that of the others.

2. There may be strong social pressure on a member who disagrees, perhaps correctly, with the group opinion to "fall in line." If that individual concedes, the group has lost the benefit of his accurate information.

3. Individuals with dominant personalities and loud voices tend to improperly influence the group's behavior.

4. If the group members are improperly selected and are not representative of the total target population, they may share a common bias. This increases the chance of occurrence of item 1 in this list.

5. The group's goal may become distorted during the process of discussion to the point that members may seek to reach agreement rather than the best solution.

6. The group discussion, if not properly moderated, may drift onto irrelevant issues or rehash past issues which were already settled.

A. BRAINSTORMING

In its raw form, brainstorming involves a group of people who meet to solve a problem. They contribute any idea for a possible solution that comes to mind. Initially, no criticism is permitted. The principle involved is that the more people participating, the more likely they are to provide a wide range of possible solutions. The criticism is prohibited to avoid inhibiting the members' creativity. It is hoped that, as more ideas are generated, the number of good ideas will increase. A synergistic effect is also assumed. Implementations of this technique generally include some sort of discussion or evaluation of the ideas as a second phase. The result of the process is a set of requirements for the system. Although slight variations of this technique are used fairly often in industry, very little has appeared in the literature. [Ref. 16] The reader should bear in mind that, despite the benefits of group interaction, brainstorming (as with all of the group techniques) is essentially just a more involved way of "asking" the manager what he needs, with all its associated pitfalls.

B. DELPHI METHOD

The Delphi method has often been proposed as a method of determining information requirements when the users are geographically dispersed yet group interaction is desired.

The "standard" Delphi technique consists of distributing a questionnaire to "experts" to elicit their views or opinions of solutions to a particular task or problem. The participants have no contact with one another and each may not even know who the other participants in the study are. When the administrator receives the responses, he summarizes and distributes them to the participants along with a follow-up questionnaire. In this way, the respondents can see how their initial response compared with those of others, who remain anonymous throughout the process. This revised questionnaire explains to each participant that several other experts in the field were also surveyed and that the summary reflects their answers. The participants must then reevaluate their initial responses in light of the responses of the other experts in completing the revised questionnaire. Often, respondents are also requested to state their reason for answering the way they did. The process is then repeated. Over the course of several iterations, the responses will tend to converge, the number of iterations performed depending upon the degree of convergence considered satisfactory by the administrator.

The advantages of Delphi are that, since the responses of other "group" members are fed back to each participant, most of the benefits of group interaction are realized while at the same time most of the problems associated with groups are reduced or eliminated. For example, dominance by one individual with a strong personality or loud voice, strong social pressure on dissenters to abandon a contrary view, the protracted discussion of irrelevant and redundant information, and the tendency of groups to work for agreement rather than the best solution to the problem is reduced or eliminated.

Sackman [Ref. 40] was quite vocal about Delphi's weaknesses. Some that he listed include:

> ...considerable evidence that results based on the opinions of laymen and "experts" are indistinguishable in many cases; aggregate raw opinion presented as systematic prediction; technical shortcomings, such as untested and uncontrolled halo effects in the application of Delphi questionnaires; unsystematic and nonreplicable definition and use of "experts"; manipulated group suggestion rather than real consensus; ambiguity in results stemming from vague questions; acceptance of snap judgements on complex issues; and the virtual absence of a vigorous critical

> methodological literature even though hundreds of Delphi studies have been published. [Ref. 40: p.v]

The difficulties experienced in connection with the use of "experts" are often not as crucial when using the technique for IRD purposes because the developers normally know who the users of the proposed system will be, although this is certainly not so in all cases.

Another problem with the technique is that it suffers from many of the same weaknesses that afflict any use of questionnaires (see Chapter 3).

When applied to determining information requirements, the first round usually involves the actual elicitation of needs and the users rank these needs in order of their importance in the subsequent rounds. Jones [Ref. 41] used Delphi to determine the requirements for a computerized military command and control system and used an unstructured questionnaire to initially obtain the requirements. Remus, Sprague, and Gilbert [Ref. 42] established the needs of the managers of the College Administration of the University of Hawaii using a slightly modified Delphi technique. They obtained the initial requirements through a brainstorming session. In the report of their study, Remus et al. hypothesized several benefits to be realized from the use of Delphi in determining information requirements:

1. Because they are exposed to the responses of other users who may be in different positions throughout the

organization, the participants are influenced to take a more organizational view of the information needs.

2. The process results in a prioritized list of needs, which provides guidance to the system developer in deciding which needs to include when constrained by resource limitations.

3. Involvement of the information users is enhanced by each participant's awareness of the needs of others.

4. The convergence of opinion obtained by Delphi serves to enhance agreement on critical information systems needs and identify those expressed needs which are significantly non-standard. [Ref. 42: p.543]

Though not addressed in any of the reports, intuition hints that Delphi would not really solve many of the difficulties involved with IRD. For instance, a simple ranking of proposed requirements, themselves obtained using a rather shaky technique, and then not validated, would probably not reveal missing, inaccurate, vague, incomplete, or exaggerated needs. There are at least two reasons for this:

1. The initial statement of requirements was not formulated using any particularly analytical or thought-provoking technique.

2. There is no opportunity for discussion and evaluation of each requirement; one invalid requirement may merely

be ranked relative to another invalid requirement which produces, not surprisingly, an invalid result.

Lederer agrees that the method is suited primarily to higher level rather than detailed tasks. [Ref. 16: p.19] A more useful application of Delphi in the systems development process is illustrated by Willoughby and Gardner [Ref. 39] who relied on a Delphi survey to determine who would be the "outside users" of an energy related information system.

C. NOMINAL GROUP TECHNIQUE

Although not effective for determining minute details, the Nominal Group Technique may be useful in uncovering more general information requirements. [Ref. 16] This method was developed by Delbecq, Van de Ven, and Gustafson [Ref. 43] and is performed in two phases.

In Phase I, the participants are given a problem or task to solve. Each individual writes down as many solutions as he/she can think of within a given time limit. It is important that no group interaction be allowed at this point so that members have a chance to respond before being influenced by the group. In Henderson and West's implementation of the technique, some of the problems used were "list those decisions you make in order to fulfill your responsibilities" and "list those things you need to know (information) in order to support this set of critical decisions." [Ref. 44: p.47] Next, the group coordinator polls each participant in round robin fashion and has them provide one item from their list. This polling continues until all the participants have exhausted their list. No criticism of solutions is allowed at this point. There are three benefits gained by using a round robin procedure: (1) it eliminates dominance of the group by any one person, (2) no individual can "hide" behind the group and avoid participation, and (3) one member's idea may stimulate other members to produce related ideas, a process called "hitchhiking" by Delbecq et al.

Once all the ideas have been recorded and displayed by the group coordinator, a period of clarification begins. It is important that no evaluations or criticisms be allowed during this step--only clarifications to ensure that all participants understand the meaning of each solution. In the course of clarification, some items may be combined, deleted or restored.

In Phase II, the group votes on the solutions, thus validating the results and ranking them by priority. The results of the vote are fed back to the group where they are discussed, sometimes ending in another vote. Hopefully, at this point a group consensus will have been reached.

Henderson and West used a slightly modified version of the Nominal Group Technique in obtaining information requirements for a medium size manufacturing firm and reported favorable results. [Ref. 44]

In a sense, this approach is a combination of the brainstorming and Delphi methods. From brainstorming it borrows the face-to-face discussion which allows evaluation of the validity of proposed solutions or information needs, and from Delphi the repetitive voting or ranking process which has been shown to be so effective in producing group consensus.

VII. OTHER APPROACHES

This grouping has been termed "other" because it is composed of several techniques which are unique and do not really align themselves well with any of the other categories. This chapter will discuss the Critical Success Factors approach, Simulation, DEFINEPAC, the Critical Incident Technique, and the Data Base approach.

A. CRITICAL SUCCESS FACTORS APPROACH

This method was developed by John F. Rockart of MIT in an attempt to eliminate the overload of irrelevant information with which managers have had to suffer since the advent of MIS and as a means of focusing the content of the information system on the really important aspects of the organization. [Ref. 45]

Rockart describes "critical success factors" (CSFs) as "the limited number of areas in which results, if they are satisfactory, will ensure successful competitive performance for the organization." [Ref. 45: p.85] In other words, satisfactory performance in the CSF areas is a prerequisite to overall success of the organization and achievement of the organization's goals. Failure to achieve adequate results in these areas almost certainly leads to a disappointing level of performance for the organization.

The first step in analyzing a manager's information needs using the CSF approach is for the manager to define his goals. Next, with the aid of the analyst, he determines the critical success factors that influence attainment of those goals. Then, ways of measuring performance in the CSF areas are discussed, and the reports and data needed to monitor this performance is defined. Some of this information may already be available; that which is not is a candidate for inclusion in a new information system. Once developed, this system is modified as necessary to reflect changes in CSFs (the changing business environment will cause a manager's view of which factors are critical to change) and even changes in the management personnel the system was designed to serve.

The major appeal of this method is that it supplies only the information that the manager feels is truly essential to the continuing success of his organization and eliminates the rest. Mintzberg points out that brevity, fragmentation, and verbal communication characterize a manager's work [Ref. 46], implying that managers simply do not have the time to dig through voluminous reports to find the few really important elements of information. Therefore, it is important to cut down the amount of information supplied by an information system, otherwise the system will not be used.

Other advantages of the CSF approach include:

1. It provides better control by enabling the manager to concentrate his attention on the areas most important to him.

2. The process of analyzing and defining CSFs and the measures for monitoring performance in these areas is helpful to the manager in that it guides him in determining the level of effort to invest in the different aspects of the organization.

3. The information system is designed to be flexible, i.e., changes in CSFs and changes in managers are considered when the system is developed and, hence, may be incorporated relatively easily.

The primary weakness of the method is that it entails interviewing the manager and "asking" him what his CSFs are. Davis commented that, "The possibilites of failure with the method center on the ability of executives to respond with critical success factors that are correct, complete, and sufficient." [Ref. 47: p.57] The same difficulties discussed in Chapter 2 are applicable here, most notably the limited capacity of humans for information storage, bias in selection and use of data, and bounded rationality.

Despite these disadvantages, Rockart reported substantial success with his method in experimental usage. Munro and Wheeler applied the technique in a study involving the information requirements of management in a large natural resources company [Ref. 48] and also reported success. They attempted to overcome the weaknesses of the method by using the corporate planning process to aid in the identification of CSFs. Their study emphasized that by examining the corporate plan, or strategy statement, the objectives of managers within the corporation could be more accurately determined, since the two are (or should be) related. Once objectives are identified, the manager and analyst determine the critical success factors that influence the successful achievement of the objectives. From here, specific performance measures and standards are identified, followed by data required to measure performance, and finally decisions and information required to implement the plan.

The strength of Munro and Wheeler's approach is that the CSF interviews are structured by the presence of goals and objectives and this, they claim, helps nullify the effect of human information processing constraints.

B. SIMULATION

In determining information requirements, simulation comes in three flavors: Paper Simulation, Human Simulation, and System Simulation (an original term of this author).

1. Paper Simulation

This entails, in its simplest form, drawing a sample output report on paper and presenting it to the user for comment/modification. Sometimes a CRT screen is used in place of paper. This is an extremely popular and inexpensive technique for verifying the format of a report when developing an MIS [Ref. 16]. More elaborate paper simulation schemes may be used, especially when the system being developed is an interactive one. [Ref. 49]

2. <u>Human Simulation</u>

A more complex form of simulation is Human Simulation. Van Cott and Kinkade [Ref. 50] studied the feasibility of this technique for determining the information needs of biologists. In their study, the researchers established an information clearinghouse that the biologists could call anytime they needed information. This clearinghouse was staffed by a Request Receiver who took the initial request from the biologists; a Request Processor who listened to the tape recorded request, interpreted and summarized it; a Search Strategist who decided which sources would be used to fill the request; an Information Searcher who obtained the requested information; and a Messenger who delivered the information to the requesting scientist. Response time ranged from one to thirty-eight days, with the average being seventeen days in the first study and seven days in each of the two follow-on studies. Over a six-week period, requests made of the clearinghouse were studied in an attempt to learn what information the scientists were demanding, what type of interaction existed between the requestor and receiver, and the requesting behavior of the scientists. Two follow-on studies were done of six weeks and five weeks duration, respectively, varying the number of scientists participating and some of the characteristics of the clearinghouse. The result of the studies was that use of such a technique in the situation studied was found to be practical.

The advantage of this technique is that the requirements determination method itself does not intrude on the behavior of the scientist, causing it to change, or confuse what a scientist says he needs with what he actually uses. [Ref. 50: p.211]

Unfortunately, this approach is very expensive in both time and personnel required and would seem to have limited applicability in the business world due to the immediacy required of the responses.

3. System Simulation

One of the weaknesses of the decision analysis approach was described earlier as the inablility of the user to articulate his decision process because he did not understand it himself. System Simulation (a term I have coined myself to describe a method studied by Werner, Greenburg, and Goldberg [Ref. 51] for determining the information needs of an outpatient clinic) tries to make up for this difficulty. Rather than attempting to analyze the user's decision process, it is much easier and more effective to design an environment in which behavior can be observed to determine what information is being used and how it is being used. Werner et al. point out that, "The behavior of the physician does not necessarily reveal his information processing model, but it does reveal the information he uses." [Ref. 51: p.43]

The method requires the creation of a large data base with the capability of returning any single item of information. This data base would need to contain all the information likely to be needed by the user. A software monitor is also necessary to record the items requested, the information extracted, and the order of extraction. This monitor is transparent to the user, so he has no idea that his behavior is being observed. An analysis of the data collected by the monitor should provide the analyst with a list of all the information important to the user.

The advantages of this method are that it should produce an accurate and complete list of user needs; since there is no communication between user and analyst, there is little possibility of ambiguity, misinterpretation, exaggeration, etc. Also, as in human simulation, the user's behavior is not changed by the intrusion of the IRD method itself.

Unfortunately, this approach requires the use of a fairly large amount of computer resources, and for this reason may be impractical. Also, should some information needed by the user be inadvertently omitted from the data base, the results will be invalid. Lastly, exceptional or unusual cases which fail to occur during the period of observation will cause important but infrequently used information to be excluded from the information system.

C. DEFINEPAC

We have already discussed the fact that simply "asking" a manager what he needs is not likely to produce an accurate and complete set of requirements. Yet, few systems analysts have the expertise to "tell" the manager what he needs. Kennedy and Mahapatra [Ref. 52] surveyed the literature base of existing techniques but found none they considered adequate to do the job when dealing with unstructured decisions. They concluded that the method most likely to succeed in determining information requirements would be one which provided some sort of structure to a normally unstructured problem. They explained:

> ...it is assumed here that effective inquiry requires a structured set of cues to trigger memory and to focus managerial attention. Unstructured inquiry may elicit good suggestions, but these will be so far from exhaustive that the resulting MIS will be of marginal value. The dilemma to be resolved, then, is to model the "unmodelable." [Ref. 52: p.74]

The model they have derived is called DEFINEPAC and is illustrated in Figure 7-1. The heart of this model is the activities and resources for which a manager is responsible.

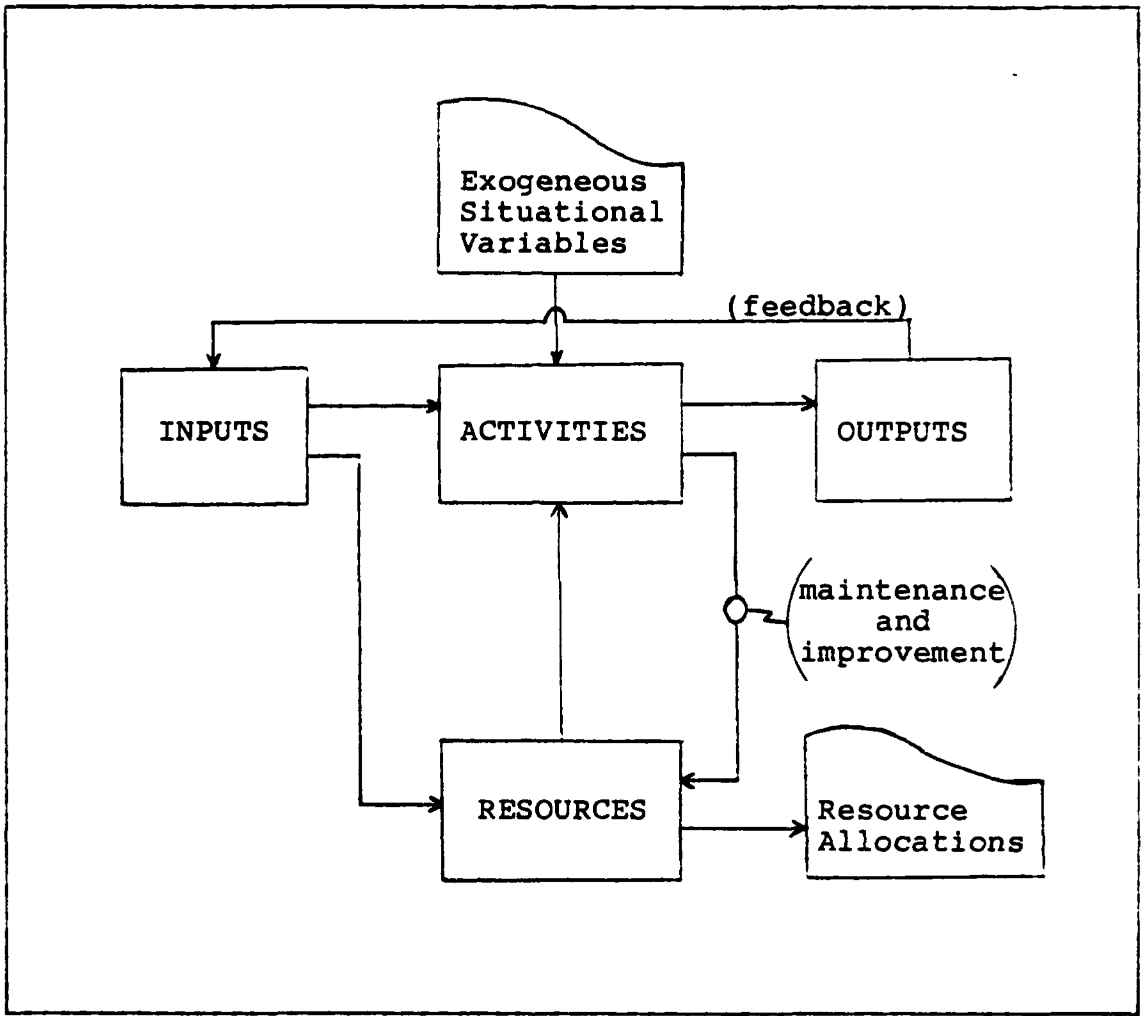


Figure 7-1: DEFINEPAC Framework for Decision Modeling
Source: [Ref. 52: p.75]

Decisions, then, should be concerned only with the flows indicated in Figure 7-1. The object is not to define the precise interrelationships between each of the variables of the model--such a task would be far too complex and would render the models useless--but, rather, to simply identify which input variables are (somehow) relevant to decisions about output variables. [Ref. 52: p.74]

After analyzing the decision process using this framework, the analyst will have a list of information elements, but with no clue as to how they fit into the system or how important they are. The DEFINEPAC process proposes that the analyst need not worry about how they fit (interrelate), only about how important they are so that the choice of which information elements to include in the system can be made in light of existing information system constraints (e.g., cost, computer resource limitations, etc.). What is needed, then, is a relative ranking of the information elements. Kennedy and Mahapatra believe that the best person to perform this ranking is the manager himself. The results of this task will not suffer from the same problems afflicting the process of "asking" a manager what his needs are because:

> ...wherever it is appropriate to entrust decision responsibility for ill-structured problems to the intuitive skill of managers, we believe it is *a fortiori* appropriate to trust the judgement of these same individuals in evaluating their actual and potential supplies of information.[Ref. 52: p.76]

The two researchers go on to describe a mathematical model for accomplishing the ranking which considers the importance of the information element to the decision being considered, the importance of the decision to the department or organizational subunit and the importance of the department to the overall system.

So, without needing to understand how information elements are used (which is the most difficult phase of analyzing an unstructured decision situation), the analyst has determined which information to include in the MIS. Granted, without bothering to study the interrelationships of the information elements a lot of irrelevant information will be identified for inclusion in the system, but if that information is truly irrelevant, it will fall out that way in the ranking and will end up at the bottom of the list.

The weakness of this method is that it is not a simple task to quantify the importance of certain elements to a larger system as the authors would have us believe. Also, it is difficult to factor in unquantifiable considerations or to indicate conditional importance (e.g., element X is important to decision Y only if condition Z exists).

Despite the early success with this approach claimed by its developers, no further references to it have been found in the eight years since the cited article was published.

D. CRITICAL INCIDENT TECHNIQUE

This is a good technique for using in conjunction with other IRD methods, but is insufficient to stand alone. It basically entails soliciting from the user events that occurred which had extremely favorable and extremely unfavorable outcomes. It then identifies the user activities which contributed to these incidents. Analysis of the activities should reveal very important information which should be included in the system and information whose absence could have undesirable affects.

Although this method has been suggested for use in determining information needs, Lederer [Ref. 16] comments that there is no known documentation of the application of the technique to automated information systems.

E. DATA BASE APPROACH

This is actually a "non-technique" for determining management's information needs; no requirements analysis is done. Instead, every piece of data being collected anywhere in the organization is thrown into the MIS data base. The manager can then use whatever he wants and the Information Services department is always prepared for any situation that might arise. Head refers to this as the "Kitchen Sink" approach [Ref. 53: p.51]. Krauss points out that:

> Much of the data-base approach is justified on the grounds that being prepared for nearly any situation has benefits that exceed the overhead or waste inherent in the excessive storage and other handling it necessitates. [Ref. 27: p.75]

Nevertheless, this is an inefficient way of providing information to management, except possibly in the case of an interactive Decision Support System. Even there, however, the cost may be prohibitive, despite the fact that Data Base Management Systems and sophisticated query languages such as

FOCUS and RAMIS II have made this approach technically feasible.

VIII. ITERATIVE DESIGN TECHNIQUES

All of the techniques discussed so far have weaknesses; none of them are perfect. The unfruitful search for the "ideal" IRD method has led many IRA researchers to conclude that there may be no such thing. Parker observed in 1970 that:

> It is not possible by questionnaire and interview techniques to determine how users will, in fact, react to a system they have not seen or experienced at the time the questions are being asked. [Ref. 24: p.283]

As has been previously discussed, managers find it extremely difficult to articulate, or even know, what information they need to do their jobs. This is complicated by the fact that they often do not understand the capabilities and limitations of the information system available to them so they do not even know what scale to use in defining what they need. Users must first have a foundation, a reference point, around which to assemble their information needs. McKeever and Kruse have pointed out that managers tend to be better at reacting than inventing. [Ref. 54: p.19]

Similarly, McCracken has suggested that "the plaintive cry of the user is 'I can't tell you what I want, but I'd recognize it if you showed me one!' " [Ref. 55: p.447] Another reason that traditional methods of requirements determination have been perceived as unsuccessful is that they do not allow for changes in the users' requirements during the course of the system development. But such change is inevitable. McCracken and Jackson draw an analogy with the "Heisenberg Uncertainty Principle," namely:

> ...any system development activity inevitably changes the environment out of which the need for the system arouse. System development methodology must take into account that the user, and his or her needs and environment, change during the process. [Ref. 56: p.31]

For these reasons, the concept of iterative design was developed. It ~~ative design involves developing a "rough" system for us evaluate, and then modifying that system in accordance with the users' wishes. This "evaluate and modify" process is iterated until the system satisfies the users. This system provides the users with a reference point from which they can move toward the appropriate solution. Recalling Davis' explanation of "anchoring and adjustment" in Chapter 2, the iterative design process is consistent with human nature.

There are essentially two approaches to iterative design: Prototyping and Heuristic Development. Each of these will be briefly described.

A. PROTOTYPING

There are four steps involved in prototyping. First, the user's basic information requirements must be identified. It is important to understand that the analyst is concerned only with the essential features of the user's information requirements, as opposed to a highly detailed analysis of specific needs. The requirements definition need not be complete, and should not involve much time or expense. Second, using these preliminary requirements, a system (called a "prototype" [Ref. 57] or a "breadboard" [Ref. 58]) is quickly developed, with an emphasis on changeability, and provided to the user. Definitions of "quickly" have ranged from "overnight" [Ref. 59] to "weeks" [Ref. 60]. Almost no consideration is given to processing efficiency of this system; in fact, it need not even be programmed in the language in which it will ultimately run. The goal in this step is not to produce a perfect system, but just to produce a system, period. In the third step, this prototype is given to the manager for him to use and evaluate. Finally, the system developer, using the manager's comments, revises and enhances the prototype, correcting undesirable or missing features identified by the user. It is important, again, that this modification be made quickly and the prototype returned to the user for another iteration of the process.

B. HEURISTIC DEVELOPMENT

Very similar to prototyping, Heuristic Development involves using an iterative design technique for building only the output system of the MIS. Wetherbe describes the process. [Ref. 61] Data currently being used to support management is collected and loaded into a data base. Next, screen formats and reports are developed to provide the information required by the users. This "output system" is given to the users for them to operate and evaluate. Just as in prototyping, the output system is refined until it meets the user's needs. At this point, a system to do the input and processing is developed using a traditional structured design approach.

C. EVALUATION OF ITERATIVE DESIGN

Iterative design has great promise for several reasons. First, it gets a working system into the user's hands much faster than traditional techniques. This is important in keeping the user happy and keeping him interested. Second, and somewhat related, is that this initial system, when used by the manager, stimulates further identification of requirements. Wetherbe explains:

> The experience gained by the user interaction with the system's technologies and capabilities functions as a catalyst that allows users to more fully envision and articulate their information requirements. [Ref. 61: p.SR/14]

Third, a user evaluation of the system will take place regardless of the development approach used. With any system, users will identify features that they need added, deleted, or changed. Iterative design approaches exploit this tendency by integrating such evaluation and subsequent modification into the technique. This way, the revisions can be made earlier in the development process and hence, more cheaply. Also, changes can be made much more easily and cheaply to the prototype than to a fully developed and implemented system because the prototype is designed from the beginning to be changed. Fourth, overall lifecycle cost of the system will probably be lower due to a significant reduction in maintenance costs, which are a major expense in conventionally designed systems. Such reduced costs are possible because most of the maintenance takes place at a higher level (i.e., in the prototype) and because once the production system is implemented, there should be less maintenance required.

A fifth advantage is that the inevitable changing requirements of the user can be accomodated more quickly and cheaply. The reader is no doubt familiar with horror stories of changing requirements causing systems development time to double, and cost to triple. Sixth, overall development time may be less, although this point is often debated. This is because not all prototypes are "throw-aways." That is, in some cases the prototype system evolves until it meets the user's needs and at that time it simply becomes the production system. Similarly, traditionally developed systems go through a "use and modify" cycle as well; hence, it becomes difficult to precisely define when either of these systems are "complete" since modifications may be made periodically throughout the lifecycle. Finally, iterative design methodologies force the users to become actively involved in the project, which is a prerequisite for success. A large percentage of the successful meeting of needs is the responsibility of the users, and they play a significant role in setting the pace of the development effort.

Arguments against iterative design techniques have centered around the fact that the development cost is greater. In the short run, this is true. Expensive computer resources are consumed in running and modifying the system. Since most prototype systems were not written for efficient processing, perhaps more resources than would otherwise be necessary are utilized. The strength of iterative methodologies lies in their long run lifecycle savings. Unfortunately, many managers are forced by the organizational environment to focus their energies on short-term efficiencies; hence, iterative design is rejected. Moreover, in systems where the task to be supported is well-defined and structured and user requirements are well understood, iterative design may, in fact, be more expensive even over the lifecycle.

In summary, iterative design, and especially prototyping, is the wave of the future. It is perhaps the most widely published IRD technique ever. As will be seen in Part II, it has been relatively slow to catch on, however, primarily because it is such a revolutionary approach.

IX. SELECTION METHODS

Which IRD technique is best? As with any management-related topic, the answer is, "It depends." Just as we have Situational and Contingency theories of management, we have Situational and Contingency Theories of Information Requirements Analysis. These theories basically hold that the best IRD method to be used in any particular case varies depending on the circumstances.

A. SITUATIONAL PERSPECTIVE

Taggart and Tharp have developed what they refer to as the "Situational Perspective on Information Requirements Analysis" (SPIRA). [Ref. 2] The authors first identified ten "aspects" of IRA techniques. See Appendix A for a brief description of each. They then reviewed much of the IRA (or, as they call it, "MIRA", Management Information Requirements Analysis) literature and rated each technique on the basis of how thoroughly the ten aspects were treated; a grade of 1 indicated that the technique gave no consideration to that aspect; 2, recognition of the aspect; and 3, significant treatment of concepts covered in the aspect. A sample of seven techniques rated by Taggart and Tharp and the grades assigned for each aspect is presented in Figure 9-1.

When a new system is being developed, the three phases of SPIRA are implemented. The first phase is Profile Development. The analyst completes an "analyst profile" questionnaire which contains one question or statement concerning the analyst's personal awareness and skills relating to each of the ten aspects. Each question has three possible responses, corresponding to a grade of 1,2,

	ASPECTS from Appendix A									
TECHNIQUE	1	2	3	4	5	6	7	8	9	10
Chapman, et al. [Ref. 31]	2	1	2	2	1	1	1	1	3	1
Hartman, et al. [Ref. 33]	3	1	2	3	2	3	1	3	3	2
Heany [Ref. 29]	2	1	1	1	1	1	1	3	2	2
Langefors [Ref. 32]	2	2	3	2	1	1	1	2	2	2
McKeever, et al. [Ref. 55]	2	1	3	3	3	1	3	1	2	2
Miller [Ref. 4]	1	1	1	2	2	2	1	1	3	2
Norton [Ref. 3]	2	1	2	1	1	1	1	2	1	3
ANALYST PROFILE	3	3	1	2	2	3	3	1	1	2
SITUATION PROFILE	3	1	2	2	3	2	3	1	2	1

Figure 9-1: MIRA Technique Ratings, including sample Profile ratings.
Source: Adapted from [Ref. 2]

or 3 respectively, similar to the MIRA technique grades described earlier. A second questionnaire, the "situation profile," is similar to the analyst profile except that the questions or statements relate to the analysis situation rather than to the analyst. The situation profile is completed by the analyst <u>after</u> throughly discussing each item with the users. Sample results of an analyst and situation profile are shown at the bottom of Figure 9-1.

The second phase, Composite Evaluation, attempts to match technique capabilites to the conditions of the situation and the skills of the analyst. The reader may follow this process graphically as it is explained by referring to Figure 9-2.

	ASPECTS from Appendix A										
	4	3		7		5	6	2		1	
<u>TECHNIQUE</u>	1	2	3	4	5	6	7	8	9	10	
~~Chapman, et al. [Ref. 31]~~	2	1	2	2	1	1	1	1	3	1	12
Hartman, et al. [Ref. 33]	3	1	2	3	2	3	1	3	3	2	
~~Heany [Ref. 29]~~	2	1	1	1	1	1	1	3	2	2	8
~~Langefors [Ref. 32]~~	2	2	3	2	1	1	1	2	2	2	11
McKeever, et al. [Ref. 55]	2	1	3	3	3	1	3	1	2	2	
~~Miller [Ref. 4]~~	1	1	1	2	2	2	1	1	3	2	9
~~Norton [Ref. 3]~~	2	1	2	1	1	1	1	2	1	3	10
ANALYST PROFILE	3	3	1	2	2	3	3	1	1	2	
SITUATION PROFILE	3	1	2	2	3	2	3	1	2	1	

Figure 9-2: Steps in the selection of a MIRA technique.

1. Examining the results of the situation profile, any aspect having a grade of 1 is unimportant to the analysis situation, so corresponding aspect columns may be eliminated (lines 1,2, and 3 in Figure 9-2).

2. Examining the results of the analyst profile, any aspect graded as 3 is no longer of concern because the analyst is well skilled in these areas and need not rely on the MIRA technique for support in those aspects. Hence, we may eliminate the corresponding aspect columns (lines 4,5, and 6).

3. Any aspect graded as 2 in <u>both</u> the analyst and situation profile is not critical because the analyst is presumed to have enough skill to cover the moderate requirement of the situation in that aspect, so the corresponding aspect column is eliminated (line 7).

4. Now examine the technique ratings. Any technique graded as 1 in any of the remaining aspects provides inadequate support to the analyst and, hence, may be eliminated (lines 8, 9, 10, 11, and 12).

5. Looking at the techniques still under consideration, eliminate any having a grade of 2 in aspects where the analyst grade is 1 and the situation grade is 3 (none in this case).

6. Of the remaining techniques, all aspects still under consideration should have a grade of 2 or 3. If not, repeat steps 1-5.

The final phase is Technique Selection. Since all of the remaining techniques are presumed to adequately support the analyst in the critical areas in which he is weak, the final selection should be made based on analyst preference, analyst and technique style compatability, cost of acquiring new technology, etc. Naturally, the analyst must have a reference describing each of the techniques; the authors have provided just such a document. [Ref. 62]

The advantage of this method is discussed by its developers:

> Through the use of SPIRA, the analyst can combine personal skills with a MIRA technique to achieve an integrated set of conceptual skills closely matching the organizational situation. SPIRA attempts to complement existing skills and knowledge and to compensate for those which are missing. [Ref. 2: p.235]

There are three problems with the method, however. First, the base of rated MIRA techniques must be continually updated as new techniques are introduced. Second, the analyst must be familiar with or be prepared to learn new techniques with each systems analysis effort. This leaves him with little opportunity to develop expertise in any one of them if the situations vary too much. Third, all ratings (technique, analyst profile, and situation profile) are subjective and, hence, subject to error or misjudgement.

While a promising method overall, there has been a lack of any further significant discussion of it in the literature.

B. CONTINGENCY APPROACH

Another selection method which takes into account the varying conditions existing in each systems development effort is the Contingency Approach, developed initially in 1978 by Naumann and Davis [Ref. 63] (see also [Ref. 18] and [Ref. 8]) and refined a couple of times since by Davis. [Refs. 6,13] Its most recent recent form will be discussed here.

The basis for this approach is that the presence of certain situational factors (contingencies) introduces uncertainty into the information analysis process [Ref. 8: p.274], and the level of this uncertainty can be determined from an analysis of the situational factors; the IRD technique which best deals with the given level of uncertainty may then be selected. In this method the term "uncertainty" refers to the state of knowledge of the "real" information needs. [Ref. 18: p.5]

Let us first examine the "situational factors" identified by Davis:

1. Characteristics of the utilizing system (i.e., the task)--a stable, well-defined, and well-understood system or one with structured activities and decisions will produce less uncertainty than an unstable and poorly understood system or one with highly unstructured activities and decisions.

2. Characteristics of the proposed or existing information or application system which supports the task--a system with simple requirements or one designed for clerical support will produce less uncertainty than a system with complex or unusual requirements or one aimed at managerial decision-making.

3. Characteristics of the users--systems serving only one or a few users or those whose users understand the task to be performed and are sophisticated with respect to information systems development and usage will produce less uncertainty than those of opposite characteristics.

4. Characteristics of the analysts--a highly trained and experienced analyst who is familiar with information systems similar to the one proposed produces less uncertainty than an analyst with little prior training or experience.

The IRD strategy chosen should be one of the following:

1. Asking--despite the plethora of problems associated with this strategy, it may be effective in cases where the users know exactly what they want; for example, Davis mentions simple reports and listings, revisions of existing reports, simple transaction documents such as acknowledgements or requests for data, an ad hoc report for a well-defined purpose [Ref. 6: p.49], or a system designed to meet very precise external requirements such as those emanating from law, regulations, or higher management.

Potential methods within this strategy include closed questions (e.g., multiple choice), open questions (user responds freely), brainstorming, and group consensus (e.g., Delphi or Nominal Group Technique).

2. Deriving from an existing information system--the "existing system" need not be the one to be replaced; it may also be a similar system in another organization, a proprietary system or package, or a system described in a published work. Data analysis, described in Chapter 5, also comes under this category.

3. Synthesis from characteristics of the utilizing system--this involves examining the tasks or activities to be supported by the information system and, from that, deriving the information requirements. Items to be analyzed could include objectives and processes (e.g., functional analysis, Chapter 4), decisions (e.g., decision analysis, Chapter 5), and critical factors (e.g., CSF Approach, Chapter 7).

4. Discovering from experimentation with an evolving information system--for example, iterative design techniques (prototyping or heuristic development, Chapter 8).

In selecting the appropriate strategy, the analyst should first examine the characteristics of the four situational factors as they apply to the systems development effort and determine how they affect (i.e., add or reduce uncertainty) the three "process uncertainties." These three process uncertainties are uncertainty with respect to "existence and stability of a usable set of requirements ... users' ability to specify requirements ... [and] ability of analysts to elicit requirements and evaluate their correctness and completeness." [Ref. 13: p.22]

Next, the analyst should evaluate the combined effects of the process uncertainties on the overall requirements uncertainty, arriving at an "estimated overall level of requirements process uncertainty."

Finally, this estimate should be used to select a strategy. See Figure 9-3 for the recommended strategies to be used with different uncertainty levels. A primary and secondary strategy may be selected. Within each one, an associated method should be selected, with supplemental methods chosen as desired. In other words, the analyst need not restrict himself to <u>one</u> strategy/method but may use several in conjunction with one other, the object being to select as a secondary strategy/method one which is strong in the areas in which the primary is weak.

The Contingency Approach is intuitively appealing despite the fact that it, like the Situational Perspective, is based almost totally on subjective appraisals which may be inaccurate, and is perhaps more practical to implement than the Situational Perspective.

OVERALL UNCERTAINTY LEVEL	CORRESPONDING STRATEGY
Low	Asking
↑	Deriving from existing system
↓	Synthesis from utilizing system
High	Discovering from experimentation

Figure 9-3: Selection of an IRD Strategy
Source: Adapted from [Ref. 13]

Other, more theoretical discussions of selection methods were published by Bariff [Ref. 64] and Dhar and Davis. [Ref. 5]

In summary, it should be apparent that <u>no</u> IRD technique is the "one best way" of determining information requirements and that there must be a framework for evaluating the available methods and choosing the best for the specific situation. This chapter contains two possible frameworks, and no IRA effort should be undertaken without reference to one of them, or at least something similar.

X. USER SELF-DETERMINATION OF NEEDS

If it is so difficult for MIS designers to determine the information needs of the users, why not let the users do it themselves? This chapter shall offer three possible solutions to the IRD problem involving user self-determination of needs. The first is a popular method, currently implemented in numerous organizations; the second is a method proposed in the literature, the extent of its implementation is unknown; and the third is an original proposal of this author.

A. USER PROJECT TEAMS

This methodology involves the use of an MIS project team composed almost exclusively of users. The key position of Project Manager, especially, is filled by someone from user management. DP personnel are assigned to do the technical portions (program design and coding) and there is usually one analyst to act as an advisor during the requirements analysis phase but the rest of the team is made up of users. In this way, not only are the users totally involved, but they are directly responsible for the success or failure of the system. Ideally, users will be assigned full-time to the project team (usually on a rotational basis). It is absolutely essential that such an endeavor have the total support of top management.

The difficulty with this technique is the disruption it causes to the users' normal jobs. If assignments are full-time, some assurance must be provided to the individuals concerned that their career progression will not be hampered by such an assignment. If users work on the project part-time, the conflict with other duties may cause the project team members to be somewhat ineffective as their efforts are diluted.

Given the proper organizational climate, this method is one of the best available for successful development of relatively large management information systems.

B. "TROJAN HORSE" STRATEGY

Proposed by Synnott and Gruber [Ref. 65], this strategy involves providing "gifts" in the form of systems professionals to user divisions. Synnott and Gruber explain, "Trojan horses quickly learn the business and promote systems solutions to business problems." [Ref. 65: p.80] While originally designed as an information technology penetration strategy, its use can be applied to IRA as well, though in a slightly abridged form.

In this strategy, the Information Systems Division transfers a systems professional into the user department requiring the system. He then becomes a user himself. Over time, as he learns the business, the analyst-turned-user gains an awareness of information systems needs. He should then be able to specify those needs without being susceptible to the usual problems associated with "asking" a user about his needs (because of his systems background).

As with user project teams, top management support is essential. The main problem with the technique is that the individual transferred is likely to be concerned about his career progression. Hence, satisfactory arrangements must be agreed upon by all concerned before such a transfer takes place.

C. INFORMATION CENTER

The Information Center concept was developed by IBM around mid-1980 and has since caught on with tremendous success. It was developed in response to the growing backlog of application development requests from which most, if not all, Information Systems departments are suffering. The idea is that if the users can do some of the minor work by themselves, without having to wait two years or more for the IS department to get around to it, they can benefit from the productivity increase provided by the minor application much sooner. This translates to overall improved user productivity.

The I/C provides the user with a terminal, a consultant for training and assistance, and software packages for solving his problem, such as a data manipulation package, report generation package, query package, etc. L. W. Hammond explains:

> The objective of an I/C is to provide users access to data on their own terms so that they can solve their own business problems. [Ref. 66: p.133]

He goes on to emphasize that:

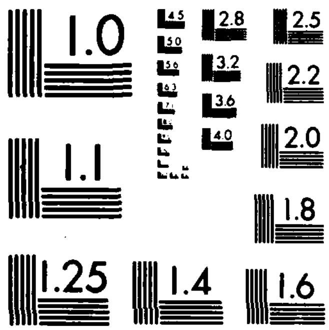
> The type of work the I/C is intended to support is the short job, the one-time query, the simple report, the minor change, etc., and not the work that requires the discipline of formal project development procedures. It is not a replacement for a way around the longer schedules usually required to develop a system. [Ref. 66: p.134]

While this is valid in regard to the original I/C concept, it seems that many management-oriented information systems and decision support systems could be more easily and cheaply implemented by the user himself using the I/C than by the traditional systems development approach. What this user-developed system would cost in processing inefficiency would probably be much less than what a full-fledged development effort would cost, even for a small system. The author believes that the I/C concept should and will move in this direction in the future. Mollen and Bakshi, from IBM, report results supporting this contention obtained from certain organizations that have implemented Information Centers [Ref 67: p.7]:

1. IBM Canada, Ltd. reported that about 50% of the project requests are being implemented by end user computing.

AD-A132 998 THE DETERMINATION OF USER INFORMATION REQUIREMENTS DURING THE DEVELOPMENT OF MANAGEMENT INFORMATION SYSTEMS(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA P R GARDELLA JUN 83 — 2/2

UNCLASSIFIED — F/G 5/2 — NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

2. The American Automobile Association of Michigan claims that, "'Soon, our professional programmers will be doing only the difficult jobs, the big online programs, and everything else will be done by the users themselves.'"

Part II will report on a survey conducted of organizations having an Information Center to further determine if industrial I/C usage supports the belief mentioned above. The results did not indicate unanimous support, but did indicate a sufficient amount to establish that the <u>potential</u> for such an evolution exists.

There are problems with user-developed systems, to be sure, not the least of which is that they tend to be individual user-specific with limited inter-departmental applicability. This may be a just tough enough problem that user-developed systems will never be in the majority, but they can nonetheless play a significant role in meeting user needs.

PART II

IRD TECHNIQUE SURVEY AND CONCLUSIONS

The second part of this thesis deals with the current application of the IRD techniques discussed in Part I to actual systems development. Chapter 11 reports on the results of an industry survey taken to determine which techniques are being used in practice, Chapter 12 comments on the success of the study, and Chapter 13 attempts to draw some conclusions.

XI. PRACTICAL APPLICATIONS

A series of generally unstructured interviews was conducted with officials from several organizations during the months of March and April, 1983. These interviews, some done in person and some done by phone, involved individuals of varying positions in fifteen organizations of different types. Appendix B lists the positions of the individuals interviewed, the type of organization, and the size of their Data Processing effort.

Although the results of this survey are considered too unreliable for any sort of rigorous statistical analysis, some useful information may still be derived from the data collected. The remainder of this chapter attempts to do just that; Chapter 12 will discuss deficiencies of the survey and recommendations for improvement.

In each of the unstructured interviews, a series of questions were asked with the interviewee free to expound as much as he wished on each. Sometimes clarifying questions were asked to sharpen understanding of certain points raised by the individual, but generally he was free to address each issue as he wished. Appendix C contains a list of the questions used during the interviews.

A. RESULTS OF THE INTERVIEWS

A "formal" IRD method is one in which the steps or activities involved are specified in advance and intentionally followed by a systems analyst. An "informal" method is one in which there are no clear and concrete steps to be followed. There is no conscious effort to use any certain technique; rather the individual analyst proceeds in a "seat of the pants" fashion, based on experience or intuition as to how the needs analysis should be conducted. All of the "Basics" in Chapter 3 may be components of informal techniques. "Ask and Analyze" and "Functional Analysis" in Chapter 4 are informal (Miller's and Sisson's methods are formal versions of the otherwise informal technique of Functional Analysis). Paper Simulation and the Data Base Approach are also informal. All remaining approaches discussed are considered formal.

Based on this distinction, ten of the fifteen organizations studied used informal IRD methods. Of the five using formal procedures, three involved the use of some form of user project teams who held overall responsibility for the success of the project. One of these three also used some prototyping. Of the ten organizations using informal procedures, it appeared that only one occasionally, or had in the past, used a formal approach; namely, prototyping. Appendix D presents a summary of approaches used and type (formal or informal). The classification of approaches as to type was a subjective one, based on the author's understanding of the method described by the interviewee. Interestingly, in twelve of the fifteen cases, the individuals claimed their techniques to be successful in accurately determining users' information requirements. Of the three remaining, two were using methods never before tried in their organization and the projects were not yet complete so no evaluation of success could be made, and the third realized that asking the users what they needed was not always successful unless the results of such a process were cycled back to the users for modification and approval.

When discussing weaknesses of the methods used, only five of fifteen interviewees acknowledged that their techniques suffered from user-analyst communication problems or the inability of users to articualte their needs accurately. This was unexpectedly low in light of the problems discussed in Chapter 2. There are two possible reasons for such a percentage: (1) the participants interviewed do not realize, or do not accept, the fact that their methods are less than totally reliable, or (2) the situation surrounding the system development effort is such that it produces very low uncertainty (in reference to the Contingency Approach). Of these five, one felt that the solution to the problem was to add new analysts to the project, one thought requirements validation was sufficient to make up for the weakness, two offered no way around the problem (the solution is just to do the best they can), and the fifth seemed to indicate that a poor statement of requirements was the users' problem to solve. This leads one to believe that there is not a wide recognition in industry that the IRD problem is significant, or even exists.

In the area of user resistance to IRD techniques, ten reported meeting some level of resistance in most cases. Four of them said this resistance was mostly centered in the lower level employees. Of the five firms reporting no resistance, one was using the user project team concept and one a "pseudo" user project team concept, both of which hold the <u>user</u> responsible for the success of the project. The other two organizations using this approach did encounter some resistance. In both of these, one of the users' complaints was that they felt uncomfortable in their new role and did not know what to do. In the other cases, the most common cause of resistance identified was that the users felt they were simply too busy to be bothered with determining information requirements.

McKeen, Naumann, and Davis observed that "the method for determining information requirements employed by a practitioner may be used either bacause the analyst has had experience with only one method or because the selected method has worked successfully before for systems of this type.... In short, current practice is based upon experience and intuition--not theory or empirical research." [Ref. 18: p.3] This statement was tested in the survey and it was found that two-thirds of the individuals interviewed knew of no IRD methods other than the one (or ones) they were currently using. Of the five who had tried different techniques, only one had intentionally tested and evaluated multiple techniques. The others merely evolved to a method incorporating greater user involvement, and the fifth one changed to an approach requiring less time due to a constraint in that area. It appears, then, that McKeen, Naumann, and Davis were correct.

King and Cleland have commented that, "Rather than creating an information system to serve an existing organizational system, [the analyst] should attempt to influence the restructuring of the decision-making process so that the MIS may be oriented toward the support of a more nearly 'optimal' process." [Ref. 38: p.292] Believing that the MIS should never attempt to impose a <u>change</u> on the manager's decision process but, rather, should <u>support</u> that which exists, the next question was designed to reveal how widespread was the view that an MIS should attempt to alter the decision process. It turns out to be very widespread. Every individual to whom this question was posed replied that they did, in fact, attempt to change the manager's decision process. The intent was not to force the manager to conform to a systems-oriented approach, but rather to optimize the decision by allowing the manager to include information that was previously unavailable in a useful form.

In Chapter 1, it was discussed that faulty information requirements were one of the major causes of MIS failure. In an attempt to gain evidence confirming this, the next question inquired as to whether any of the interviewees had experienced unsuccessful MIS's, and if so, what were the causes. Three of the survey participants reported they had never had a system failure (a "failure" was defined as a system which was not used after implementation or one with which the users were dissatisfied). Of the twelve who did, only half laid the blame on inaccurate or incomplete user requirements. Other reasons given included the assignment to the job of an untrained analyst, insufficiently motivated users who refused to take the time to learn the system or to update the data in the system, and other similar ones. These responses were surprising in view of the discussion in Chapter 1, but in retrospect, the question posed was a difficult one to answer for two reasons. First, no system ever meets the users' needs the first time but gradually attains that objective only after being modified and refined. Second, over time, the users of the system change, and the situation and environment in which the system operates changes. If the system does not also change, even the best is bound to fall into disuse or will eventually cease to satisfy the needs of the users. Hence, it is difficult to adjudge a system as a "failure" or "unsuccessful," and even more difficult to determine the exact cause of failure.

Prior to conducting this study, it had been the author's belief that the practical application of much of the academic research in IRA was quite limited. The research was designed, therefore, to discover how many IRA practitioners in industry were aware of the different techniques developed over the years through academic research. So, each interviewee was confronted with the techniques listed in Appendix C, question 14 (each of which were discussed in Part I except for the Infological Development and the REP Test which were omitted due to their complexity and relatively light treatment in the literature). The responses are tabulated in Appendix E.

If we consider each cell in the table an "opportunity" for a practitioner to be aware of an IRD technique, then out of 224 opportunities, only 39 instances of awareness were found (17.4%). This seems to reveal a rather large gap between IRA research and practical applications. Ahituv, Munro, and Wand also noticed this problem, identifying a "need to bridge the gap between the abundant conceptual literature on the one hand and practical applications of IA [Information Analysis] activities on the other." [Ref. 20: p.144] This gap exists despite the fact that some of the techniques (most notably decision analysis, data analysis, and paper simulation) are very similar to techniques in use, though informally, in many of the organizations surveyed. The similarity may be due to the fact that these three techniques are a logical outgrowth of the currently accepted undesirability of accepting users' statements of needs at face value. In other words, these techniques were perceived to be useful by analysts apparently using their own intuition, as opposed to analysts who were knowledgeable of IRA research results.

Another observation that may be made is that some form of iterative design technique is being used in many cases, although the formal procedures of neither prototyping nor heuristic development are being followed in most of them. The researcher tends to suspect that many, although certainly not all, of the individuals who listed "prototyping" or "heuristic development" as one of their techniques were attempting to use a more traditional approach but were forced to repeatedly refine their systems upon discovering that those systems did not satisfy the users. There is no hard data to support this suspicion, but an intuitive evaluation of the comments made by many of the interviewees points in this direction.

The final area of the survey to be reviewed deals with the use of Information Cente. . M .ny of the participants reported that their organization had no I/C, so some additional firms, not otherwise a part of the survey, were contacted for information. Of fifteen I/C's contacted, only three (20%) reported any large-scale system development taking place. The rest reported developing mostly one-time, ad hoc reports as well as some continuing small-scale, intra-departmental reporting systems. Further, only four individuals (26.7%) foresee any full-scale development in the future, and one of these believed that only Analysis and Reporting type systems, rather than Transaction Systems, would be built in this manner. Only two of the four felt that the I/C would eventually take over all systems development from the IS Department.

The reasons most often given for retaining full-scale systems development within the IS Department were:

1. Users do not have the expertise to build large-scale transaction systems or reporting systems that cross departmental lines;

2. User-friendly software tools used in the I/C's such as FOCUS, RAMIS II, etc., are too inefficient to be used for large production systems; and

3. Most users simply do not want to get involved with full-scale systems development.

Drawing any conclusions from the above is exceedingly difficult, since the comments made are subjective opinions. Also, the Information Center concept is still only about three years old and, hence, has a lot of growing and evolving yet to do. But the author stands firm in his belief that in the future, more and more management-oriented information systems and decision support systems will be developed by the users themselves using Information Centers, thus eliminating the IRD problem in those cases.

XII. EVALUATION OF THE SURVEY

As was briefly alluded to early in the last chapter, the results of the survey undertaken as part of this project, while perhaps interesting, are of questionable validity. In this chapter, we shall discuss each of the four flaws which became evident in retrospect and will then suggest possible alternate methods for conducting a similar study.

A. COMMUNICATIONS

Communications between interviewer and interviewee were flawed for four basic reasons:

1. The terminology used in the questions revealed itself to be very confusing. The DP world has so many different meanings for the same term and so many different terms for the same concept, that the interviewees often had difficulty understanding what the researcher was asking. For example, many of the participants misunderstood the terms "Management Information System," "Decision Support System," "Management-Oriented Information System," and "Transaction Processing System." Similarly, most of the interviewees misinterpreted the names of many of the published IRD techniques about which they were questioned (see question 14, Appendix B). For instance, many of the

participants are familiar with the word "heuristic," namely, "trial and error", and assumed that "heuristic development" merely referred to a situation where the system was modified if not correct the first time. While this is the basic premise behind the concept of "heuristic development," it is not consistent with the formal description of the method provided by Wetherbe. [Ref. 61] Also, in more than one case, the Nominal Group Technique was mistakenly assumed to be any method which involves a group meeting to discuss requirements, and the Contingency Approach was interpreted as reflecting the organization's plans for dealing with physical disasters involving the computer system.

The result of these misinterpretations was that, often, participants claimed they used a particular method when in fact they did not. Some of these instances were uncovered during the interviews and the issues clarified, but it seems certain that many were not.

2. Many of the responses received are incomplete and oversimplified to the point that important information is missing. This may be due to the fact that, quite often, the individuals interviewed were unsure of the level of detail desired in their answers. Consequently, they summarized their explanations and just presented the salient features of their IRD methods, thus omitting a great deal of valuable information. Partly contributing to this problem was the time limit of the interviews. Though no explicit limit was

set, there is a practical limitation on the amount of time a manager will take away from his work to participate in an interview from which he or she will derive no benefit.

3. Much of the information received during the interviews is tainted by the bias of the managers involved. Recall that the main thrust of this paper is toward management-oriented information systems as opposed to transaction processing systems. In industry, however, the great majority of applications systems currently in place are transaction processing systems. Hence, when the managers interviewed spoke about requirements analysis during systems development, they were really addressing these issues in the context of transaction processing systems rather than management-oriented information systems, despite the fact that the managerial orientation of the survey was explained beforehand. Alloway and Quillard identified this problem in the report of a survey published in 1982. They observed:

> I/S policies and procedures, organizational structure, and expertise in developing applications are dominated by transaction processors. [Ref. 68: p.10]

They further point out:

> In most companies the established standard procedures for needs identification, project prioritization, and project selection are the result of institutionalized transaction processing experience. [Ref. 68: p.20]

4. Having never participated in a systems development effort and having never experienced first-hand the IRD

problem, a total grasp of the issues involved was missing. A secondary objective of this study was that it would be an educational tool to provide the researcher with an understanding of this apparently problematic area. Therefore, when confronted with DP professionals who also seemed to not understand the problem and who requested a clearer explanation of what was wanted, a clarification was not always satisfactorily provided.

B. PARTICIPANTS

When planning this survey, it was assumed that the appropriate person to speak with would be an organization's lead, or senior, systems analyst. This seemed to be the best place to find an individual who had the "big picture" while at the same time was not so far removed from the "action" that he or she would be unfamiliar with the IRD techniques used. Much to the researcher's surprise, few people understood what was meant by the terms "lead systems analyst" or "senior systems analyst." It was therefore decided to move up the organizational ladder and look for the systems development manager or someone of a similar title. As shown in Appendix B, the participants often ended up being inappropriate for the survey. Most participants appear to have been too far removed from the actual information requirements determination activity, despite the fact that, when setting up the interview, assurances were received that he or she was the proper person to interview.

C. SYSTEM TYPES

Once again, recall that this study intended to focus on management-oriented information systems. When arranging interviews, interest was expressed in those types of systems as opposed to transaction processing systems. In many of the cases, however, it became evident at some point during the interview that the organization, or manager, involved did not deal to an adequate extent with management-oriented systems. Hence, much of the data collected is inapplicable to the type of systems being studied and therefore is invalid.

D. RESEARCH PRACTICES

Due to a lack of training and experience in conducting studies such as this, the approach to the problem was inappropriate. Because of the way the interviews were structured, the types of questions that were asked, and an inability to clarify what was being sought, the resulting responses are difficult to compare since they are based on different levels of understanding and interpretation on the part of the interviewee and different probing on the part of the interviewer. Additionally, in each of the cases, different I/S situations and conditions existed. Further, the same type of information was not collected at each interview. For example, due to the relatively unstructured nature of the interview, the participants were free to expound on each question as they wished, with very little prompting from the interviewer. The result of this is that just because one manager addressed a certain point and another did not does not mean that that point applies only in the first situation. It merely means that the point did not come up in conversation in the second instance--it may, however, apply equally in both cases. This makes the drawing of any firm conclusions extremely hazardous.

E. ALTERNATE METHODS

Rather than restricting data collection efforts to interviews, a technique of direct observation augmented by interviews would have been much more effective. "Sitting in" on the requirements analysis phase of a system development process and observing first-hand which techniques were used would have solved the problems with communications, participants, and system types. This latter problem could also be lessened by better screening of a potential participating organization's systems development projects before commencing the observation phase. Of course, an interview with the cognizant manager beforehand to gain his approval and support would be essential.

To eliminate the problems caused by the research practices used, the following methodology is proposed.

Practicing systems analysts, actively involved in the requirements analysis phase of a system development project should be observed and interviewed to determine what techniques they are actually using. Whether or not they have "heard" of one of the published techniques is not important, because the analysts may have received informal training on one of these techniques without being aware of the name assigned to it by academic researchers. Therefore, the study should involve determining the techniques used through observation and interview, followed by an attempt to categorize the observed techniques into one or more of the published IRD techniques, if possible. If any of the techniques fall into the informal, or more primitive (e.g., "Ask and Analyze," Chapter 4), technique categories and/or the analyst is apparently unaware of the more formal and higher level techniques, then an effort should be made to find out why. For example, is his lack of sophistication based on inadequate education, training, or experience? Or does he use the observed techniques based on an informed and deliberate choice, made after considering all the relevant factors (Chapter 9)?

A more theoretical question may be: does the problem lie with the IRA researchers and institutions (such as MIT's Center for Information Systems Research [CISR] or the University of Minnesota's Management Information Systems Research Center [MISRC]) for developing IRD methods inappropriate for practical application? Has there been any effort to establish a mechanism for transferring the knowledge of IRD techniques to industry?

These procedures would have to be applied to at least 25-30 organizations so that the study would be statistically significant.

Admittedly, this proposed methodology would be very expensive in both time and money, and for this reason it would not have been possible within the constraints existing in the environment in which this study was performed. However, this is what is necessary to produce valid and significant results.

F. SUMMARY

The value of the project just completed is as a pilot study for the type of research just proposed. It has established the impracticality of using a pure interview approach and has helped clarify and solidify the areas of importance and specific objectives of such a project. Therefore, a follow-on research project is necessary to determine if, in fact, systems analysts are using the IRD techniques found in the literature and if not, why not. Such research is necessary because the results will no doubt prove useful in the future reduction or elimination of MIS's which fail to meet the needs of the users.

XIII. CONCLUSIONS

Despite the problems associated with this survey, it is still useful in that it provides us with a general, though not entirely accurate, indication of the current state of Information Requirements Analysis as practiced in industry. This indication is that there is a large gap between the IRD techniques discussed in the MIS literature and the IRD techniques applied in industry.

Why does this gap exist? Ahituv et al. lay the blame on the same problem identified by Alloway and Quillard mentioned in the previous chapter. They explain:

> ... most systems analysts have been involved in developing information systems for the operational levels of the organization. These applications...tend to be structured so that most of the information requirements are obvious. As a result, systems analysts do not always perceive the importance of the IA [Information Analysis] phase when faced with less-structured situations. [Ref. 20: p.144]

Another reason for this gap is the lack of education of practicing analysts in the field of IRA. The only survey participant who was familiar with a significant number of the published IRD techniques explained that he had gained this knowledge from reading on his own. This is commendable, but it is unfortunate that only one in fifteen has taken this extra step.

How can this gap be bridged? Ahituv et al. offer two ways. First, more experimental work on IRD techniques is necessary to determine how different methods perform under different situations. Second, the results of this research must be translated into language that is understood by systems analysts in industry. The paucity of IRA information in DP management periodicals is astounding. It seems to be confined to academic journals where managers and systems analysts are not likely to see it. It is essential that discussion of IRD techniques migrate to publications more widely read by the people who need to know about those techniques.

Additionally, most managers and analysts are not interested in theory, but rather in step-by-step, cookbook approaches to accomplish a task. Hence, Ahituv et al. argue that "structured methodologies based on the research results should be developed." [Ref. 20: p.144] Education of systems analysts in these structured methodologies is vital if we expect use of the methodologies to spread. All formal data processing educational programs (at vocational schools, colleges, and universities) include a course in IRA and that continuing education be provided in the form of seminars.

The basic goal of any program to bridge the conceptual literature-practical application gap should be to ultimately enlarge the "problem space" of systems analysts so that they can intelligently survey the situation, make an informed and deliberate choice of what they believe to be the optimum out of a large repertoire of possible approaches, and then competently determine the users' <u>true</u> requirements. We must achieve this goal if we wish to take full advantage of the capabilities of today's (and tomorrow's) information technology in a timely, economical, and effective manner.

APPENDIX A

ASPECTS OF TAGGART AND THARP'S IRA TECHNIQUE FRAMEWORK

1. Evaluation criteria used: evaluation scope encompassing the analysis phase as well as including operational and technical criteria.

2. Information characteristics: recognize key characteristics of information and their impact on the cost of information needs.

3. Information need scope: recognize the current scope of need satisfaction with the implied potential for expansion in the universe of managers' information need.

4. Degree of sophistication: evaluationary expansion through information systems stages implies increasing sophistication in requirements analysis approaches.

5. Decision process: recognize the need to support the information requirements of the intelligence and design phases as well as the choice phases of the decision process.

6. Decision-making hierarchy: nonprogrammed decision type activity and higher levels in the decision hierarchy require more sophisticated information support which should be considered in requirements analysis.

7. <u>The decision maker</u>: the decision maker as a human information processor exhibits varying degrees of ability on several behavioral dimensions.

8. <u>Organizational environment</u>: the simplicity and complexity of information needs depends on the stability of the organization's external environment and internal structure.

9. <u>Organizational subsystems</u>: a generalized scheme for organizational subsystems provides the analyst with a broadly applicable basis for need determination.

10. <u>Management function and level</u>: the character of information requirements varies with different combinations of management function and level.

Source: [Ref. 2: p.232]

APPENDIX B

CHARACTERISTICS OF PARTICIPANTS

	POSITION	TYPE OF ORGANIZATION	SIZE
1.	Independent Consultant	Private Consulting Business	Small*
2.	Manager of Planning Administration, and Finance	Diet product manufacturer and Distributor	Medium
3.	Chief of Systems Analysis and Programming	Military DP Facility	Small
4.	Systems Leader	Manufacturer of Technology Products	Large
5.	General Manager	Software House	Medium*
6.	Manager of Management Sciences	Major Oil Company	X-Large
7.	Systems Development Group Manager	Investment Firm	Large
8.	Systems Development Manager	Large, Diversified Manufacturing Firm	Large
9.	Financial Data Administrator, Financial Systems Project (Project Team Member)	Engineering and Construction Firm	X-Large
10.	Manager of Product Development and Programming Dept.	Bank	Medium
11.	Manager of Management Consulting	Accounting Firm	Medium*

12.	Head, Requirements Analysis and Design Division	Military DP Service Facility	Medium
13.	Deputy Director of Information Mgt Division	U. S. Government Agency	Medium
14.	Manager of Systems Development	Forest Products Manufacturer	Medium
15.	Vice President for Data Systems Design and Support	Bank	X-Large

* These organizations sell their systems development services to outside organizations; hence, the size is based on approximate yearly revenues vice budget and a different classification scale is used.

APPENDIX C

INTERVIEW QUESTIONS

1. What techniques do you use (or are used here) for determining information requirements?

2. How successful are they?

3. What are the weaknesses of your methods and how do you make up for them?

4. Do you meet any resistance from the users in the use of this technique?

5. Do you have experience with any other techniques? If so, what were the results of using those techniques?

6. (If answer to #5 was "yes") Why do you prefer your methods over these other techniques?

7. Do you try to improve the decision-making process in any way when developing the MIS?

8. Have there been any MIS developed that were unsuccessful?

9. Do you have an Information Center?

If "yes":

10. Do you consider it successful?

11. Is it used solely for special, one-time, and ad hoc reports or is it used for full-scale systems development as well?

12. Do you foresee it being used for systems development in the future?

13. Will it replace traditional MIS departments or will they work together?

14. Are you familiar with or do you use any of the following techniques?
    a. Decision Analysis
    b. Data Analysis

c. Infological Development
d. Human Simulation
e. Paper Simulation
f. Nominal Group Technique
g. Contingency Method
h. Situational Method
i. Prototyping
j. Heuristic Development
k. Critical Success Factors Approach
l. Protocol Analysis
m. Delphi Method
n. Critical Incident Technique
o. REP Test Methodology (Role Construct Repertory Test)
p. Data Base or "Kitchen Sink" Approach

APPENDIX D

TECHNIQUES DESCRIBED IN INTERVIEWS

INTERVIEW NUMBER	TECHNIQUE	FORMAL(F) INFORMAL(I)
1	Look at existing system, organizational goals, current information inputs to decisions.	I
2	Acquire knowledge of business through involvement and conduct interviews.	I
3	Ask, document examination, look at existing system.	I
4	STRATUS system development method: user project teams.	F
5	Ask, or may be found already specified in RFP	I
6	If scope is large, study info flow and managerial objectives; if scope narrow, start with something simple and evolve.	I
7	Interview; familiarization with user environment; geographically dispersed users just write down requirements and send to head office.	I
8	Interview between systems analyst and user liason personnel; some prototyping on short projects.	I/F
9	SDM-70 systems development method; user project teams.	F
10	Pseudo-user project teams: requirements analysis delegated back to systems personnel who ask the users about their needs and then iteratively refine those needs.	I
11	Ask; sometimes group meetings.	I

12	Ask; some direct observation; user reviews of requirements.	I
13	Questionnaires to be followed by group discussion/evaluation.	F
14	METHOD 1 systems development method; direct observation of what users do, then interviews to refine and validate.	F
15	User project teams; some prototyping	F

APPENDIX E

TABULATION OF RESPONSES TO SURVEY QUESTION #14

TECHNIQUE	PARTICIPANT 1	2	3	4	5	6	7	8	10	11	12	13	14	15
a. Decision Analysis	*	+	*	*	0	3	*	*	1	3	*	1	*	*
b. Data Analysis	1	+	*	0	1	3	0	0	3	3	*	*	*	+
c. Infological Devel.	0	0	0	0	0	0	0	0	0	0	0	0	0	1
d. Human Simulation	0	0	0	0	0	0	0	0	0	+	0	0	0	0
e. Paper Simulation	*	0	*	*	*	*	*	*	*	*	+	*	*	*
f. Nominal Grp. Tech.	0	0	0	0	0	0	0	2	0	0	0	0	0	0
g. Contingency Method	0	0	0	0	0	0	0	0	0	0	0	0	0	0
h. Situational Method	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i. Prototyping	1	3	0	1	0	@	3	2	@	@	1	@	@	3
j. Heuristic Devel.	1	0	0	0	0	0	0	@	1	0	@	@	@	0
k. CSF Approach	0	0	0	0	0	0	0	2	0	0	0	+	*	0
l. Protocol Analysis	3	0	0	0	0	0	0	0	0	0	0	0	0	0
m. Delphi Method	0	3	0	0	0	1	0	3	0	0	0	1	1	0
n. Crit. Incid. Tech.	0	3	0	0	0	0	0	0	0	0	0	+	0	0
o. REP Test	0	0	0	0	0	0	1	0	0	0	0	0	0	0
p. Data Base Approach	+	+	0	0	0	0	1	0	0	+	0	0	0	0

Key: 0 = Never heard of it
1 = Heard of it but never used it
2 = Used it once or twice
3 = Use quite often
+= Never heard of it by that name, but a similar concept has been used once or twice informally

* = Never heard of it by that name, but a similar concept is used frequently informally
@ = Heard of it, and sometimes use an informal version

NOTE: Interview #9 has been omitted because the interviewee was a user on the project team, not a DP professional and, hence, would not be expected to be familiar with these techniques.

LIST OF REFERENCES

1. Ackoff, R.L., "Management Misinformation Systems," Management Science, v.14, p.B147-B146, December 1967.

2. Tharp, M.O. and Taggart, W.M., "Management Information Analysis: a situational perspective," Management Datamatics, v.5, p.231-239, December 1976.

3. Norton, J.H., "Information Systems: Some Basic Considerations," Management Review, v. 58, p. 2-8, September 1969.

4. Miller, J.C., "Conceptual Models for Determining Information Requirements," AFIPS, Proceedings of the Spring Joint Computer Conference, 1964, p. 609-620, 1964.

5. Dhar, V. and Davis, J.G., "A Process Model of Information Requirements Analysis for Strategic Decision Support," American Institute of Decision Sciences, Thirteenth Annual Meeting, 18-20 November 1981, Vol. 1, p. 191-193, 1981.

6. Davis, G.B., "Information Analysis for Information System Development," Systems Analysis and Design: A Foundation for the 80's, ed. W. W. Cotterman, J. D. Cougar, N. L. Enger, and F. Harold, p. 41-57, North-Holland, 1981.

7. Cooper, R.B. and Swanson, E.B., "Management Information Requirements Assessment: The State of the Art," Data Base, v. 11, p. 5-16, Fall 1979.

8. Naumann, J.D., Davis, G.B., and McKeen, J.D., "Determining Information Requirements: A Contingency Method for Selection of a Requirements Assurance Strategy," The Journal of Systems and Software, v.1, n. 4, p. 273-281, 1980.

9. Anthony, R. N., Planning and Control Systems: A Framework for Analysis, Harvard University Graduate School of Business, 1965.

10. Simon, H.A., The New Science of Management Decision, Harper and Row, 1960.

11. Gorry, G.A., and Scott Morton, M.S., "A Framework for Management Information Systems," Sloan Management Review, v.13, p. 55-70, Fall 1971.

12. Ein-Dor, P. and Segev, E., Managing Management Information Systems, Lexington Books, 1978.

13. Davis, G. B., "Strategies for Information Requirements Determination," IBM Systems Journal, v. 21, p.4-30, 1982.

14. Canning, R.G., ed., "Getting the Requirements Right," EDP Analyzer, v.15, p. 1-14, July 1977.

15. Guerrieri, J.A., "Establishing True User Requirements," Small Systems World, p.26-32, September 1980.

16. Lederer, A.L., "Information Requirements Analysis," Journal of Systems Management, v.32, p. 15-19, December 1981.

17. Langefors, B., "Analysis of User Needs," Information Systems Methodology, ed. G. Bracchi and P. C. Lockemann, p. 1-38, Spinger-Verlag, 1978.

18. McKeen, J.D., Naumann, J.D., and Davis, G.B., Development of a Selection Model for Information Requirements Determination, Management Information Systems Research Center (MISRC) Working Working Paper 79-06, June 1979.

19. Ein-Dor, P., and Segev, E., A Paradigm for Management Information Systems, Praeger, 1981.

20. Ahituv, N., Munro, M.C., and Wand, Y., "The Value of Information in Information Analysis," Information and Management, v.4, p. 143-150, July 1981.

21. Newell, A. and Simon, H.A., Human Problem Solving, Prentice-Hall, 1972.

22. Miller, G.A.. "The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information." The Psycological Review, v.63, p. 81-97, March 1956.

23. Keen, P.G.W., "Information Systems and Organizational Change," Communications of the ACM, v.24, p. 24-33, January 1981.

24. Parker, E.B., "Behavorial Research in the Development of a Computer-Based Information System," Communication Among Scientists and Engineers, ed. C.E. Nelson and D.K. Pollack, p. 281-292, D.C. Heath, 1970.

25. Courtney, J.F., "Evaluating Information Requirements in MIS Design," Journal of Applied Systems Analysis, v.5, p. 137-141, May 1978.

26. Senn, J.A., Information Systems in Management, Wadsworth, 1978.

27. Krauss, L.I., Computer-Based Management Information Systems, American Management Association, 1970.

28. Heany, D.F., Development of Information Systems, Ronald Press, 1968.

29. Sisson, J.A., "The Management Information System for U.S. Naval Shipyards, Design for the Future," International Federation of Information Processing, Computer Application in the Automation of Shipyard Operation and Ship Design III, ed. C. Kuo, K.J. Mac Callum, and T.J. Williams, North-Holland, 1979.

30. Chapman, E.A., St. Pierre, P.L., and Lubans, J., Library Systems Analysis Guidelines, Wiley-Interscience, 1970.

31. Langefors, B., "Some Approaches to the Theory of Information Systems," BIT, v.3, p.229-234, 1963.

32. Hartman, W., Matthes, H., and Proeme, A., Management Information Systems Handbook, McGraw-Hill, 1972.

33. Levinton, P.H., and Dunning, T.F.E., "Clinical and Management Requirements for Computerized Mental Health Information Systems," IEEE, Proceedings of the Fourth Annual Symposium on Computer Applications in Medical Care, Part III, ed. J.T. O'Neill, p. 1917-1924, IEEE, 1980.

34. Munro, M.C., and Davis, G.B., "Determining Management Information Needs: A Comparison of Methods," MIS Quarterly, v.1, p. 55-67, June 1977.

35. Munro, M.C., "Determining the Manager's Information Needs," Journal of Systems Management, v.29, p. 34-39, June 1978.

36. Zani, W.M., "Blueprint for MIS," Harvard Business Review, v.48, p. 95-100, November-December 1970.

37. Penney, G., "Identifying Information Needs," NCC Interface, p.372-373, March 1975.

38. King, W.R., and Cleland, D.I., "The Design of Management Information Systems: An Information Analysis Approach," Management Science, v.22, p. 286-297, November 1975.

39. Willoughby, J.K., and Gardner, J.A., "User Requirements Analysis in Design of Information Systems," Online Review, Proceedings of the Second National Online Meeting, ed. M.E. Williams and T.H. Hogan, p. 515-530, Learned Information, Inc., 1981.

40. Sackman, H., Delphi Critique: Expert Opinion, Forecasting, and Group Processes, Lexington Books, 1975.

41. Jones, W. O., The Determination of Future User Requirements for an Existing Management Information System Using a Delphic Methodology, Ph.D. Dissertation, University of Georgia, 1978.

42. Remus, W., Sprague, R.H., and Gilbert, P., "Assessing Information Needs Using a Modified Delphi Technique," American Institute for Decision Sciences, Proceedings of the 8th Annual Conference of the National AIDS, p.543-545, 1976.

43. Delbecq, A.L., Van de Ven, A.H., and Gustafson, D.H., Group Techniques for Program Planning: A Guide to Nominal Group and Delphi Processes, Scott, Foresman and Company, 1975.

44. Henderson, J.C., and West, J.M., "Planning for MIS: A Decision-Oriented Approach," MIS Quarterly, v.3, p. 45-58, June 1979.

45. Rockart, J.F., "Chief Executives Define their Own Data Needs", Harvard Business Review, v. 57, p. 81-93, March-April 1979.

46. Mintzberg, H., "The Manager's Job: Folklore and Fact," Harvard Business Review, v.53, p. 49-61, July-August 1975.

47. Davis, G.B., "Comments on the Critical Success Factors Method for Obtaining Management Information Requirements in Article by John F. Rockart, 'Chief Executives Define Their Own Data Needs, Harvard Business Review, March-April, 1979," MIS Quarterly, v.3, p. 57-58, September 1979.

48. Munro, M.C., and Wheeler, B.R., "Planning, Critical Success Factors, and Management's Information Requirements." MIS Quarterly, v.4, p. 27-38, December 1980.

49. Ramsey, H.R., Atwood, M.E., and Willoughby, J.K., "Paper Simulation Techniques in User Requirements Analysis for Interactive Computer Systems," Human Factors Society, Proceedings of the Human Factors Society--23rd Annual Meeting, 1979.

50. Van Cott, H.P., and Kinkade, R.G., "Human Simulation Applied to the Functional Design of Information Systems," Human Factors, v.10, p. 211-216, June 1968.

51. Werner, D.J., Greenburg, A.G., and Goldberg, M., "An Analysis of Decision Making and Information Requirements in an Outpatient Clinic Environment," Proceedings of the American Society for Information, Vol. 8: Communication for Decision Makers, p. 41-44, 1971.

52. Kennedy, M.H., and Mahapatra, S., "Information Analysis for Effective Planning and Control," Sloan Management Review, v.16, p. 71-83, Winter 1975.

53. Head, R.V., "Management Information System Structure," Data Management, v. 9, p.51-53, September 1971.

54. McKeever, J.M., and Kruse, B., Management Reporting Systems, Wiley-Interscience, 1971.

55. McCracken, D.D., "A Maverick Approach to Systems Analysis and Design," Systems Analysis and Design: A Foundation for the 80's, ed. W.W. Cotterman, J.D. Couger, N.L. Enger, and F. Harold, p. 446-451, North-Holland, 1981.

56. McCracken, D.D., and Jackson, M.A., "Life Cycle Concept Considered Harmful," ACM SIGSOFT Software Engineering Notes, v.7, p. 29-32, April 1982.

57. Bally, L., Brittan, J., and Wagner, K.H., "A Prototype Approach to Information System Design and Development," Information and Management, v.1, p. 21-26, 1977.

58. Hancock, J.L., "User Requirements and Systems Productivity," Society for Management Information Systems," Proceedings of the Ninth Annual Conference, p.13-19, September 1977.

59. Online Systems, Inc., OLIVER Reference Manual, 1973.

60. Donovan, J.J. and Madnick, S.E., "Institutional and Ad Hoc DSS and Their Effective Use," Database, v.8, p. 79-88, Winter 1977.

61. Wetherbe, J.C., "Systems Development: Heuristic or Prototyping?", Computerworld, v.16, p. SR/14-SR/15, 26 April 1982.

62. Taggart, W.M. and Tharp, M.O., A Survey of Management Information Requirements Analysis Techniques, Florida International University School of Business and Organizational Science Working Paper 76-1, 1976.

63. Naumann, J.D. and Davis, G.B., "A Contingency Theory to Select an Information Requirements Determination Methodology," IEEE, Proceedings of the Second Software Life Cycle Management Workshop, p. 63-65, 1978.

64. Bariff, M.L., "Implementation Strategies for Information Requirements Analysis," SMIS, Proceedings of the SMIS-UA MIS National Conference on Information Systems Development, p. 55-61, 1978.

65. Synnott, W.R. and Gruber, W.H., Information Resource Management: Opportunities and Strategies for the 1980's, John Wiley and Sons, 1981.

66. Hammond, L.W., "Management Considerations for an Information Center," IBM Systems Journal, v.21, p. 131-161, 1982.

67. Mollen, D.C. and Bakshi, V., "How to Support Company End Users," IBM Data Processor, v.24, p.6-8, May/June 1981.

68. Alloway, R.M. and Quillard, J.A., User Managers' Systems Needs, Center for Information Systems Research, Sloan School of Management, Massachusetts Institute of Technology Working Paper #86, April 1982.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93940	2
3.	Chairman Code 54 Department of Administrative Sciences Naval Postgraduate School Monterey, CA 93940	1
4.	Mr. E. F. Fischer Manager, IS & DP Customer Services 3M Center Bldg 224-4E-02 St. Paul, MN 55144	1
5.	Mr. David Peterson c/o Georgia Pacific Corp. 133 Peachtree St. NE Atlanta, GA 30303	1
6.	Mr. Wino J. Geelen Vice President Data Systems Design and Support 3439 Bank of America 150 Spear St. San Francisco, CA 94105	1
7.	Gordon C. Howell, Code 54HV Department of Administrative Sciences Naval Postgraduate School Monterey, CA 93940	1
8.	Professor Carl R. Jones, Code 54JS Department of Administrative Sciences Naval Postgraduate School Monterey, CA 93940	1

9. LT Paul R. Gardella, Jr., USN 1
NIPSSA-04
4600 Silver Hill Rd.
Washington, DC 20389

10. Naval Postgraduate School 1
Computer Technology Curricular Office
Code 37
Monterey, CA 93940

END

FILMED

10-83

DTIC